CSE 123b
Communications Software

Spring 2003

Lecture 10: Mobile Networking

Stefan Savage

Quick announcement

- My office hours tomorrow are moved to 12pm

Last class

- Multicast communications
  - One-to-many
  - Publish and subscribe model (receiver-based)

- Routing protocols
  - Per-source tree routing
    - » RPF, RPB, RPM
    - » Builds Efficient trees
    - » S*G state explosion for large networks/groups
  - Shared tree
    - » Unicast to rendezvous point
    - » More complex, fragile, hard to manage
    - » Trees inefficient by as much as 2x
    - » Only requires G state on routers

Today's issues

- What are implications of hosts that move?
  - Remember routing? It doesn't work anymore…
- Problem review
- Design issues
- Case studies
  - Mobile IP [Johnson96]
  - TCP Migrate [Snoeren et al00]

The Mobility Problem

- Implicit assumption that Internet hosts are fixed
  - IP addresses used to name hosts; cached by higher layers
  - IP routing breaks if addresses change location. **Why?**
- Unfortunately, the buying public likes mobility

Problems

- How does a mobile host get a local IP address?
- How do you know which IP address to use when sending to a mobile host?
- If a host moves during communication how do you know how to migrate state to the new IP address?
- Backwards compatibility (higher-layer state caching)

Application demands

- **Geographic scope**
  - Switching 802.11 LANs at UCSD vs visiting IBM in Zurich
- **Rate of change**
  - Cell-phone in airplane vs hotel room dialin
- **Continuity**
  - State associated with session (e.g. ordering plane tickets)
- **Interactivity**
  - Cell-phone handoff?
- **Remote accessibility**
  - Client vs Server

Some simple solutions

- **Datalink layer mobility**
  - AP's connected to single virtual LAN (UCSD campus solution)
  - Underlying Ethernet switches "learn" which MAC addresses they use and tell other switches
  - Pro: transparent to all higher layers
  - Con: Local scope only
- **Dynamic Host Configuration Protocol (DHCP)**
  - Request IP address dynamically (special broadcast address)
  - How do you get contacted at new IP address?
    - » One solution: dynamic DNS
  - Authentication issues (who can use 802.11 in AP&M?)
  - Pro: Great over longer time scales…
  - Con: What happens during a session?

Mobile IP: Johnson96

- **Current IETF proposed standard for mobility**
  - Dates back to research in the early 90s
  - IPv4 (RFC 2002), IPv6 version is roughly the same

- **Design constraints**
  - Network layer solution
  - Only requires changes to mobile hosts
  - Stationary hosts oblivious to mobility
  - Incrementally deployable

Mobile IP Approach

- **Mobile Host (MH) has two addresses**

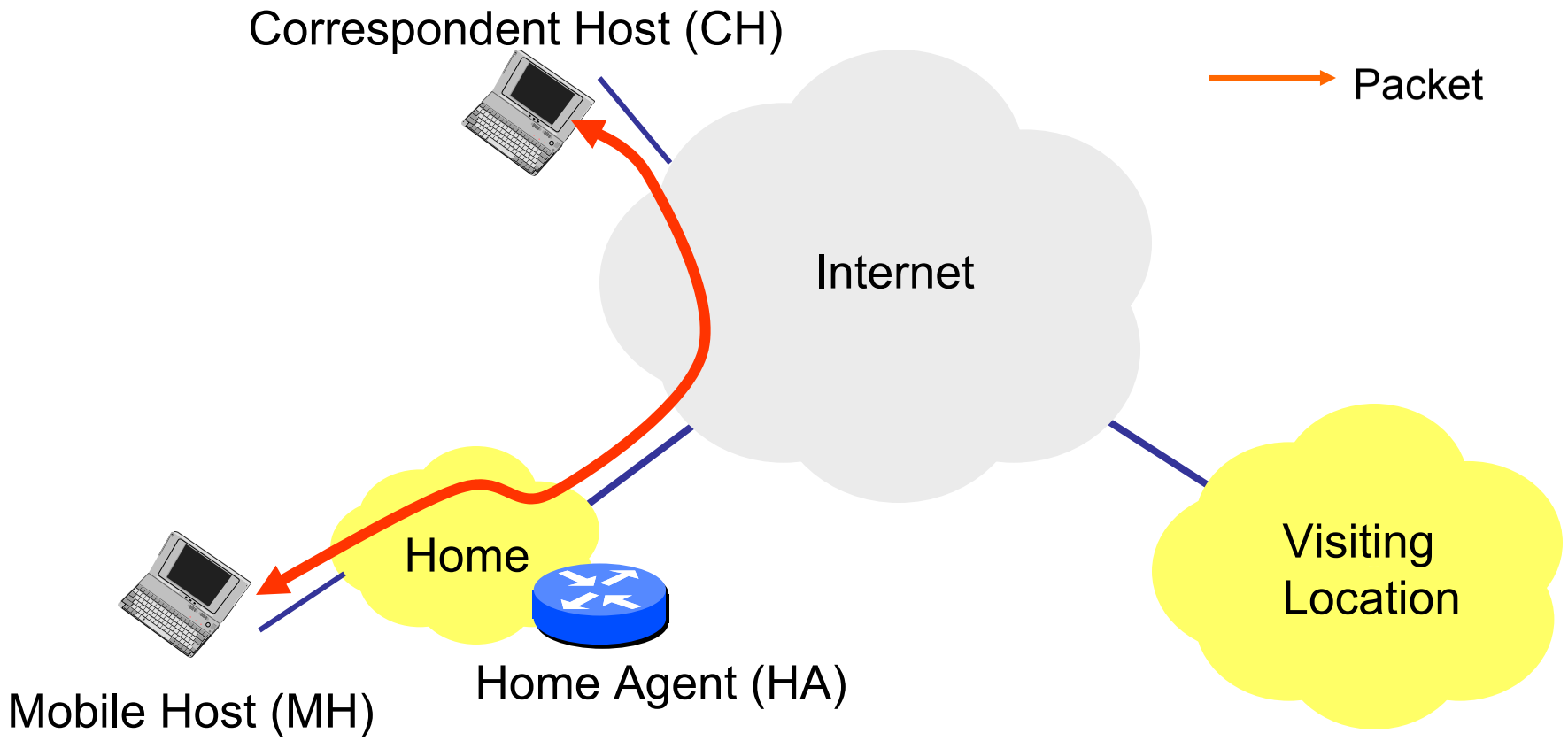
- **Home address**
  - Never changes, uniquely identifies the host
  - In “home network”
  - Correspondent host (CH) addresses all packets to the home address
- **Care-of address**
  - Will change, perhaps frequently
  - In “foreign network”
  - Related to current location (IP routing gets it to the right place)

Home and Foreign Agents

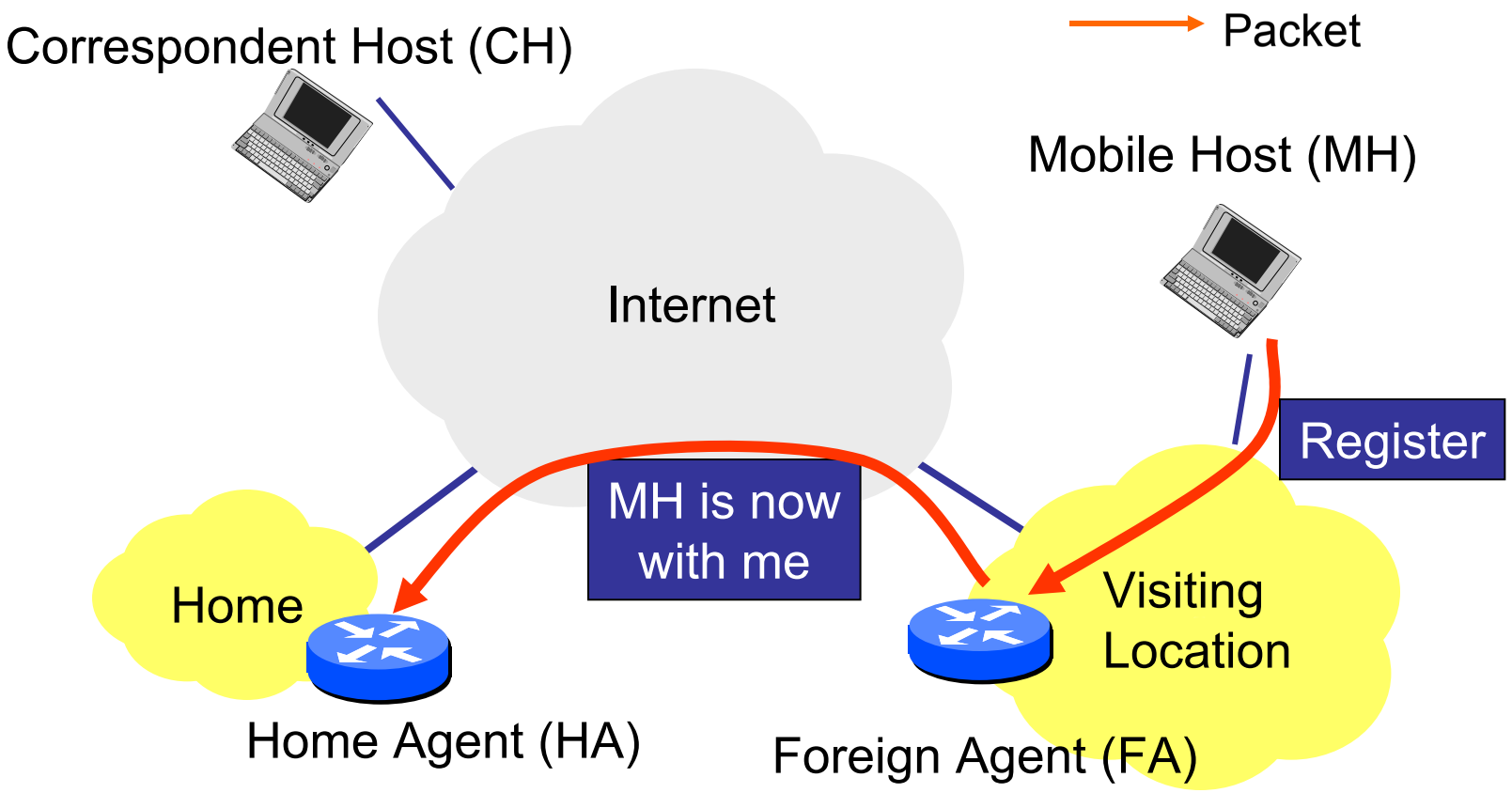
- **Home agent** (HA) implements level of indirection between the mobile host and correspondents
  - Accepts traffic sent to home address
    - » What about requests ***from*** home network?
  - Tunnels traffic to the mobile host (using care-of address)
  - And vice versa, correspondent none the wiser

- **Foreign agent** (FA) represents mobile in foreign network
  - Foreign agent can be care-of address
    - » Mobile host does not need its own address in foreign network
    - » Potential advantage: deal with local mobility locally

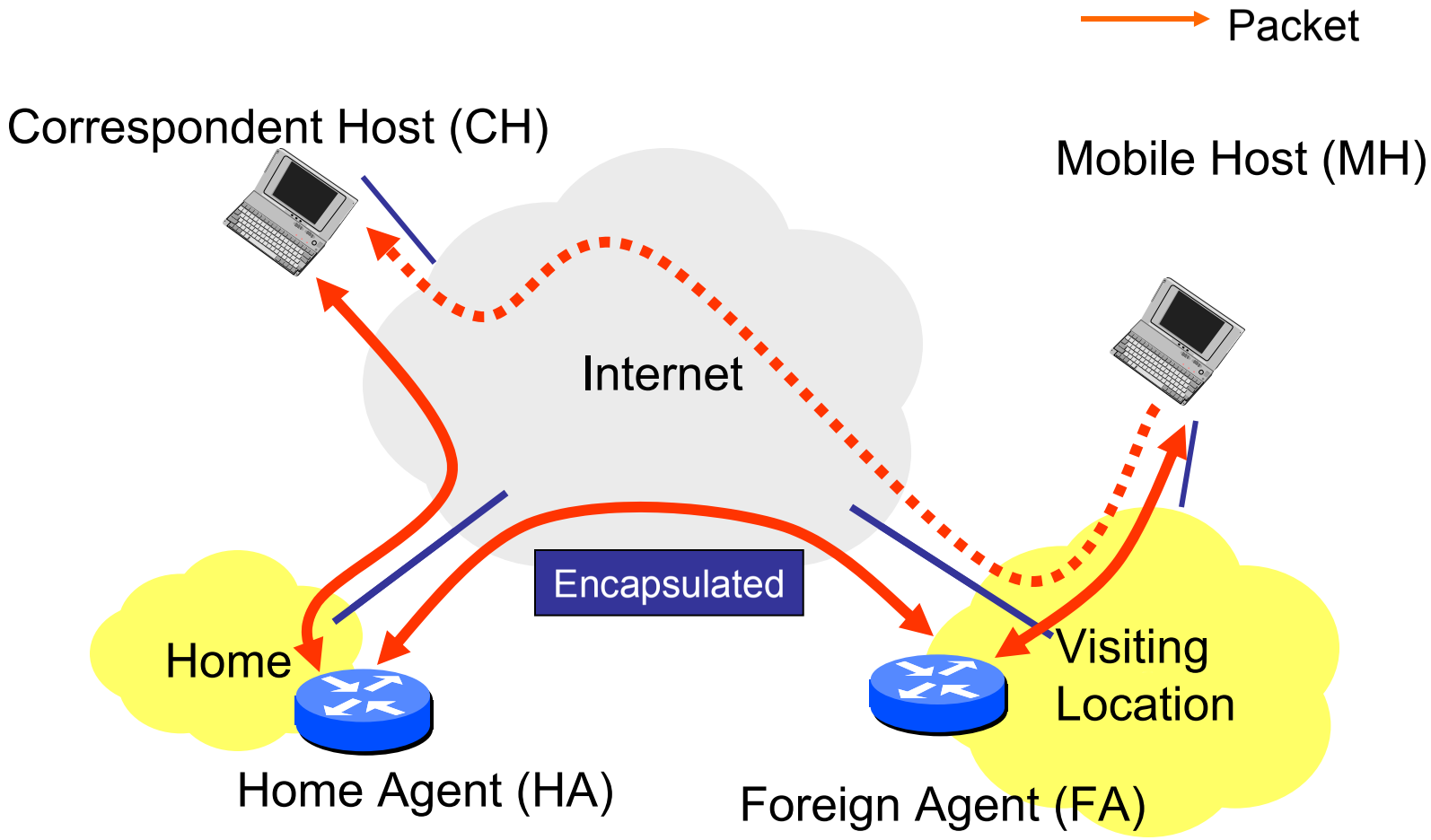
Mobile IP (MH at Home)



Mobile IP (MH Moving)



Mobile IP (MH Away)



Mobile IP Issues

- To make all this happen, a number of issues have to be addressed
  - Discovering agents
  - Registering addresses with agents (establishing bindings)
  - Authentication
  - Tunneling
  - Performance (!)

Agent Discovery

- **Agent discovery** enables a mobile host
  - To notice when it changes networks
  - To notice when it is home again
    - When home, take down the tunnel
  - To find a foreign agent to register with
- Agents multicast **agent advertisements** locally
  - Beacons that tell the mobile who it can hear
  - Start in network A, move to network B
  - Lack of A's beacons and presence of B's tells mobile it has switched networks
- Mobile can also multicast an **agent solicitation**
- Why does multicast work here?

Registration

- Mobiles must register care-of addresses with their home agents
  - So that the home agent knows where to tunnel packets
  - Registration needs to be updated when location changes
- Multiple steps
  - Registration requests first go to foreign agent, then to home agent, which replies to foreign agent, which forwards back to the mobile
- Lifetimes
  - Registrations have TTLs

Registration Authentication

- Registration requests can be used by attackers to hijack tunnels from home agent
  - Hey, send all the mobile's traffic to me now
- Need to authenticate that a registration
  - Came from mobile host (authenticity)
  - Has not been altered (integrity)
  - Has been sent recently (freshness)
- Mechanisms
  - Shared keys (mobile and home are from same admin domain)
  - MD5 digests (secure hash over the data)
  - Nonces or timestamps

Tunneling

- Home agent and mobile communicate using a tunnel
  - IP in IP encapsulation
- Original packet
  - Correspondent address (src) → mobile home address (dest)
  - Gets sent to home agent
- Tunnel packet
  - Encapsulates original packet
  - Home agent (src) → care-of address (dest)
  - Gets sent to foreign agent (or mobile, depending on care-of)
  - Mobile can respond back directly (which source address?)
- Asides
  - Bit of overhead (20 byte header for every packet…poor telnet)

Performance

- The good: No overhead in local operation
  - Home agent out of picture, no longer intercepts packets
  - The common case?
- The bad: Significant overhead in remote operation
  - Triangle routing: Packets between two hosts separated by inches can travel 1000s of miles
  - Wide-area effects can determine “local” connection performance
  - The uncommon case? Even so, a steep price to pay
- Hence: Route optimization

Route Optimization

- Route optimization shortcuts the triangle
  - Correspondents can learn and use mobile care-of addresses
  - Tunnel packets directly to care-of address, skip home agent
  - Requires changes to correspondents
    - » Or to routers
- Issues
  - Binding cache updates (consistency)
  - Binding update authentication (more trust)
  - Yet more complexity
  - Necessary for scalability?
- End result: Mobile IP is not widely deployed

Alternative: Transport-level mobility [Snoeren00]

- Same goals
  - Do not disrupt connections when network address changes
- Different approach
  - Combination of DNS naming and connection migration
  - Naming + transport (vs. network-layer w/ Mobile IP)
  - Based upon observation of how connections are made from mobile
- Three components
  - Addressing
  - Locating mobile hosts
  - Connection migration

Addressing

- Mobiles obtain an network-local IP address
  - No home agent, no home address
  - No foreign agent
  - No tunneling
  - Communication between correspondent and mobile uses addresses directly
- Problem: How does the correspondent learn the mobile's address?
  - If the mobile initiates the connection, the mobile tells the correspondent its address with the SYN packet
  - What about mobile servers?

Locating Mobiles

- Observation: Whenever connections are established, a DNS lookup is performed (e.g. www.yahoo.com)
- Idea: Use the DNS lookup to return latest mobile address to correspondent
  - In Mobile IP, home address is used to unique identify mobile
  - In Transport-level mobility, DNS name is used for this purpose
  - When mobile moves and obtains a new IP address, it updates its DNS entry (e.g. myhost.ucsd.edu is now at 132.239.10.1)

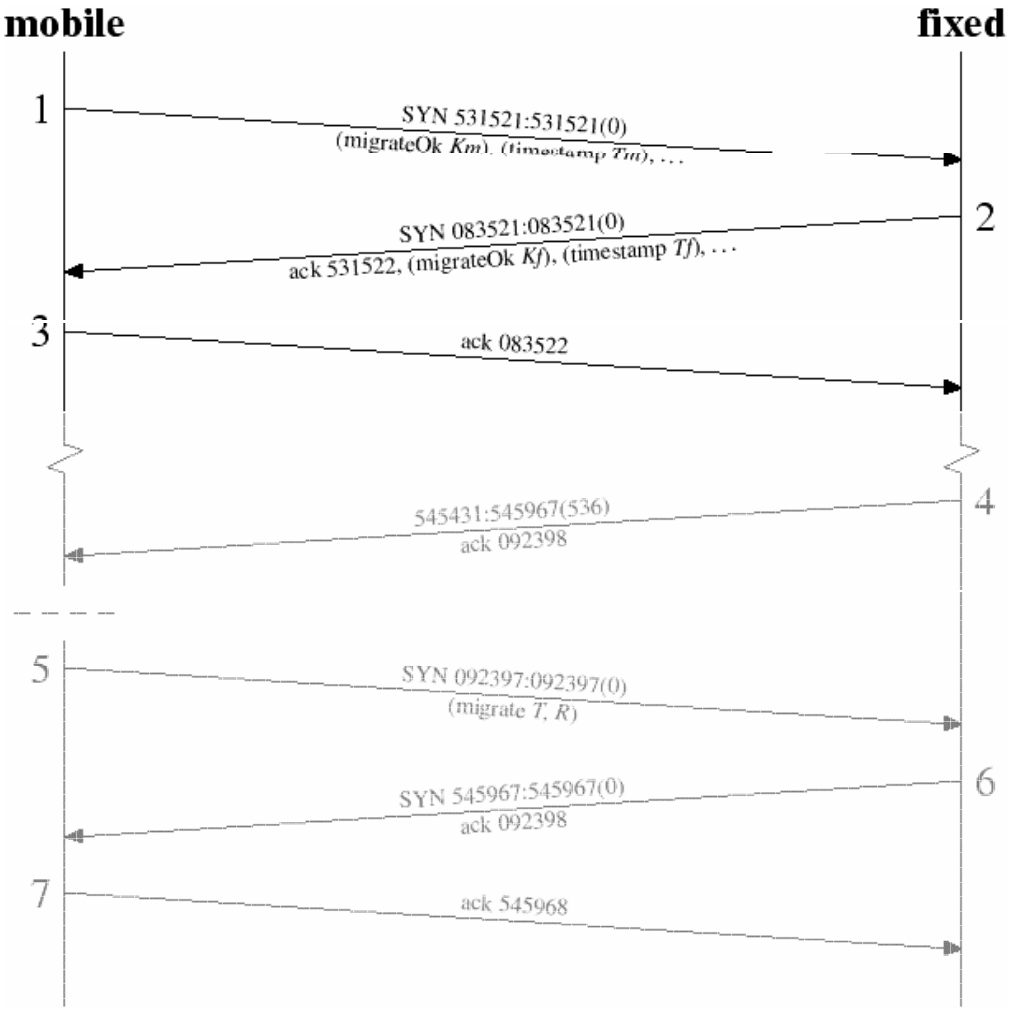
Connection Migration

- Problem: What about existing open connections?
- Solution: TCP Connection Migration
  - New TCP Option: Migrate
  - Negotiated with Migrate-Permitted option in SYN
  - Requires modification to TCP stacks at both ends

Basic Idea

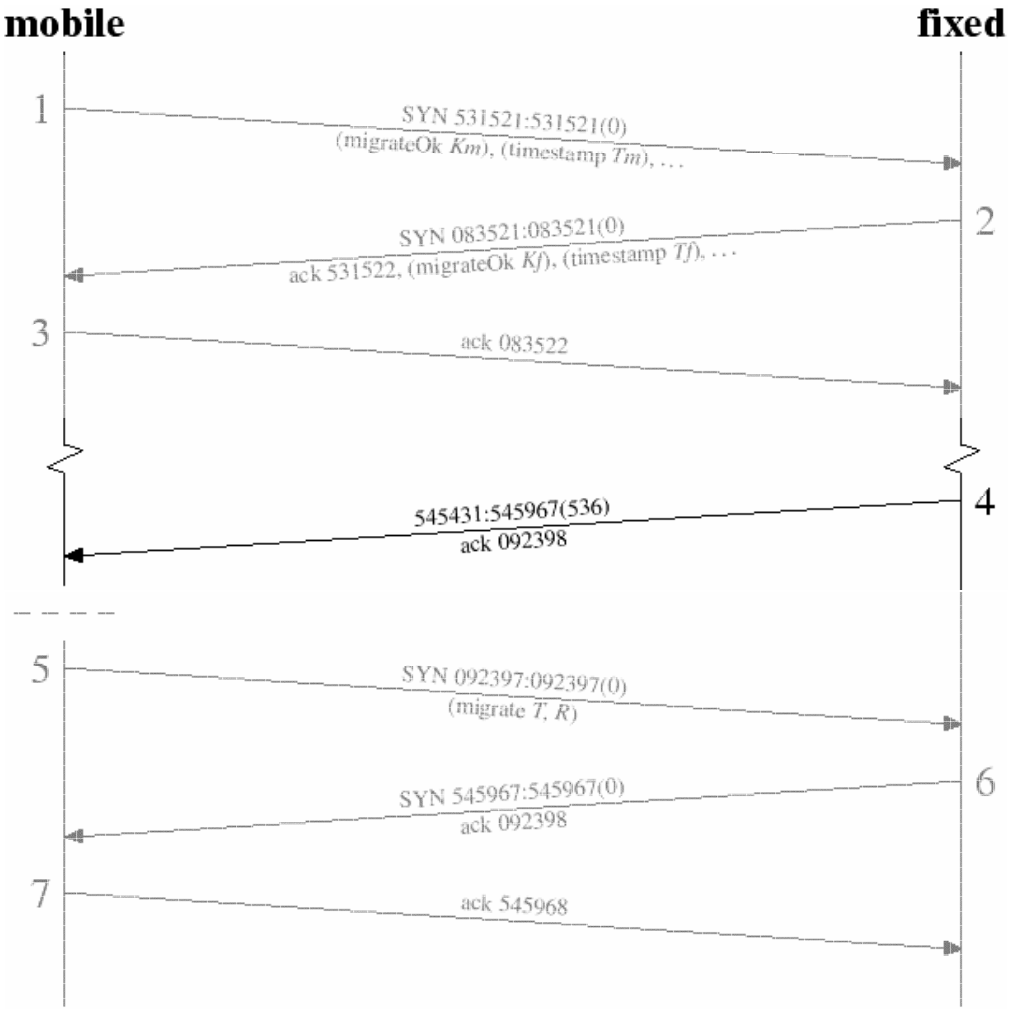
- We have an open connection between correspondent (src) and mobile (dest)
  - Doesn't matter who initiated the connection
  - Connection represented by
    - » <src IP, src port, dest IP, dest port>
  - Mobile moves
    - » Now has new <dest IP*, dest port*>
  - Want to change connection to
    - » <src IP, src port, dest IP*, dest port*>
  - Mobile creates a new connection to the correspondent, forces correspondent to migrate old connection to new one
  - Uses token to show that connections are associated

TCP Migration example



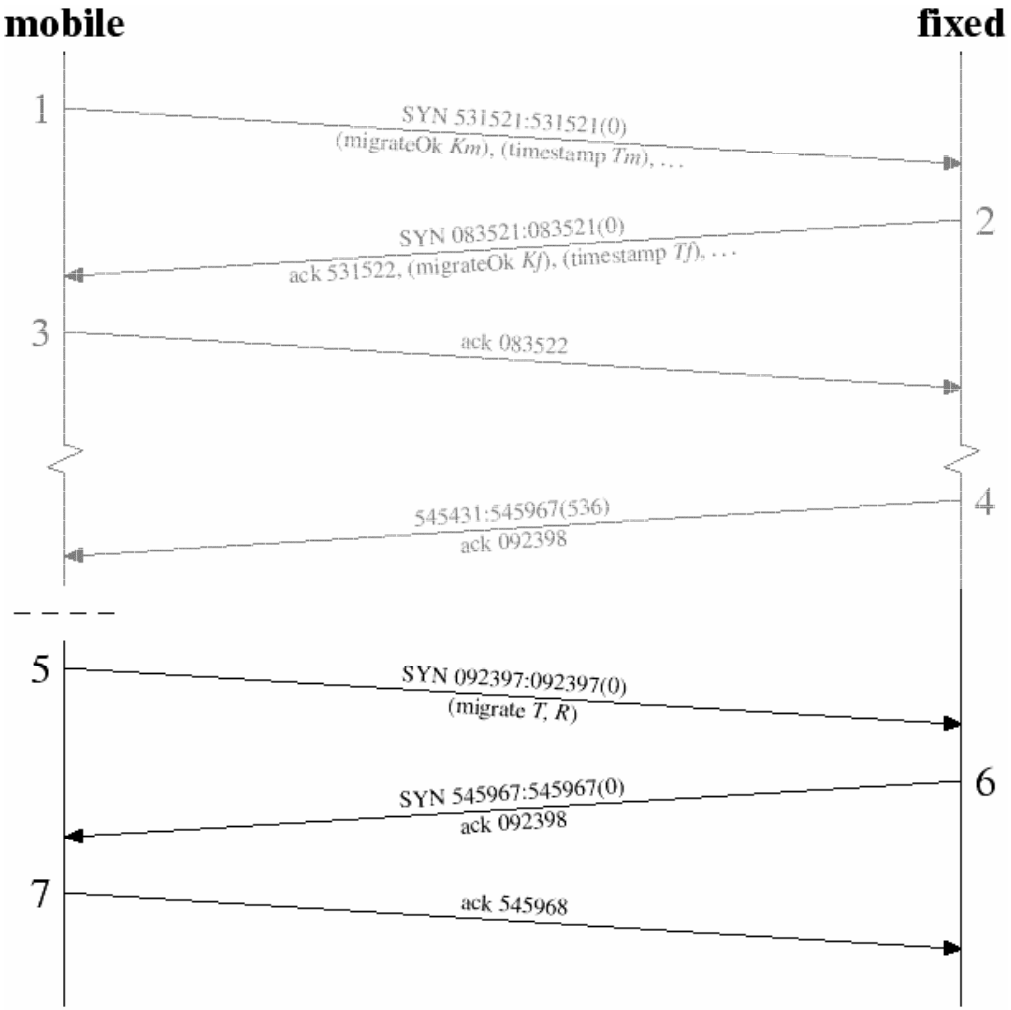
1. Initial SYN
2. SYN/ACK
3. ACK (with data)
4. *Normal data transfer*
5. Migrate SYN
6. Migrate SYN/ACK
7. ACK (with data)

TCP Migration example



1. Initial SYN
2. SYN/ACK
3. ACK (with data)
4. *Normal data transfer*
5. Migrate SYN
6. Migrate SYN/ACK
7. ACK (with data)

TCP Migration example



1. Initial SYN
2. SYN/ACK
3. ACK (with data)
4. *Normal data transfer*
5. Migrate SYN
6. Migrate SYN/ACK
7. ACK (with data)

Issues

- Pro
  - No change to routing infrastructure
  - No triangle routes
  - Simple
- Con
  - Requires changes to both end-points
  - TCP-specific

Next time

- Midterm☺