Identifiers and Network Association

Prof. C. Tschudin, M. Sifalakis, T. Meyer,
G. Bouabene, M. Monti

University of Basel

Cs321 - HS 2011

Overview

- Identifier Schemes (Names, Addresses, Labels)
- Network Association
  - Dynamic Host Configuration Protocol
  - Stateless Address Auto-configuration Mechanism
  - 802.11 Radio-network association
- Multiple identifier schemes in network association
  - HIP
  - LISP
  - SHIM6

Fundamental aspects of networking

- *Association*
  - How to join/participate in a network
- *Topology* management
  - Who can “speak” with who
- *Routing*
  - How to reach others
- *Resource sharing and Communication*
  - How information exchange takes place
  - … and to allow others to do so too. I.e. sharing the network!

Association

- Membership in the network: communication potential
  - Access resources, across systems, via channels
- Agreement on a shared view of the communication space
  - Which resources, computers and channels
- Chicken-and-egg problem ?
  - "shared view of space enables communication … but how can consensus on a shared view be reached before enabling communication" ?

Divide and conquer

- Solution: divide in 2 sub-problems:
  i. Agreement on scheme for identifying users, computers, channels
  ii. Identifier acquisition mechanisms
     - Request-Allocate : The network decides
     - Select-Advertise: The client decides
- Possibly multiple identifiers involved
  – Different combined operations may use different identifier spaces or combination thereof

Elements of Identifier schemes

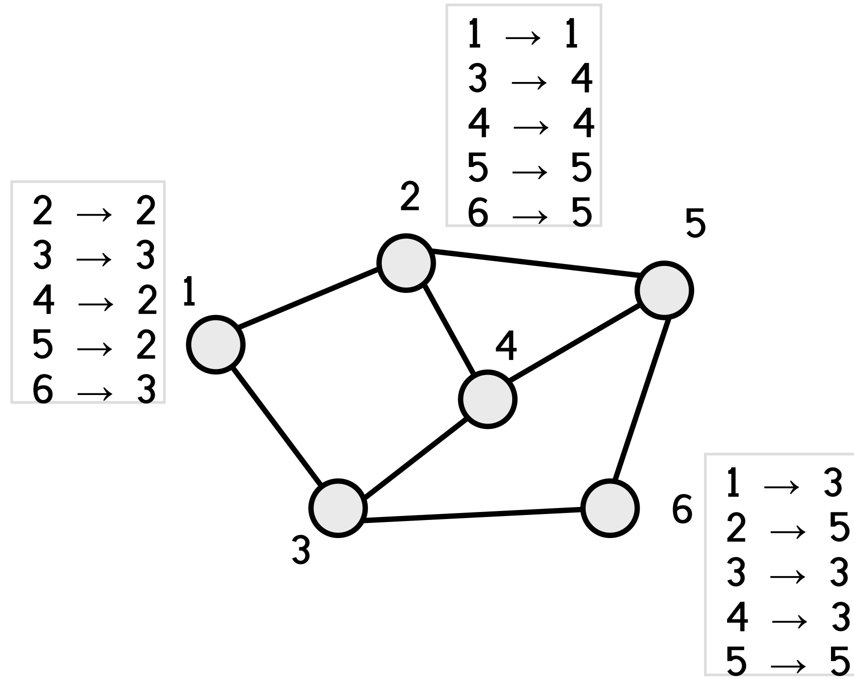
- Alphabet
  - Numbers (Base 10, Base 2, Base 16, …)
  - Letters and symbols (typically Latin alphabet, ASCII charset, …)

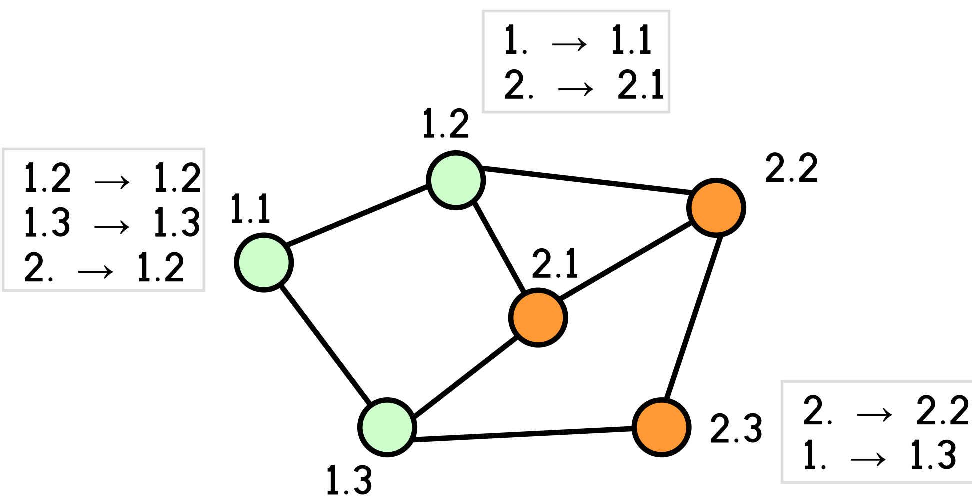
Elements of Identifier schemes

- Alphabet
  - Numbers (Base 10, Base 2, Base 16, …)
  - Letters and symbols (typically Latin alphabet, ASCII charset, …)
- Structure
  - *Flat*: no structure, they can be chosen randomly
  - *Hierarchical*: organisational significance, series of components that position them in a hierarchy for technical convenience
  - Hybrid or other

Hierarchical versus Flat

- *Flat* implies lack of structure in the identifier space
  - Not scalable storage: N entries require N-1 records everywhere





- *Hierarchical* entails structure (organisational hierarchy)
  - More effective storage
  - But imposes allocation constraints

Elements of Identifier schemes

- Alphabet
  - Numbers (Base 10, Base 2, Base 16, …)
  - Letters and symbols (typically Latin alphabet, ASCII charset, …)
- Structure
  - *Flat*: no structure, they can be chosen randomly
  - *Hierarchical*: organisational significance, series of components that position them in a hierarchy for (functional) convenience
  - Hybrid or other
- Semantics
  - *Names*: have functional significance (What ?)
  - *Addresses*: have topological significance /location (Where ?)
  - *Labels*: semantic-free

Different network operations based on the predominant functions on identifiers (search, store, locate, route) determine a selection of identifier scheme.

Some real world use of identifiers

- Street names/numbers (flat struct., location)

    E.g. *Bernoullistrasse 16*
- Telephone system

    PSTN (hierarchical struct., location): E.g. *+41-61-267-0395*

    Mobile (flat struct., person): E.g. *0789499631*
- Names and Surnames of people
    - Used to be names (functional significance), describing quality or role!

        E.g. *Alexander = "defender of people", Sifalakis="healer", Schmidt= "farrier or blacksmith"*
    - Surnames also have grouping semantics (familly/relatives)
- Geo-coordinates (2-level hierarchy, location in a geometrical space):

    E.g. *47.599998N, 7.516667E*
- Postal addresses (hierarchically combined identifiers, location)

    E.g. *Bernoullistrasse 16, Basel, Switzerland*

Internet world identifiers

- Fully Qualified Domain Names (hierarchical struct., reflects location in organisational structure)

  E.g. *www.cs.unibas.ch :* often first part describes role
- IP addresses (hierarchical struct., location in network topology)

  E.g. *(IPv4) 143.233.5.56*

  E.g. *(IPv6) 2001:0660:3003:0001:0000:0000:6543:210F*
- MAC addresses (partly flat struct., labels)

  E.g. *00-13-02-7C-BC-D5 :* usually 3-bytes Manufacturer, 3-bytes: flat
- Port service numbers (flat, labels but implicitly associated with services)

  E.g. *23 (telnet), 20 (ftp), 80 (http)*
- Email addresses (hierarchically combined identifiers, mailbox label or name and organisational location)

  E.g. *sifalakis.manos@unibas.ch*

Relating & Combining identifier spaces

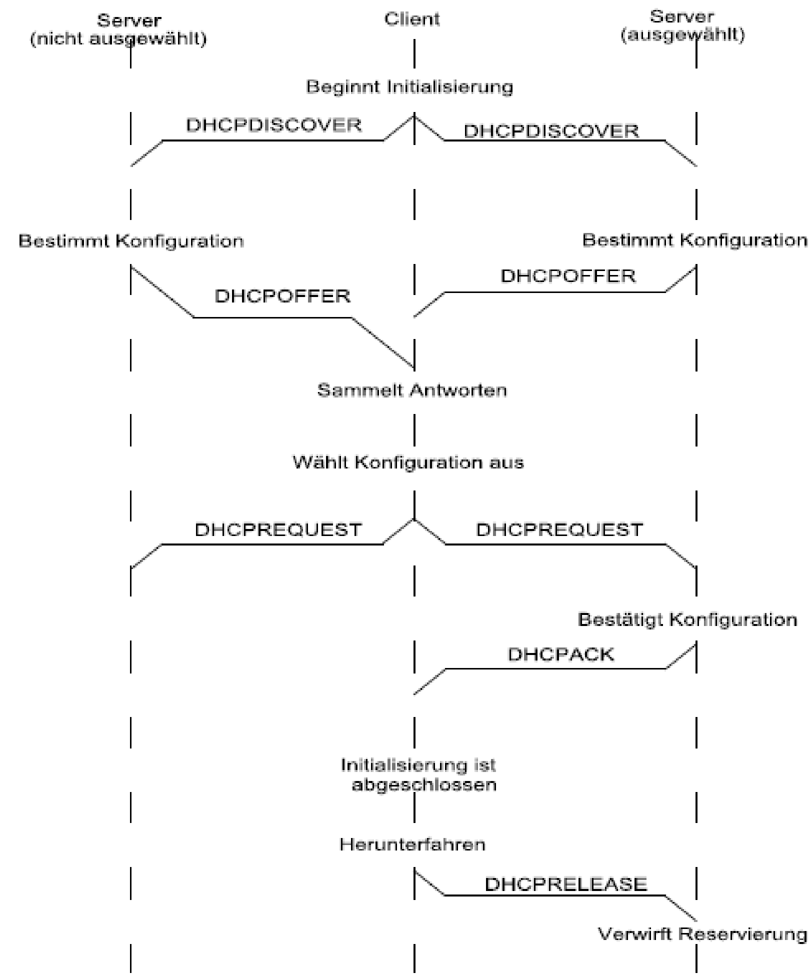
- Translation from one identifier space to another is facilitated through a directory/mapping/translation system
  - DNS: *1.2.3.4 ↔ myhost.mydomain.net*
  - White pages: *M. Sifalakis ↔ +41-61-267-0395*
  - LDP: *M. Sifalakis ↔ sifalakis.manos@unibas.ch*
  - ARP: *00-13-02-7C-BC-D5 ↔ 143.233.5.56*
  - NAT: *143.233.5.56 ↔ 10.1.2.3*
- Composite identifier schemes are used for functions that necessitate communication which is part-taking at multiple levels
  - SMTP over IP: sifalakis.manos @ smtp.unibas.ch
(name @ net-domain)
  - Postal service: M. Sifalakis, Hegenheimerstr 85
(person-name, home address)
  - Service endpoint: http:://143.235.6.5:80
(protocol://net address:port)

Identifier acquisition mechanisms

- In the early days, manual configuration of identifiers was the norm. Still practiced…
- With node mobility network association became episodic and short lived
- 3 common mechanisms for automatic identifier acquisition
  - Dynamic Host Configuration Protocol
  - Stateless Auto-configuration (SAA)
  - Auto-association

DHCP - Dynamic Host Configuration Protocol (1/2)

- DHCP server holds all configuration profiles and options for clients
  - More than 100 configuration options
  - host receives complete network configuration via a DHCP server



© Figure by Uwe Homm

DHCP - Dynamic Host Configuration Protocol (2/2)

- Advantages
  - **1** manually-configured system, **N** automatically configured hosts
  - Address reuse
    - Only enough addresses for max number of simultaneously active systems
  - Some degree of access control (MAC based)

- Disadvantages
  - Manual configuration of server required
  - Server is a central point of failure
    - In practice network operators run backup DHCP servers
  - Fully-distributed managed systems with no centralized control authority cannot be based on DHCP servers (e.g. Berlin Freifunk mesh network)

SAA - IPv6 Stateless Address Auto-configuration (1/2)

- IPv6 addresses are represented in human readable form (!) like this ☺ :

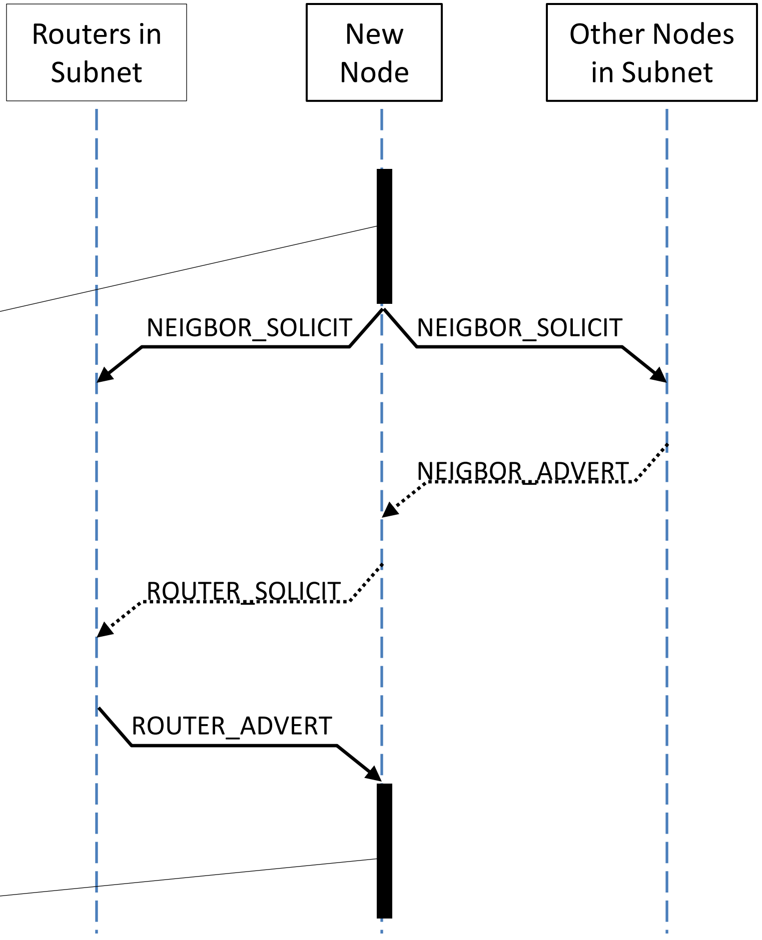
  2001:0660:3003:0001:0000:0000:6543:210F

- A new node on the network generates a tentative *Link-Local (link scope)* address on the interface.

  Auto-configured address:
  fe80:0000:0000:0000:0219:99ff:fe0f:2384

1. Typically uses a variation of the MAC ID as a basis. Checks uniqueness of address (transmits on that address and waits for responses).
2. Then waits an advertisement or explicitly solicits on-link routers for the subnet prefix
3. Finally, re-configures its address with the subnet prefix so as to have global scope

   Re-configured address:
   2001:620:200:0000:0219:99ff:fe0f:2384



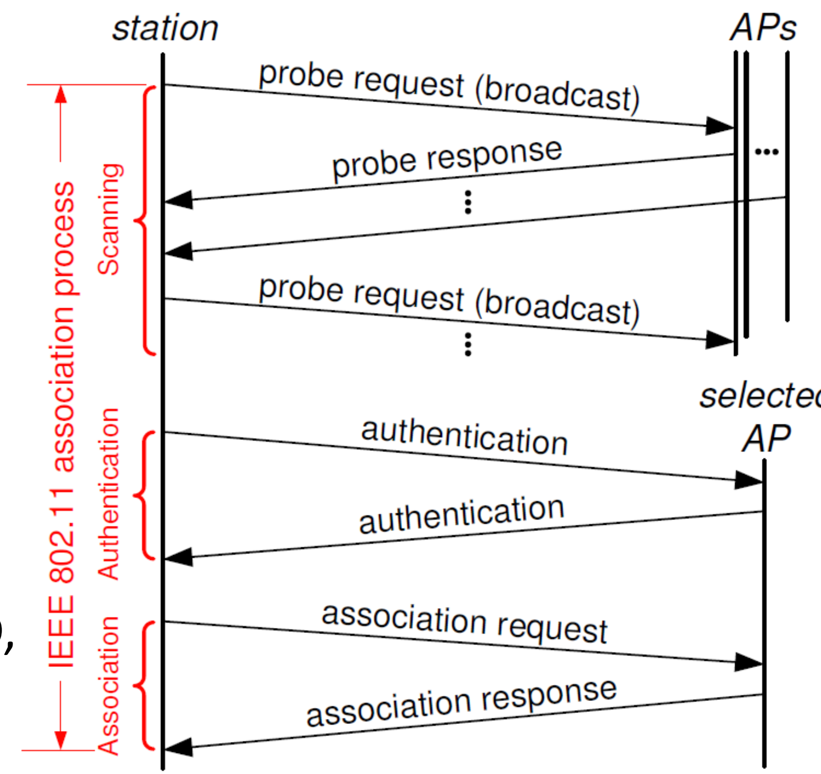
SAA - IPv6 Stateless Address Auto-configuration (2/2)

- Advantages
  - **1** manually configured system, **K** automatically configured routers, **N** automatically configured hosts
  - The IETF IPv6 working group has proposed a prefix delegation mechanism via DHCPv6
    - Delegating router runs a DHCPv6 server
    - Requesting router runs a DHCPv6 client
    - Options sent by server are called IA_PD (Identity Association – Prefix Delegation) and IAID (IA-Identifier)
    - IA_PD contains prefix, prefix length, and lifetime. E.g. 2001:620:200::/48
    - Requesting router can create subnets at will: e.g. 2001:620:200:1::/64 from the delegated prefix
- Disadvantages
  - Limited to address auto-configuration. The other useful options (e.g. DNS server address) are handled by … DHCPv6
  - No access control

IEEE 802.11 Radio-Association

1. Scanning Phase
   - *Active*: Node broadcasts PROBE_REQ on all possible freq. channels
   - *Passive*: Node listens on all freq. channels for periodic BEACON frames by available Access Points
2. Select an AP based on received S/N (highest-signal-strength), or other criterion
3. Exchange authentication information (e.g. WEP, WPA, etc)
4. If authentication successful send ASSOSIATION_REQ with data rate, SSID, and other info to synchronise radio

NOTE: Analogous process in cellular networks with mobile phones



© Figure by Heeyoung Lee, Seoul National University

Need for multiple identifiers

- The more dynamic/complex the service environment, the more identifier schemes needed to satisfy orthogonal requirements w.r.t. network association
  - E.g. Overlays, Mobility, Multi-homing
- Trade-off between
  - *Simplicity*: Re-using same identifier scheme for many different functions
  - *Efficiency*: Using different identifier schemes for different functions
- Canonical Example: “Locator-Host Identifier split”
  - HIP protocol
  - LISP protocol
  - SHIM6 mechanism

Location-ID / Host-ID split Problem Statement

- Most known Internet identifiers
  - FQDN or DNS names
    - Have a functional purpose. E.g. mail.unibas.ch, www.unibas.ch, ftp.unibas.ch
    - Used as a human mnemonic. E.g. the Web server of Basel Uni in Switzerland
  - IP address:
    - Used to specify which subnet a computer is located

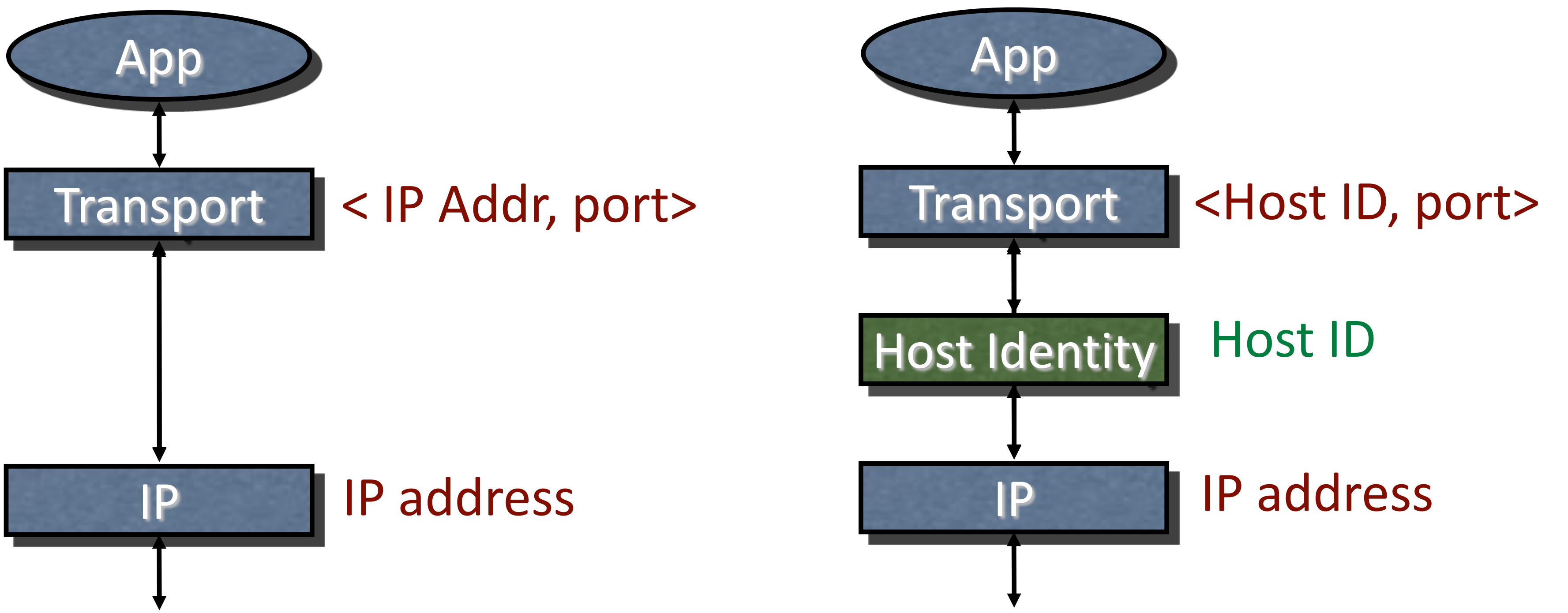
      E.g. 1.2.3.4/24 → a computer in the subnet 1.2.3.0
    - Used (combined with a port no) as endpoint for a communication session

      E.g. 1.2.3.4:6130 → a media stream to a computer in 1.2.3.0 subnet

  - *What happens if the node has a second interface on subnet 5.6.7.8/24 and wants to load-balance the media stream across the two interfaces ?*
  - *What happens to the media stream if the node moves to another subnet ?*
  - *What happens if the user moves to another device and wants to continue receiving the media stream there ?*

HIP: Host Identity Protocol (1/3)

- RFC 5201: provides a method of separating the communication endpoint identifier from node location (IP address)
  - An additional to IP and DNS identifier scheme: public keys, called *Host Identifiers* (HIs)
  - A protocol for generating session keys and discovering/authenticating bindings between HIs and IP addresses (based on the D-H algorithm)

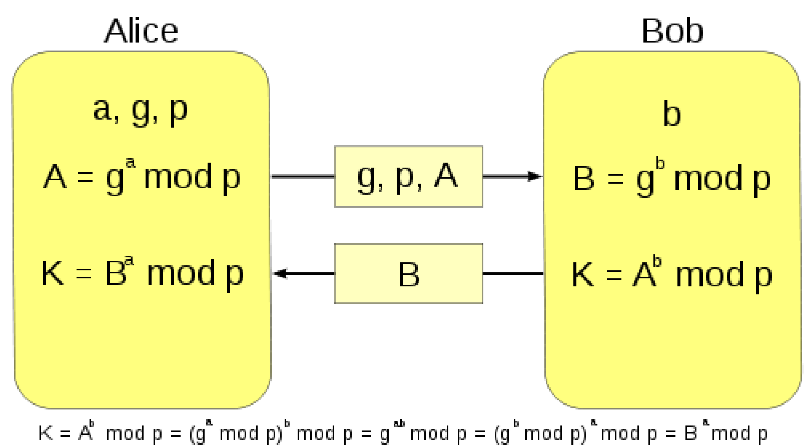


HIP: Host Identity Protocol (2/3)

Diffie-Hellman key-exchange

Alice				Bob		
Secret	Public	Calculus		Calculus	Public	Secret
	p, g				p, g	
a						b
		g^a mod p			…	
	…		→	g^b mod p		
$(g^b \text{ mod } p)^a$ mod p			← =			$(g^a \text{ mod } p)^b$ mod p

1. Alice and Bob agree to use a prime number p=23 and base g=5.
2. Alice chooses a secret integer a=6, then sends Bob A = g^a mod p
   - A = 5^6 mod 23 = 8.
3. Bob chooses a secret integer b=15, then sends Alice B = g^b mod p
   - B = 5^{15} mod 23 = 19.
4. Alice computes **s** = B^{a} mod p
   - 19^6 mod 23 = **2**.
5. Bob computes **s** = A^{b} mod p
   - 8^{15} mod 23 = **2**.

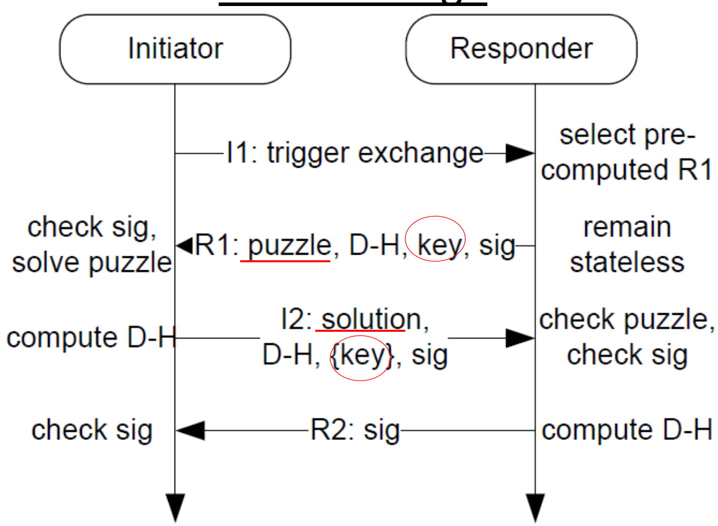


© Figure by Benutzer DaMutz

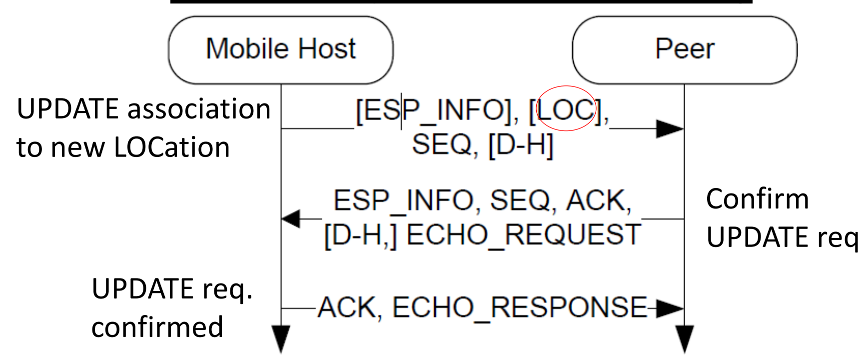
HIP: Host Identity Protocol (3/3)

- *Host Identifier* (HI) : Cryptographic public key (RSA or DSA)
  - The private key is used as secret for computation of D-H session keys
- *Host Identity Tag* (HIT) : 128-bit hashes derived from HI
  - Fixed length: replace use of IP addresses as communication endpoints for layers above IP

Generation of HIP session keys
D-H exchange



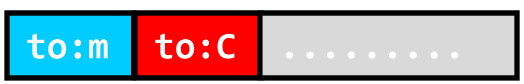
Update of active HIP session key to IP addr. binding



 (See RFC 5201 and RFC 5683 for more information)

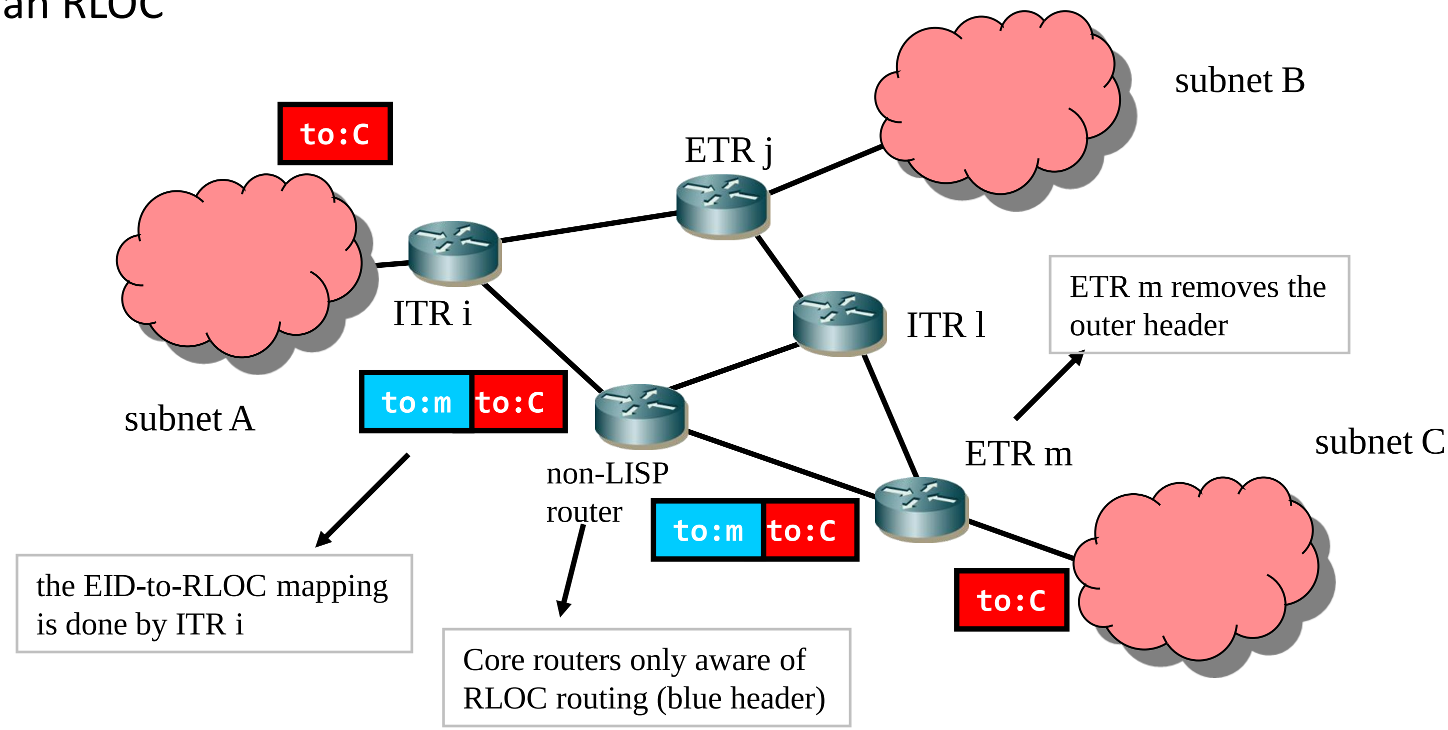
LISP: Locator ID Separation Protocol (1/2)

- HIP introduces an additional flat identifier scheme on top of the hierarchical one of IP, to decouple communication sessions identifiers from network address identifiers
- LISP divides and superimposes use of IP address identifiers as two isolated identifier spaces
  - Edge-network ID space: → where IP = EID (end-point ID)
  - Core-network ID space: → where IP = RLOC (route locator)
- Basic idea (LISP v1.5): Overlay IP topologies
  - Border routers en-capsulate packets traveling towards the Internet core
  - Border routers de-capsulate packets coming from the core
  - End-point identifiers (EIDs) are used in inner header
  - Route locators (RLOCs) are used in outer header



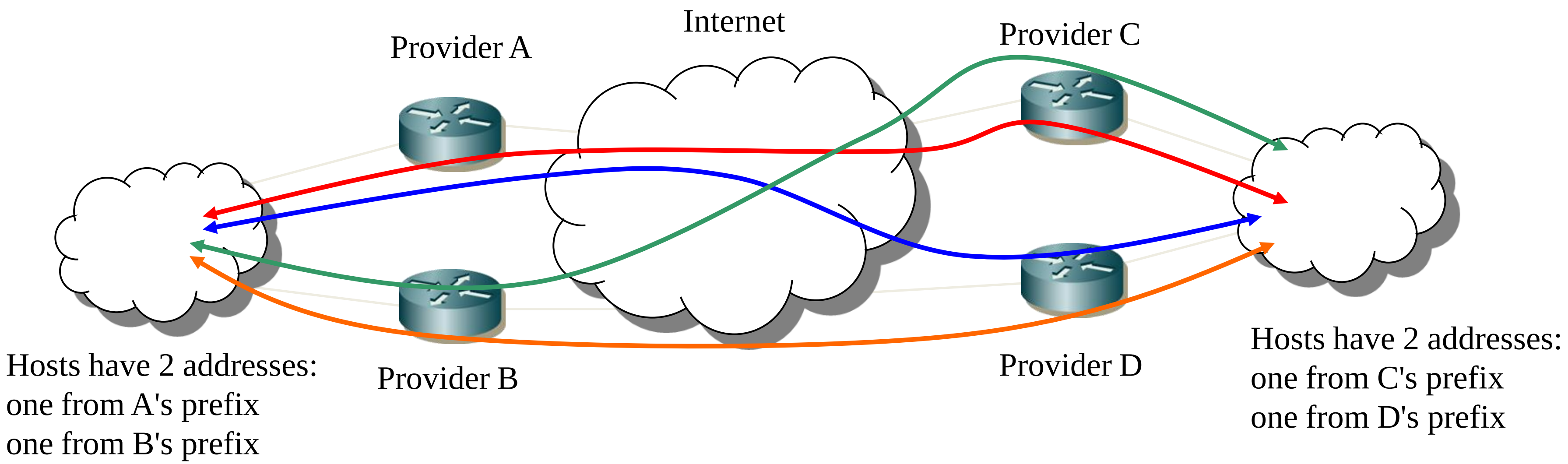
LISP: Locator ID Separation Protocol (2/2)

- LISP uses Ingress Tunnel Routers (ITR) and Egress Tunnel Routers (ETR):
  - Packet is sent by a host using EIDs, it travels until an ITR
  - At ITR EID-to-RLOC mapping is performed. The ITR adds an IP header that uses RLOCs
  - When packets reaches the ETR, the outer header is removed
- Routers forward packets based on IP addresses the IP address can either be an EID or an RLOC



SHIM6: Site Multi-homing by IPv6 Intermediation

- A third solution was to identify a group of IP addresses by a group-ID and use it as communication endpoint
- SHIM6 protocol is used for multi-homing in IPv6
  - The first address used to setup a communication is used as the session ID for the duration of the communication (ULID = Upper-Layer IDentifier)
  - If any other addresses from the group is used, the ULIDs are preserved



HIP vs LISP vs SHIM6

- HIP:
  - clean approach, uses 2 orthogonal identifier schemes for 2 functions
- LISP:
  - re-uses (independently) the same identifier scheme in two separate functions
- SHIM6:
  - a hack practically, overloading of IP address semantics as session ID and network locator, limited only to multi-homing

Questions ?