

HIP Tutorial

And Mobile Access Project

Andrei Gurtov

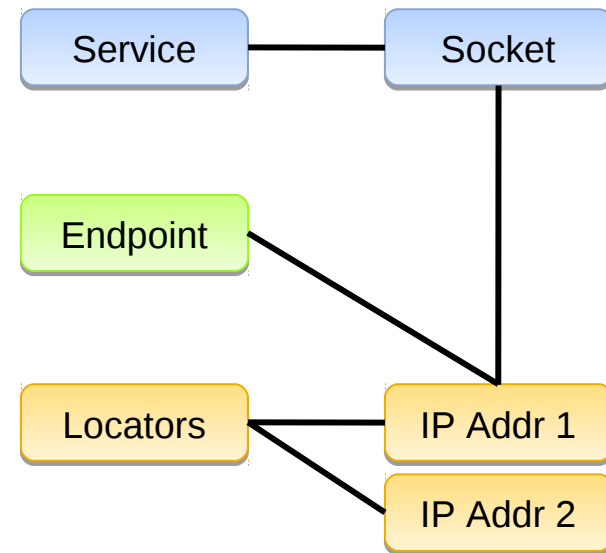
Tobias Heer

René Hummen

University of Oulu, HIIT,
RWTH Aachen University

Today's Internet Infrastructure and Protocols

- Current Internet uses the TCP/IP stack
 - Developed for non-mobile, single-homed hosts
 - Dual role of IP addresses: identify and locate end-hosts
 - Offers no security mechanisms
 - End-hosts cannot prove their identities
 - No data confidentiality and integrity protection



- Additional protocols extend specific IP functionality
 - Mobility support: Mobile IP, ...
 - Requires additional infrastructure elements (*see next slide*)
 - Security: IPsec, ...
 - Requires session setup → e.g. with IKE protocol

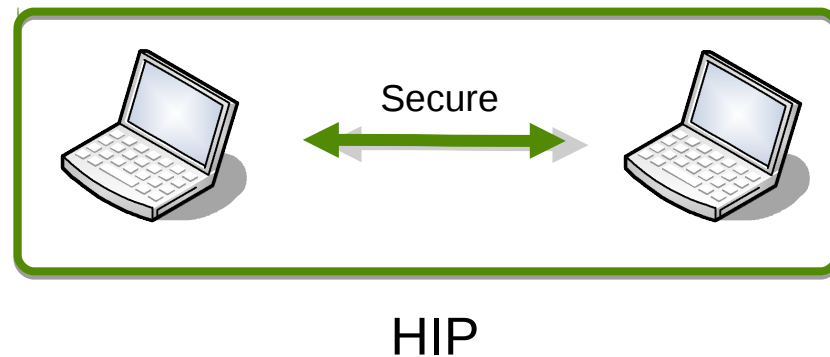
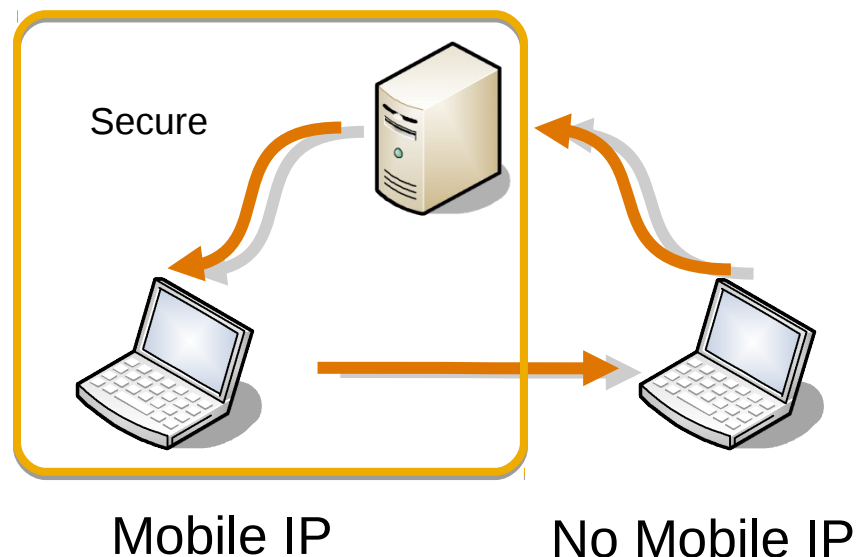
HIP vs. Mobile IP in a Nutshell

- Mobile IP

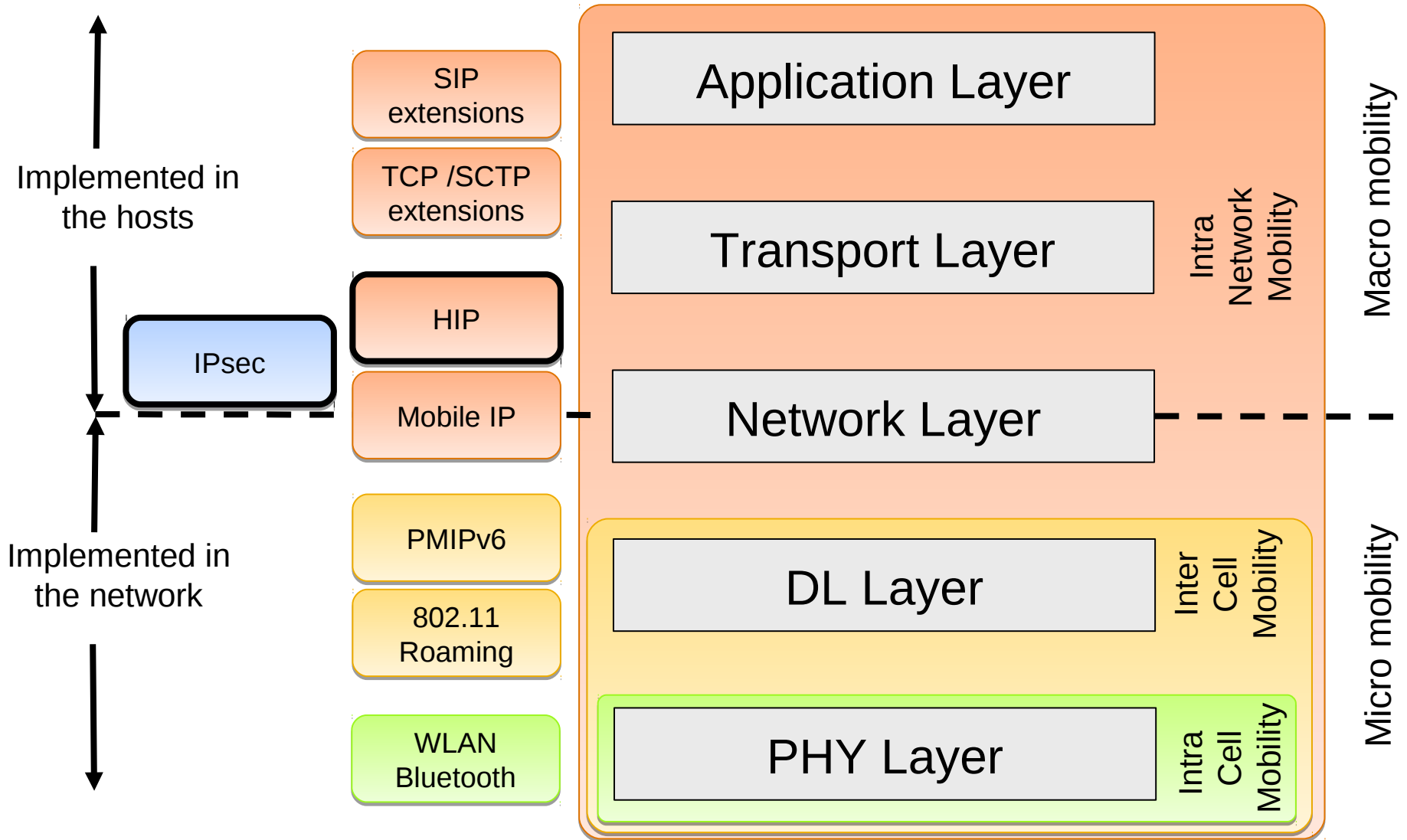
- Home agent as fixed point
- Support for un-modified correspondent node
- Indirect mobility management
- Triangular routing
- Infrastructure support (FA, HA)

- Host Identity Protocol

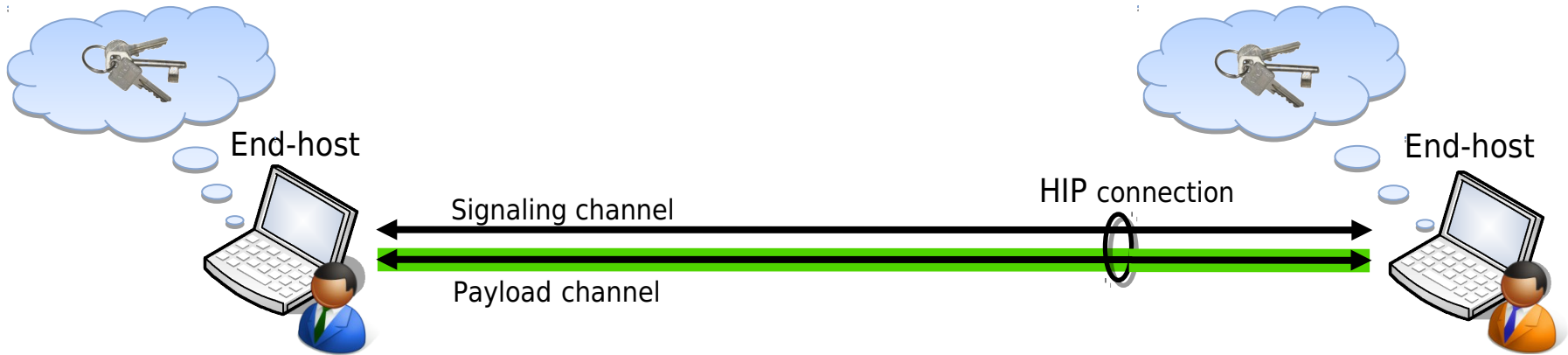
- End-to-end associations
- HIP-aware end-hosts
- Direct mobility management
- Authentication
- End-to-end security
- No infrastructure support needed (in most cases)



Mobility in the Network Stack



Host Identity Protocol (HIP)



- Signaling and key-exchange protocol
 - Separate control and payload channel
 - Allows use of security services → e.g. IPsec payload channel
 - Similar to Internet Key Exchange (IKE)
- Introduces new namespace
 - Namespace is cryptographic in nature
 - Provides support for mobility and multi homing

Cryptographic Namespace

- Host authentication is essential when supporting mobility and multi homing
 - End-hosts have to verify they still talk to the same peer
 - State changes at middleboxes may be required
 - Self-generated public and private key-pair provides the host identity (HI) in HIP
 - RSA by default, DSA also supported in HIP specification
 - Length of the public key - 512, 1024 or 2048 bits
 - Abstraction required for use in network stack due to large and variable size of the public key
- Two additional forms of host identities: HIT and LSI

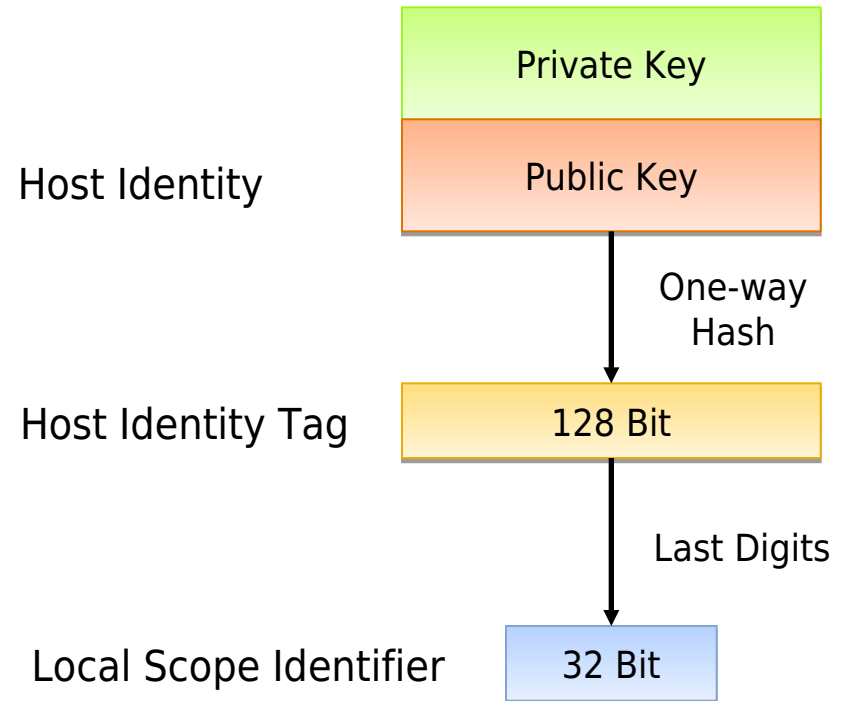
Globally Unique and Locally Unique Identifiers

- Host Identity Tag (HIT)

- Compatible with IPv6 address
- Statistically unique
- Probability of collisions is negligible

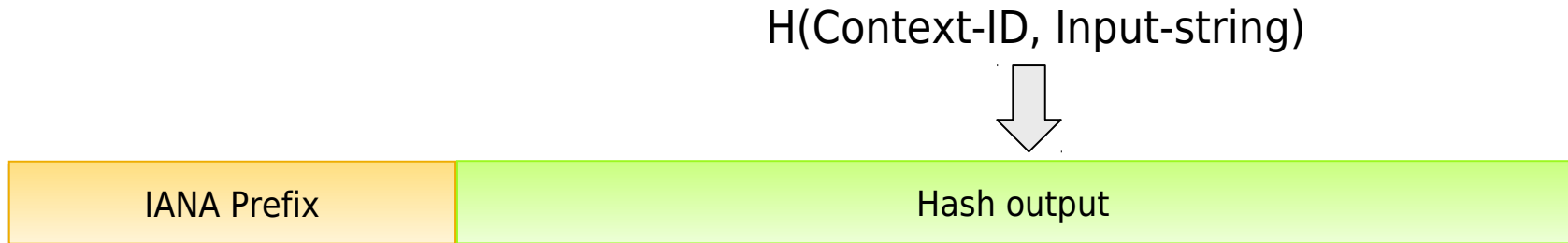
- Local Scope Identity (LSI)

- Compatible with IPv4 address
- Probability of collisions is significant
- Restricted to local scope



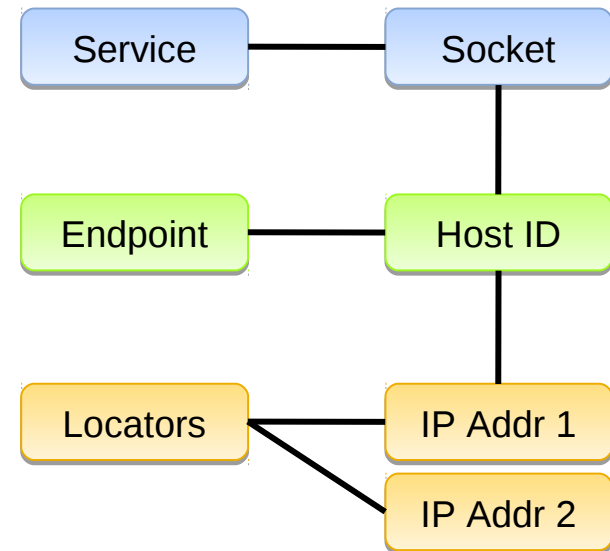
Computation of a HIT

- HIT generation follows the Overlay Routable Cryptographic Hash ID (ORCHID) method
- Components of a HIT
 - Not routable IPv6 prefix assigned by IANA (2001:0010::/28)
 - 100-bit string extracted from SHA1 hash over 128-bit context ID and input string
 - Context ID – randomly chosen value for HIP
 - Input string must be statistically unique (here: public key)

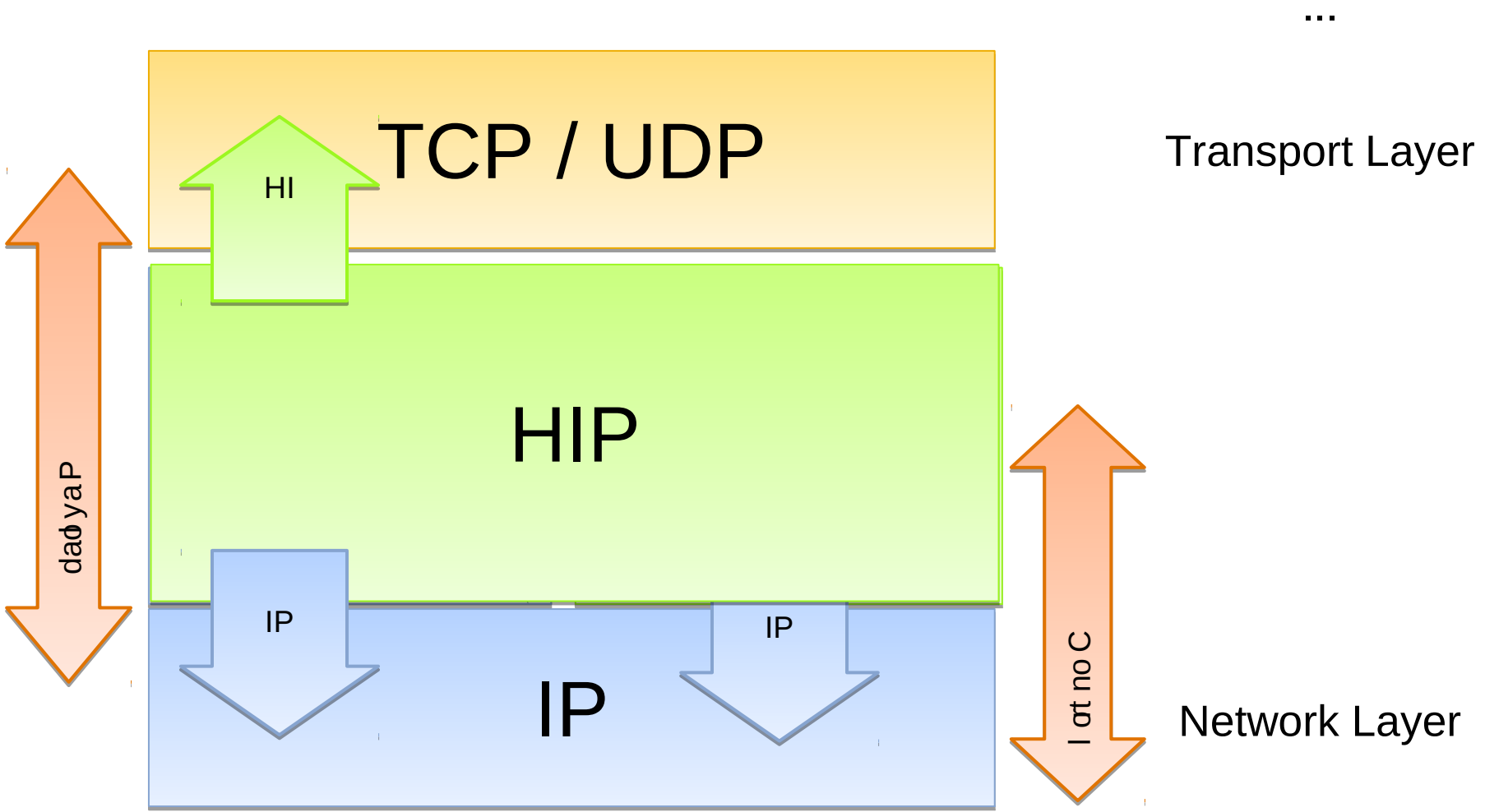


Identifier / Locator split

- Major problem in the original Internet architecture:
 - Tight coupling between networking and transport layers (e.g. TCP checksum calculation)
 - Mobility breaks transport layer connections
- Separation of location and identity of networked hosts
 - HIP replaces role of IP as identifier
 - IPv4 and IPv6 run underneath HIP
 - Transport protocols bind to His
- Benefit
 - Applications see stable identity instead of a locator
 - Routing decisions still based on locator
 - No changes to core infrastructure required



HIP in the Communication Stack

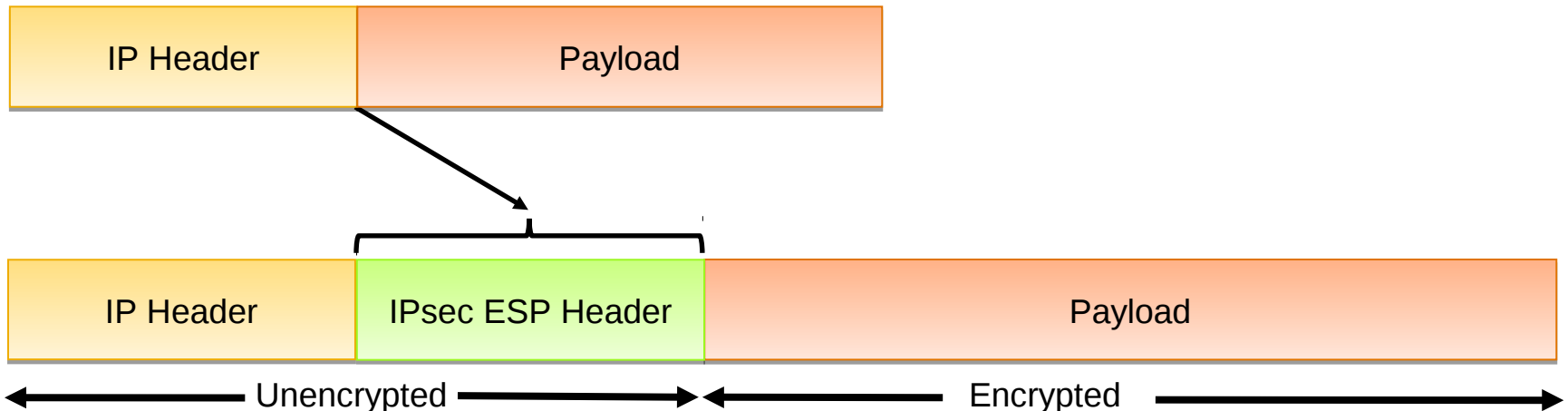


IPsec Internet Protocol Security (RFC 2401 / RFC 4301)

- Protection for IP packets
 - Unreliable, out-of order, ...
- 2 different header types
 - Encapsulated Security Payload (ESP)
 - Per-packet encryption (AES) and/or integrity protection (HMAC)
 - Authentication Header (AH)
 - HMAC – Keyed Hashed Message Authentication Code
- 2 different modes (*see next slides*)
- Replay protection
 - Replay window – restricts packet processing of packets arriving late
- Specifies no key exchange protocol for association setup

IPsec Transport Mode with ESP

- Standard IPsec Mode
 - Used for end-to-end payload confidentiality and integrity protection
 - Creates secure tunnel between two hosts
- IPsec ESP header information
 - Security Parameter Index (SPI) – required for multiplexing
 - Sequence number for replay protection
- Payload encrypted and covered by HMAC



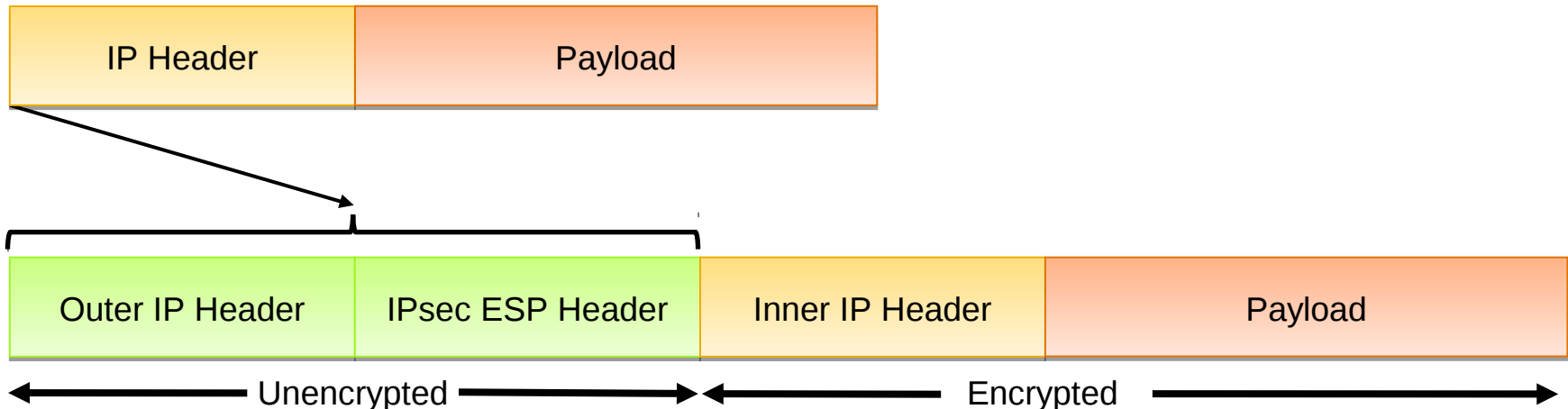
IPsec Tunnel Mode with ESP

- Standard IPsec Mode

- Originally used to securely connect entire networks
- IPsec connection between gateways
- Possible to use in combination with HIP (end-to-end)

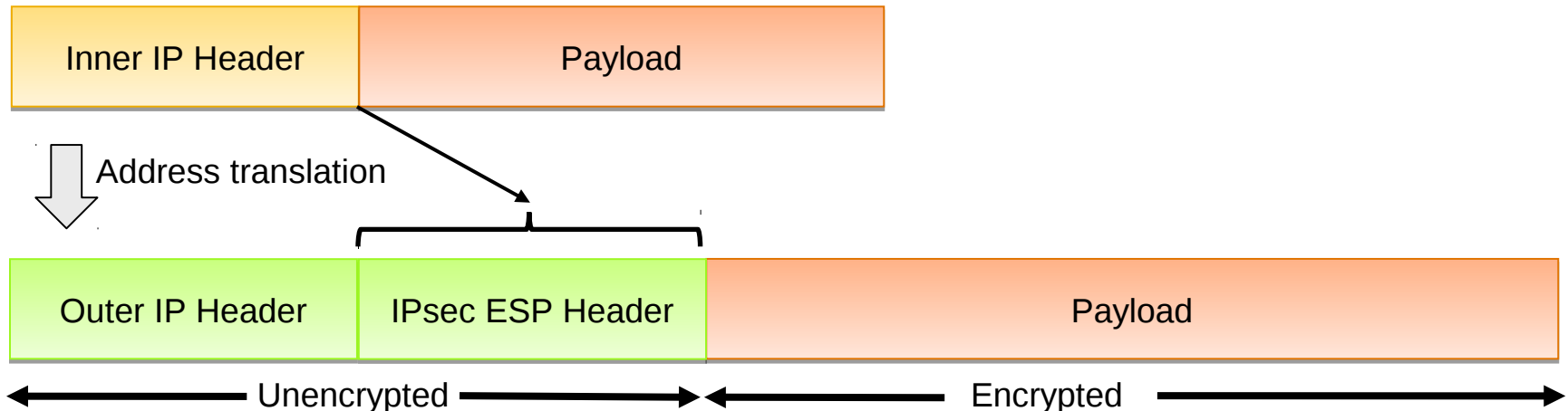
- Packet contains inner and outer header

- Inner header: addresses of communicating end-hosts
- Outer header: addresses of tunnel end-points



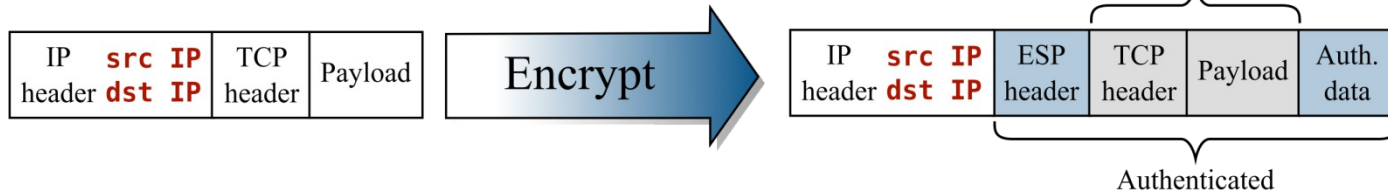
IPsec BEET mode with ESP

- New IPsec mode introduced with HIP
 - Signaling of SPIs and stable host IDs affords header compression
 - Syntactics of transport mode and semantics of tunnel mode
 - HIP namespace denotes addresses of local network
- Address translation defined by HIP
 - Inner header: HIP host ID
 - Outer header: routable IP address → visible on the wire

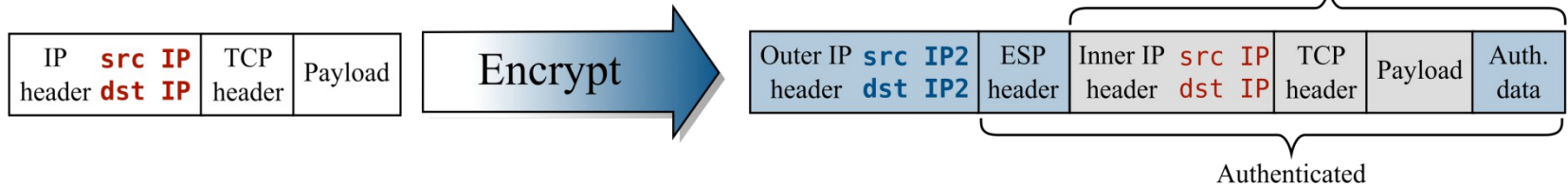


IPsec Modes with ESP Header - Overview

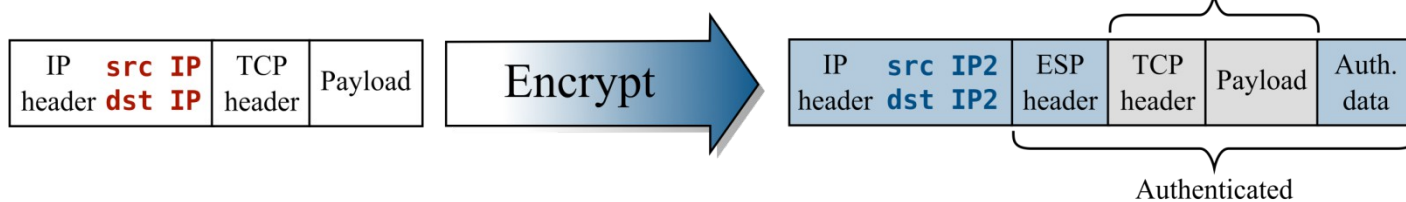
ESP transport mode



ESP tunnel mode



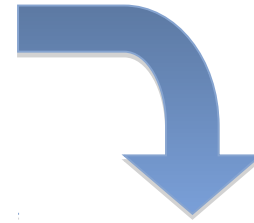
ESP Bound End to End Tunnel (BEET) mode



Lifecycle of a HIP Association

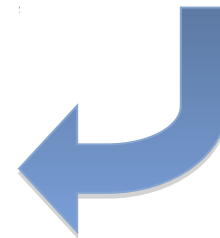
Resolve HI to IP

- DNS
- DHT
- RVS



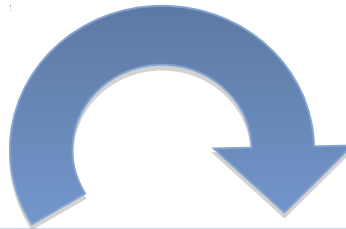
HIP Base EXchange

- Mutual authentication
- Generate shared secret
- Set up IPsec



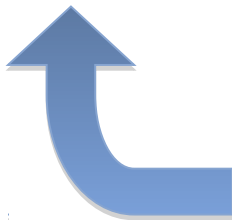
HIP Update

- Authentication
- Modify association
- New IP address



Close association

- Authentication
- Delete state

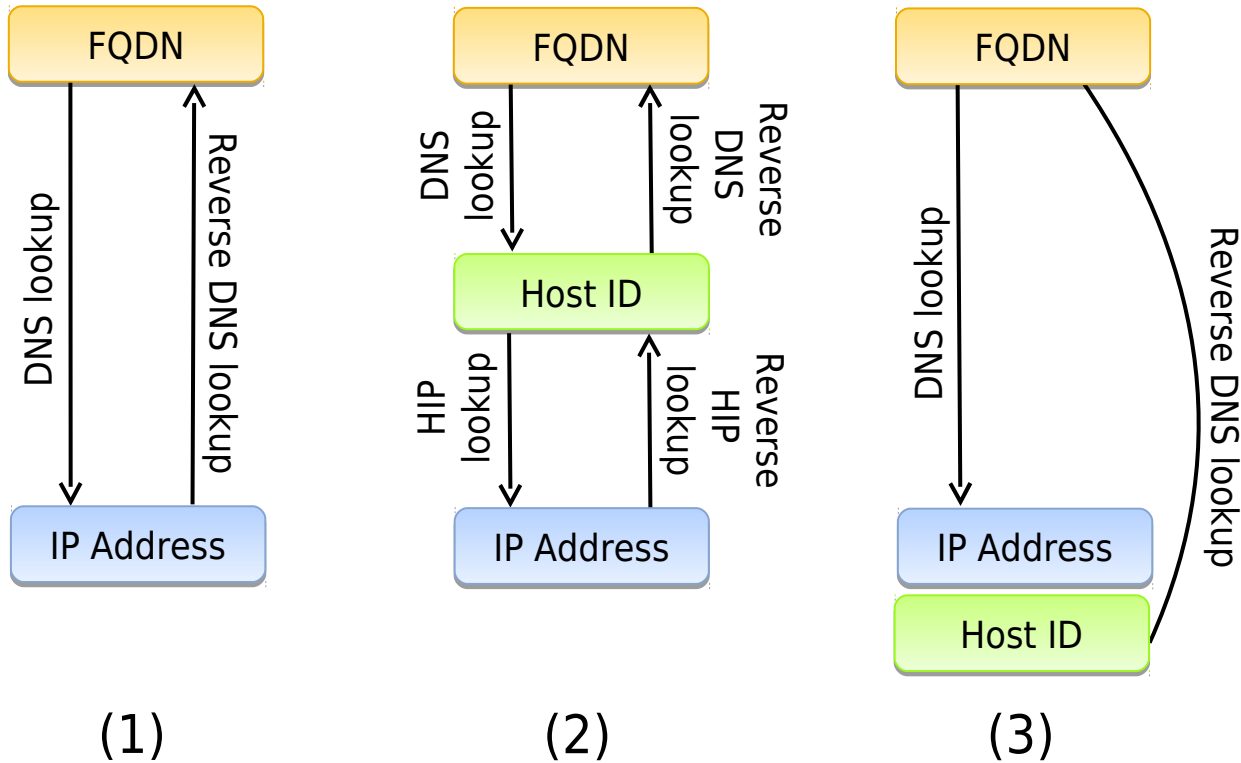


DNS Extension

Why name resolution?

- 3 different name spaces
 - Users require some kind of human-readable representation
 - With HIP, applications work on HIs or HITs
 - HIP layer needs knowledge about HI \leftrightarrow IP mappings for address translation step in the stack
- Different a priori knowledge before lookup
 - Domain name or HI/HIT
 - IP \rightarrow opportunistic mode (not covered here)
- Possible lookup architectures
 - Usage of overlays (e.g. Distributed Hash Tables (DHT))
 - Integration with DNS

Name resolution mechanisms



(1) DNS resolution in the current Internet

(2) Logical resolution for HIP

(3) Resolution proposed in HIP DNS extension

Name resolution mechanisms (cont.)

- FQDN resolution in the current Internet

- DNS resolves a FQDN to a set of registered IP addresses
- Reverse DNS lookup maps an IP to the a corresponding FQDN
- IP address of a local DNS is pre-configured or obtained from DHCP

- Logical name resolution for HIP

- HIP splits the resolution process into two logical parts
 - FQDN is resolved to a set of HIs
 - Each HI can be resolved to a set of IPs
- HI resolution can be implemented using DNS lookup
- Reverse DNS lookup maps an HI to one of the FQDN
- Reverse HIP lookup maps an IP to one of the registered HIs

HIP Name Resolution

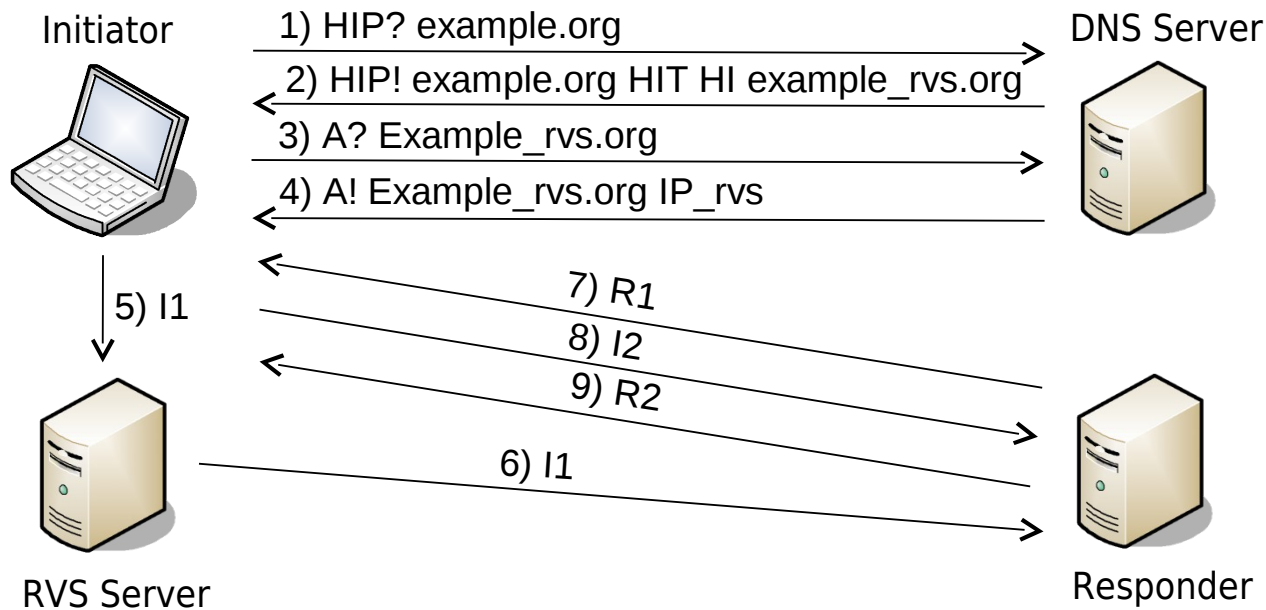
- Resolution proposed in HIP DNS extension
 - Two logical steps are combined into one DNS lookup
 - DNS lookup returns both a public HI and a set of IP addresses
 - HI is stored in a new DNS Resource Record in a HIP-aware DNS
 - Legacy DNS support
 - HI is stored in form of a HIT in AAAA field
 - Solution benefits from simplicity and existing infrastructure
- Issues with approach of HIP DNS extension
 - No support for direct communication between unnamed hosts
 - Dependency on upgrade to DNSSEC for secure HIP connection establishment
 - Protection against spoofing attacks

- A mobile host may often change its IP address
 - Mobile host can notify connected hosts about IP address changes
 - Problem: Hosts cannot initiate a new HIP connection with the mobile host
- Frequent IP address changes are a problem for DNS
 - Resource Records might be cached in DNS or locally
 - Changes take a long time to propagate
 - Propagation is inefficient in case of rapid mobility
 - Hard to update DNS information efficiently and fast enough
- Solution: mobile host stores the IP address of its rendezvous server (RVS) in the DNS
 - IP addresses of mobile host only known to RVS

Rendezvous Server

- RVS provides host with a stable IP address
 - Comparable to Home Agent of Mobile IP
- Integration of RVS with name resolution
 - HIP RR from DNS stores pointer to responsible RVS
 - Mobile host registers with RVS (see reading material)
 - Mobile host updates its registration with RVS after mobility event
- 2-step address lookup
 - HIP Initiator first queries a HIP RR from DNS
 - Returns IP address of the Responder or its RVS
 - If RVS, additional address lookup

HIP Name Resolution with RVS



1-4) Initiator obtains IP address of RVS from DNS

5) HIP BEX initiated with the mobile host through the RVS

6) RVS relays the I1 packet to the mobile host

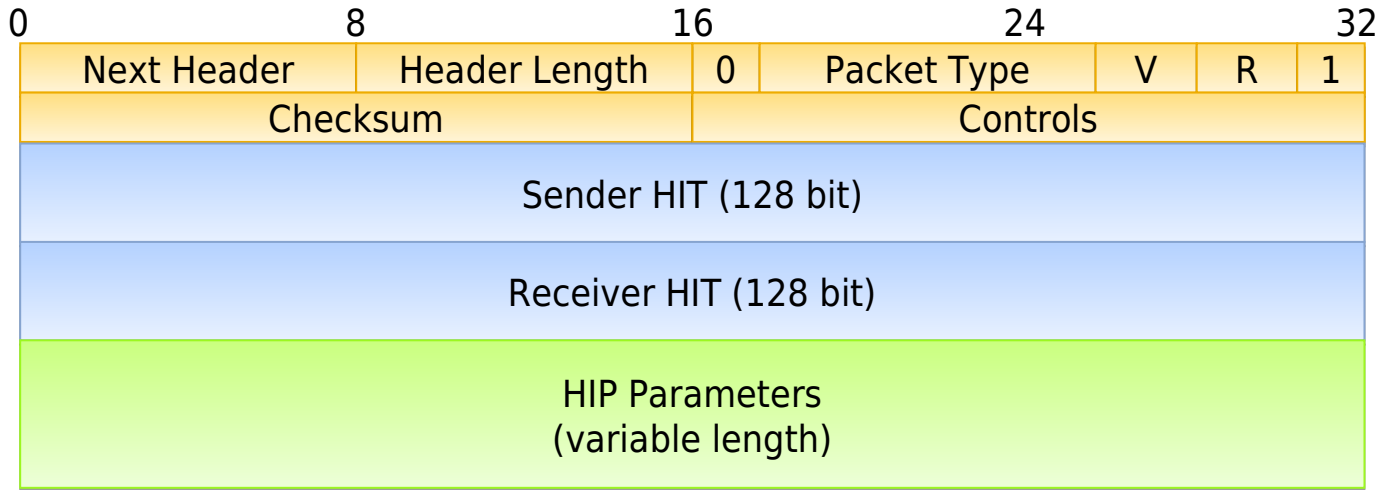
7-9) HIP BEX continues between the two end-hosts

Base Protocol

HIP Control Packets

- End-hosts transmit control packets on signaling channel
- HIP control packets provide...
 - Current HIT-IP mappings
 - Verifiable identities of communicating end-hosts
 - Keying material required for setup of security mechanisms
 - Replay and DoS protection
- Definition of HIP control packets
 - Added behind IPv4 header or as IPv6 extension header
 - HIP protocol number is 139
 - Common HIP header defined for all HIP control messages

HIP Control Message Format



- Next Header: set to “no next header”
- Header Length: length of the HIP Header and HIP parameters in 8-byte units
 - excluding the first 8 bytes
- Packet Type: indicates the HIP packet type
- HIP Version: current version is 1

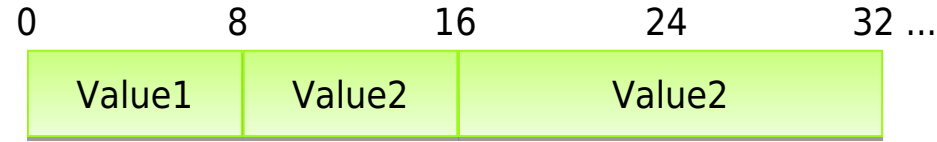
HIP Control Message Format (cont.)

- Checksum: also covers IP addresses
 - Calculated over pseudo-header
- Controls: convey information about the structure of the packet and capabilities of the host
- HIT fields: 128 bit address fields
 - Syntactically compatible with IPv6
- HIP Parameters: additional signaling parameters
 - Maximum length of the HIP Parameters field is 2008 bytes

HIP Parameter Format

- Classical parameter formats

- Hard-coded control packet layout
- Fixed parameter position and lengths
- Not flexible
- Space-efficient

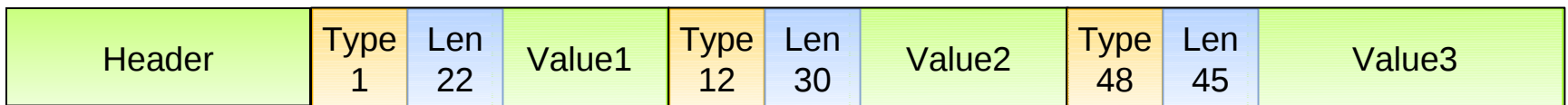


Protocol definition

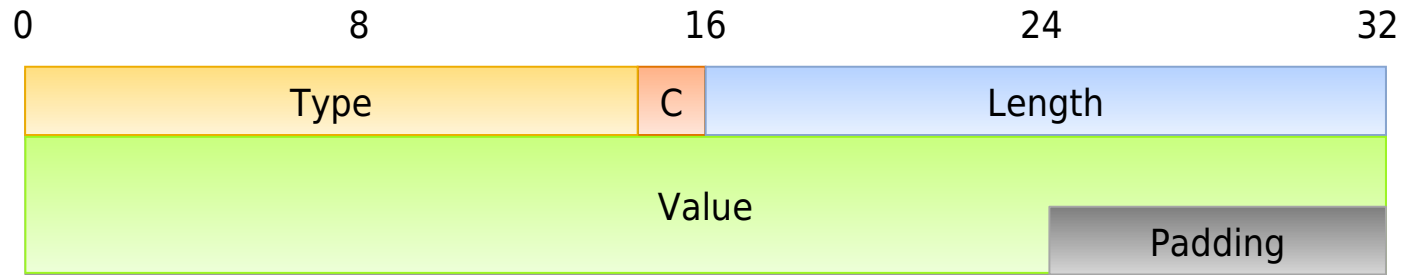
Bit position 0-7: Type 1
Bit position 8-15: Type 2
Bit position 16-31: Type 3

- TLV coding (used in HIP)

- Variable contents of a control packet
- Easy to extend
- Modular and flexible
- Less space-efficient



HIP Parameter Format

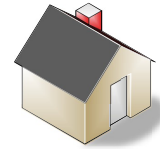


- HIP parameters are encoded in TLV format
- Type value denotes order of parameters in the packet
 - Types form an increasing order
 - Critical bit: parameter must be recognized by the recipient
- Length indicates the length of the value field
 - multiple of 8 bytes
- Value can be any kind of data structure
- Padding bytes must be zeroed

HIP Handshake



Mobile Host
(Initiator)



Remote Host
(Responder)

I1: Handshake request



R1: Host Identity, {Diffie-Hellman}, Sig



Shared key

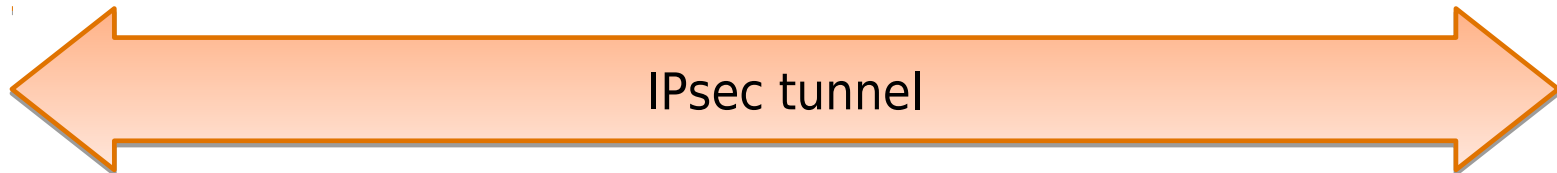
I2: Host Identity, {Diffie-Hellman}, IPsec Info, HMAC, Sig



R2: IPsec Info, HMAC, Sig



Shared key



IPsec tunnel

I1 Packet - Initiating the Handshake

- Starts the HIP Base EXchange
- Basic hello message
 - Part of return-routability test
- Contains the Initiator and the Responder HITs
 - HITs can be obtained e.g. from DNS (see *DNS Extension*)
- Consists of minimal HIP header without HIP parameters
 - Must not trigger CPU-intensive operations at Responder

R1 Packet - Starting Session State Setup

- Pre-created response to I1 packet
 - Signature only covers immutable fields
 - Fields such as HIT of the Initiator are filled in on reply
 - Pre-created templates used for several handshakes
 - Protection against DoS attacks
- Contains challenge for the Initiator
 - Part of return-routability test
 - Address verification for sender
- Consists of many parameters (*see next slides*)

R1 Parameters - Puzzle

- Problem: Responder may be subject to I1 flooding (DoS attack)
- Idea: Allow Responder to cause CPU load on Initiator before Responder performs CPU-intensive operations and builds up state
- Cryptographic puzzle parameter
 - Task for Initiator to find a value that produces zeros when applied to a SHA-1 hash function
- Puzzle difficulty
 - The required number of zeros
 - Determines number of iterations for Initiator
 - Dynamic adjustment of the puzzle difficulty possible (in cases of attacks)
- Fast Verification for the Responder: simple hash computation
- Puzzle lifetime
 - The number of seconds the Initiator may use for finding a solution

```
H(seed,1) → 1FA3DE04... != 00000...  
H(seed,2) → D02FFB41... != 00000...  
H(seed,3) → FFE32889... != 00000...  
.....  
H(seed,223588) == 000001FA... !  
Solution: 223588
```

R1 Parameters - Diffie Hellman Key Exchange

- Goal: Establish a shared secret between Initiator and Responder
 - Used to derive share key for HMAC and IPsec session key
- Diffie Hellman key exchange
 - Negotiation of a mutual secret keying material
 - Responder sends $A = g^x \text{ mod } p$ to Initiator
 - Initiator returns $B = g^y \text{ mod } p$
 - Shared secret: $A^y \text{ mod } p = B^x \text{ mod } p = g^{xy} \text{ mod } p$
 - Modular differentiation is computationally hard
 - Attacker can not compute x , y or $g^{xy} \text{ mod } p$

p : large prime number, g : generator for sub-group in $GF(p)$

R1 Parameters - Diffie Hellman Key exchange (cont.)

- Diffie-Hellman parameter
 - contains the public key (g^x or g^y) from a certain group
 - 384-bit and 1536-bit Modular Exponential (MODP) groups
 - Groups defined in RFC4753

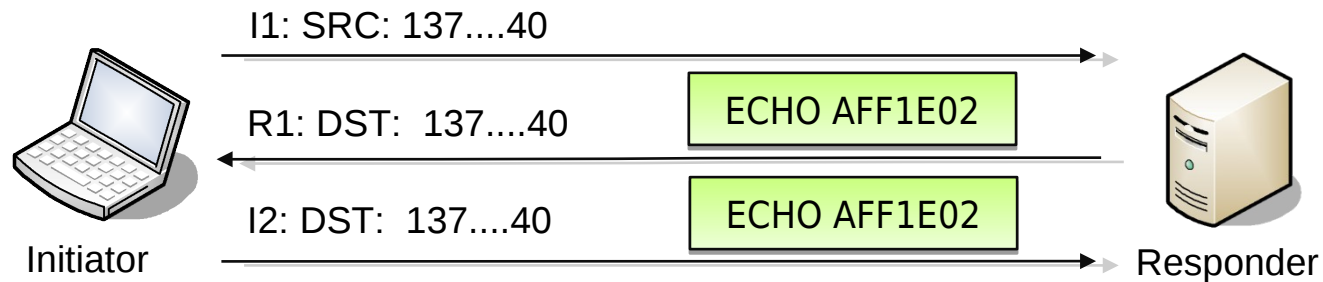
R1 Parameters (cont.)

- HIP TRANSFORM parameter
 - A list of cryptographic algorithms supported by the Responder
 - Up to six transform identifiers (Suite-IDs)
 - Used for negotiating cryptographic algorithms for HIP control channel
- HOST ID parameter
 - Public key of the Responder (required for authentication)
 - Optionally contains domain identifier (Fully Qualified Domain Name or Network Access Identifier)
- HIP SIGNATURE
 - Generated with the private key matching the HOST ID
 - RSA or DSA algorithm is used to generate the signature

R1 Parameters (cont.)

● ECHO REQUEST

- Challenge cookie used in the return routability test
- Validates Initiator's source address
- Ensures freshness of response and allows to delay state creation
- Initiator has to echo cookie in I2



● R1 COUNTER parameter in R1

- Indicates the current generation of valid puzzles
- Responder should increment counter periodically

- ESP INFO

- Index to keying material
- SPI number for inbound connection

I2 Packet – The Second Initiator Packet

- Consists of similar fields as the R1 message
- Puzzle parameter is replaced by the puzzle solution
 - Indication that the Initiator has conducted effort to solve puzzle
- Additional parameters
 - HMAC parameter
 - Keyed hash over packet content
 - Computationally inexpensive sender authentication and integrity protection
 - Unmodified R1 COUNTER parameter
 - Index to the valid generation of puzzles
 - ECHO REPLY parameter
 - Response to the address verification test (return routability test)

R2 Packet - Conclusion of the Handshake

- Carries ESP INFO parameter
 - Index to keying material
 - SPI number for inbound connection
- HIP control packet protected by HMAC and signature
 - Responder proves correct derivation of the keying material
- HIP connection is fully established when processing of the R2 packet is complete
 - IPsec tunnel ready for transmission of payload data

Other HIP Control Packets

- UPDATE packet

- Handling IPsec rekeying, mobility, and multi homing
- Contains mandatory HMAC and HIP SIGNATURE parameters
- Provides SEQ/ACK parameter to ensure reliable transmissions
- Presence of SEQ indicates that receiver must ACK the UPDATE

- NOTIFY packet

- Informs the peer about protocol errors or negotiation failure
 - errors types (INVALID SYNTAX, CHECKSUM FAILED, etc.)
- The HIP host should not base state changes on this message type
- Contains NOTIFY parameters, HI, and HIP SIGNATURE

Other HIP Control Packets (cont.)

- CLOSE packet

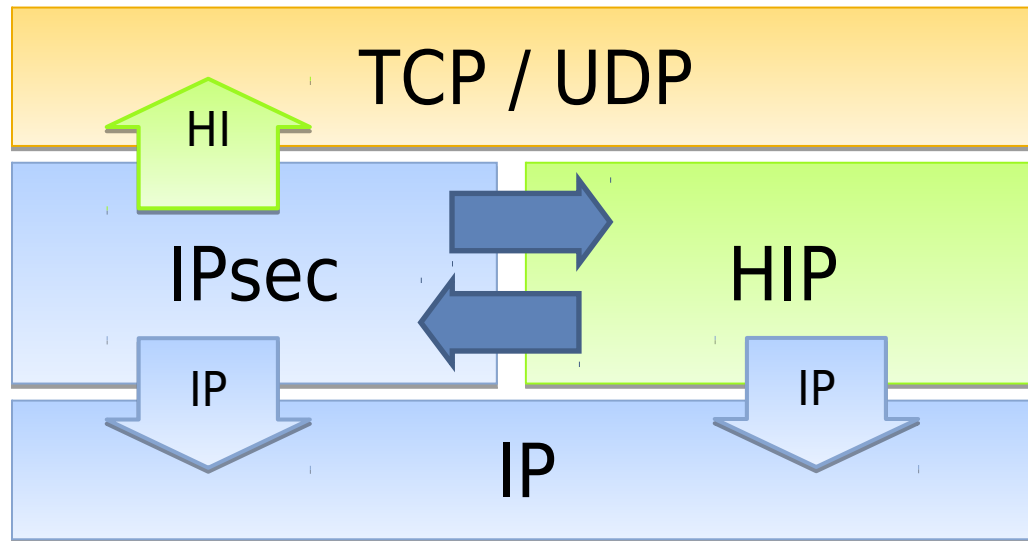
- Terminates HIP associations
- Contains ECHO REQUEST, HMAC, HIP SIGNATURE parameters
- Authenticates state removal (End-systems and middleboxes)
- Acknowledged by a CLOSE ACK packet

- HIP-specific ICMP messages

- Indicate unsupported protocol version, invalid puzzle solution, non-existing HIP association, or malformed parameter
- Rate limited for protection against reflection attacks
- Not covered by cryptographic mechanisms
- Caution about trusting content of ICMP messages

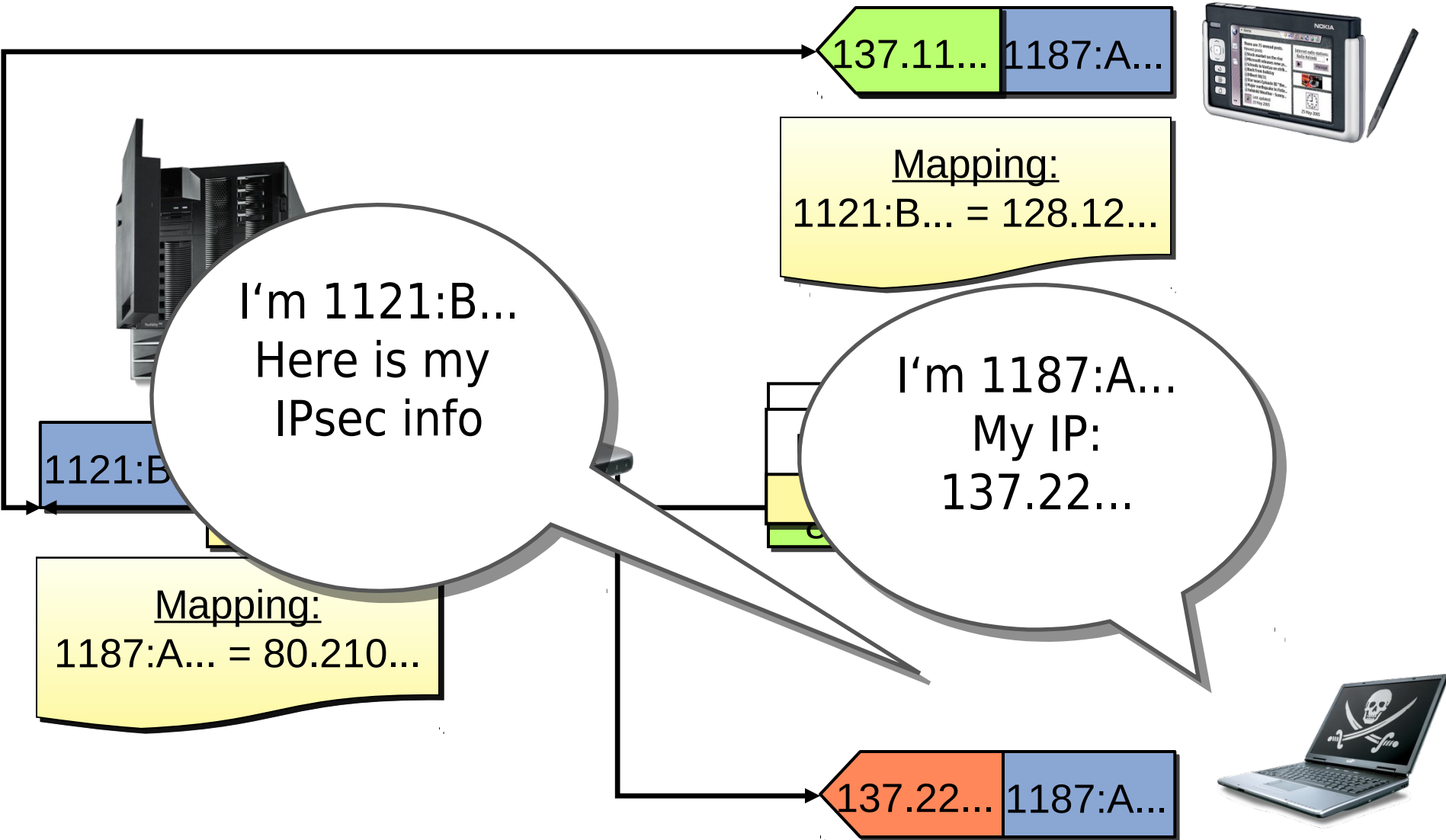
Mobility

Recap: Network Stack supporting HIP



- Applications bound to (stable) HITs
 - New points of network attachment do not influence this binding
 - Allows to handle mobility transparently for transport layer
- HIP layer maps HITs to routable IPs
- Mobility event requires update of mapping information at peer host

HIP Mobility in a Nutshell



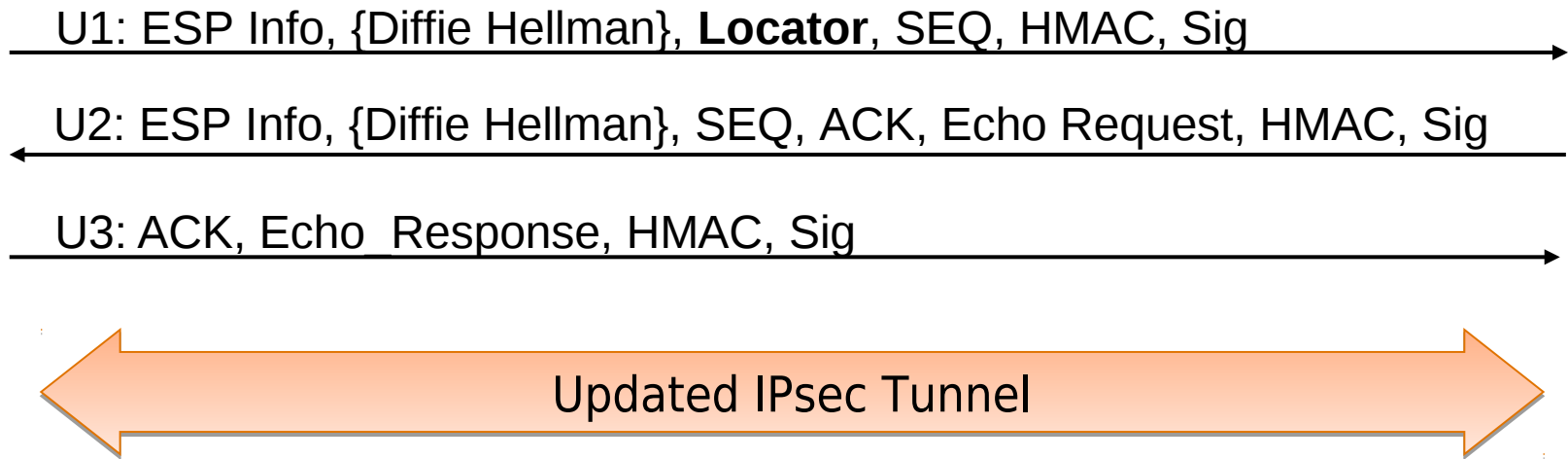
Mobility signaling



Mobile Host



Server



- Similar message exchange for...
 - IPsec rekeying
 - Multi homing (*see next chapter*)

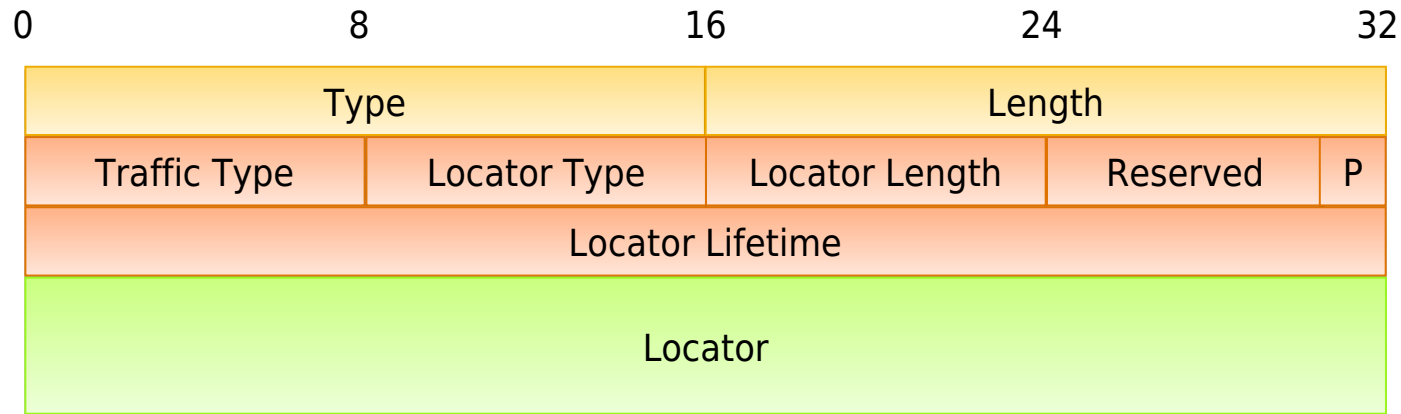
Simple Mobility Scenario

- Mobility with a single IPsec SA pair without rekeying
- First UPDATE packet
 - Mobile host obtains new IP address → sends UPDATE message to the peer
 - New address advertised in the LOCATOR parameter
 - Old SPI and new SPI fields in the ESP Info parameter are set to the current outbound SPI → no modifications to IPsec SA
 - Contains a sequence number for replay protection and retransmission support

Locator Parameter

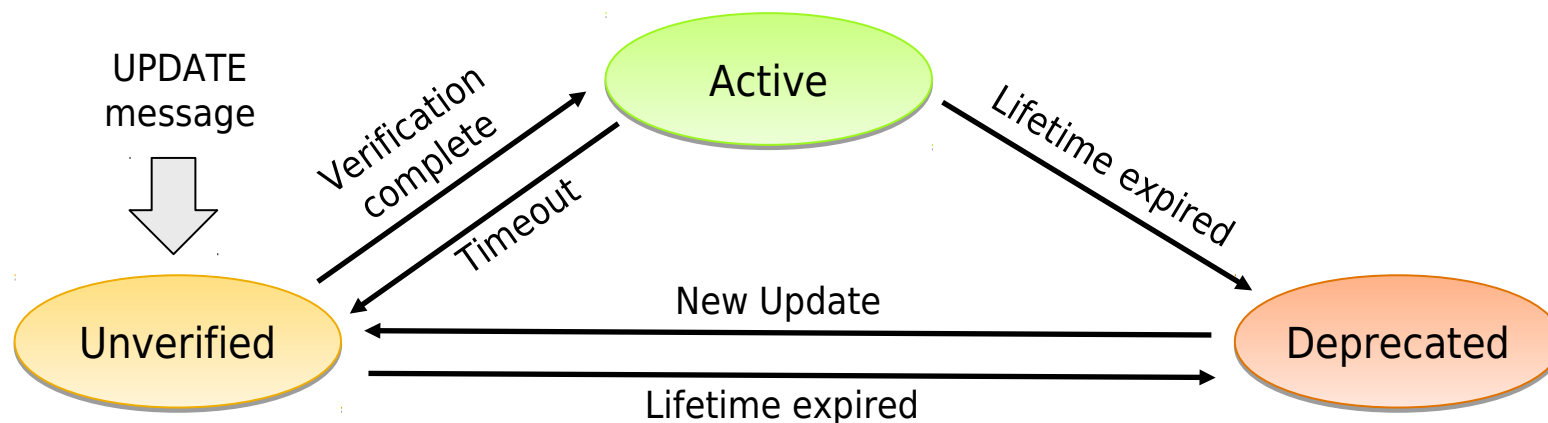
- Locator is a generalization of a network address
 - Specifies a point-of-attachment to the network
 - IP addresses alone may not be sufficient to describe how packets should be handled below HIP
 - May also include additional end-to-end tunneling or per-host demultiplexing context (e.g. SPI)
- Locator is a HIP critical parameter
- Zero locator field deprecates all existing host addresses

Locator Parameter Format



- Standard HIP parameter type and length fields
- Consists of zero or more locator sub-parameters
 - Traffic Type – locator available for HIP control packets, HIP data packets or both
 - Locator Type – semantics of the locator field (e.g. only IP)
 - P – set to 1 if the locator is preferred for that traffic type
 - Locator Lifetime – validity period

Three States of a Locator



- UNVERIFIED - address reachability not tested yet
- ACTIVE - address is valid and has not expired
- DEPRECATED - expired originally valid locator
- No transition from DEPRECATED to ACTIVE
 - Requires another address reachability verification

Credit-based Authentication

- Mobility allows for redirection-based flooding attacks
 - Attacker forges locator address in UPDATE messages
 - Then initiates bandwidth-intensive message exchange
 - Prevention through credit-based authentication (CBA) mechanism when sending data to an address in UNVERIFIED state
 - CBA approach is not HIP-specific (also Mobile IP)
- Idea: limit the rate of data transmissions
 - Host maintains a credit for each peer
 - Credit = number of bytes received from IP of the peer
 - Only send packet if credit suffices
 - Credit aging decreases the credit over time
 - Prevention of credit build-up

Simple Mobility Scenario (cont.)

- Second UPDATE packet

- Allows peer to verify host reachability at the claimed new address
- Old SPI and New SPI in ESP Info parameter are set to the current outbound SPI
- Includes ACK for the first UPDATE → handling of retransmissions
- Contains sequence number → replay protection and retransmission support
- ECHO Request – challenge for the mobile host

Simple Mobility Scenario (cont.)

- Third UPDATE packet
 - Echo Response proves that peer receives data at the new address
 - Acknowledges the second UPDATE message
 - Message itself is un-acknowledged
- Mobility with a single SA pair with rekeying
 - E.g. for IPsec SEQ wrap-around
 - ESP Info contains old SPI and random new SPI
 - Optional Diffie Hellman parameters in U1 and U2 packets
 - Allows to generate new keying material

Multi Homing

Multi Homing as Extension of Mobility

- Typical multi homing scenario
 - Host connected through 2 or more interfaces
- Handling of multi homing in HIP corresponds to mobility signaling
 - Multi-homed host informs peer of further addresses with additional Locator parameters in UPDATE message
 - Peer runs reachability check for all advertised locators
 - Possibility to set new preferred locator → data is sent here
- Multi-homed host can also send locator during BEX
 - IP addresses used during BEX are preferred locators by default
 - Verification of host reachability is necessary as well

Multi Homing and ESP Anti-replay Window Effects

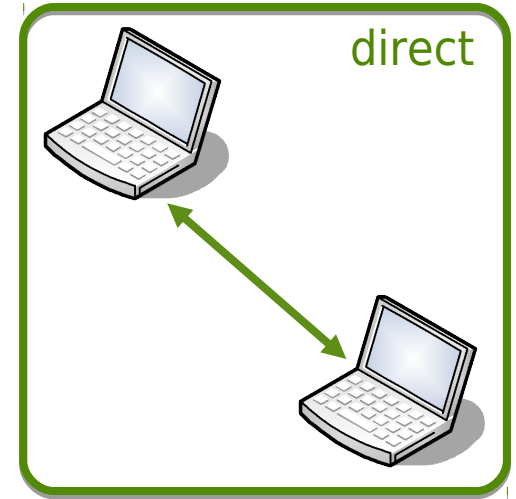
- Problem: single SA for multiple IP addresses
 - Separate interfaces → different paths latencies in the network
 - Anti-replay window limits the range of sequence number accepted by IPsec
- Recommendation: separate SA for each physical interface
 - Requirement to set up new SA when new address is advertised
 - Old SPI set to zero and new SPI set to a new value for incoming SPI
 - Session key can be derived from existing keying material
 - Peer uses the locator information and destination address of UPDATE to create the new SA

NAT Traversal

Infrastructure Elements in the Internet

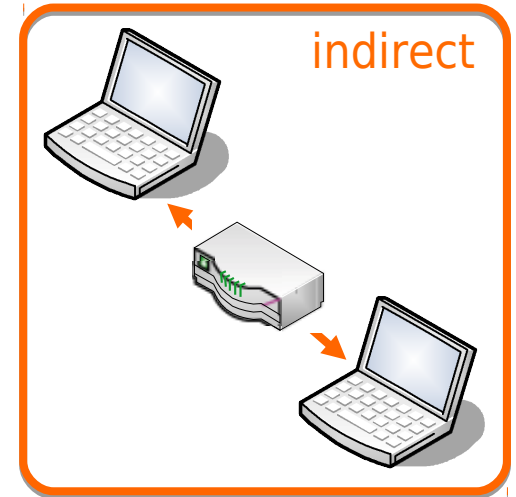
- Original Internet architecture

- Intelligence expected to be located at end-hosts
- Network is „dumb“ → best effort packet delivery
- Packet header inspection up to layer 3
- Packet not modified in transit



- Emergence of middleboxes

- Intelligence at the edges of the network
- NAT: compensation for lack of IPv4 addresses
- Firewall: prevention of attacks, access control
- QoS and Proxy: data delivery performance



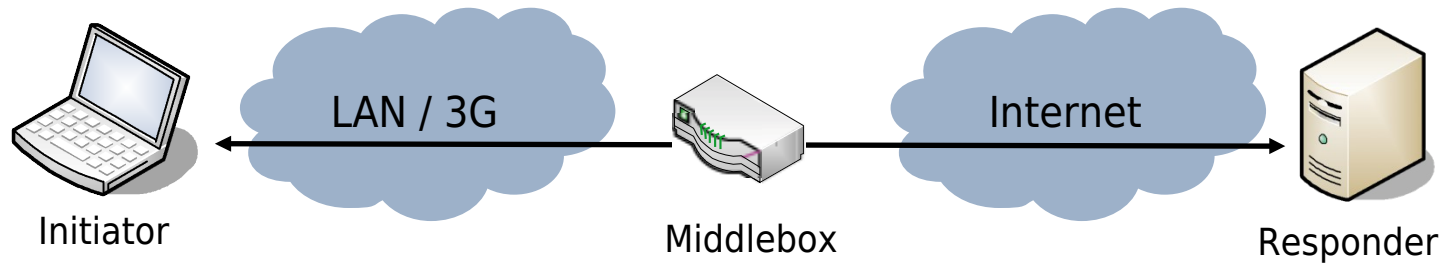
Legacy Middleboxes and HIP

- Strong (theoretical) advantage of HIP architecture
 - Ability to function without changes to existing IP routers
 - BUT: Middleboxes can affect HIP packet delivery
 - Support only limited set of protocols (often TCP and UDP over IPv4)
 - E.g. NAT requires port information
 - Restrictive firewall rules may prevent HIP traffic
- Requirement to engineer support for legacy middlebox traversal

Legacy Middleboxes and IPsec

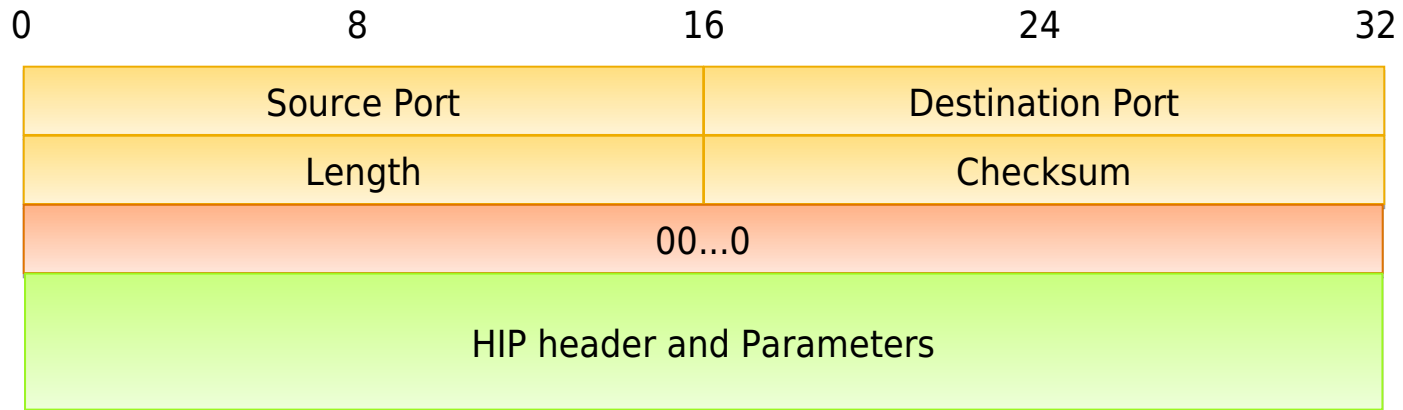
- Similar problem with IPsec
 - Does not provide port information either
- Some NATs offer VPN pass-through feature
 - Attempt to learn SPIs in both direction
 - Setup SPI \leftrightarrow IP mappings
- BUT: all on-path middleboxes need to support pass-through

HIP Strategy for Legacy Middlebox Traversal



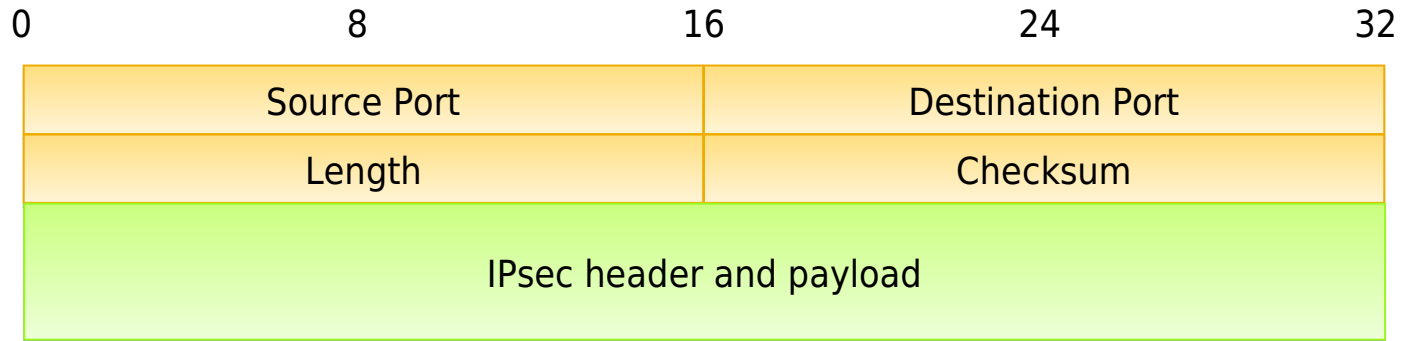
- First step – UDP encapsulation for HIP control and payload packets
 - UDP understood by majority of middleboxes (MB)
 - Port information allows MB to multiplex and demultiplex traffic
 - Sufficient if only Initiator behind NAT
- Deficiencies of UDP encapsulation
 - Header overhead
 - Does not help if Responder or both end-hosts are located behind NAT

HIP UDP encapsulation format



- Insertion of UDP header in front of HIP header
 - Destination port 50500 should be used for initial experimentation
 - Length and checksum fields are computed as usual (RFC 768)
- UDP header is followed by 32 zero bits of 0
 - Used to differentiate HIP control packets from ESP packets
- HIP checksum is not used and is set to zero

IPsec UDP encapsulation format



- Insertion of UDP header in front of IPsec header
 - Follows RFC3948
 - Checksum should be transmitted as a zero value

Strategy for Legacy Middlebox Traversal (cont.)

- NATed HIP hosts need to communicate their locators to each other
 - Locator pairs can then be tested sequentially
 - Hope that pairing allows direct connection (see next slide)
- Second step - Additional HIP infrastructure element
 - Connection establishment through publicly available HIP relay server
 - Forwards **all** HIP control messages
 - Extension of rendezvous server
 - Allows exchange of locators
 - Address of relay server included in DNS entry of Responder

Strategy for Legacy Middlebox Traversal (cont.)

- Third step – Probing for direct connection

- Use mechanisms of Interactive Connectivity Establishment (ICE)
 - Defined in RFC 3948
- Technique for NAT traversal of UDP-based media streams
- Based on peer-to-peer connectivity checks
 - Performed using the Session Traversal Utilities for NAT (STUN)
 - End-hosts test connectivity between different locators and try to discover a direct end-to-end path between them
- No direct path found → relay the traffic (TURN server)

→ HIP uses a HIP relay server for relaying control traffic and TURN server for the payload traffic

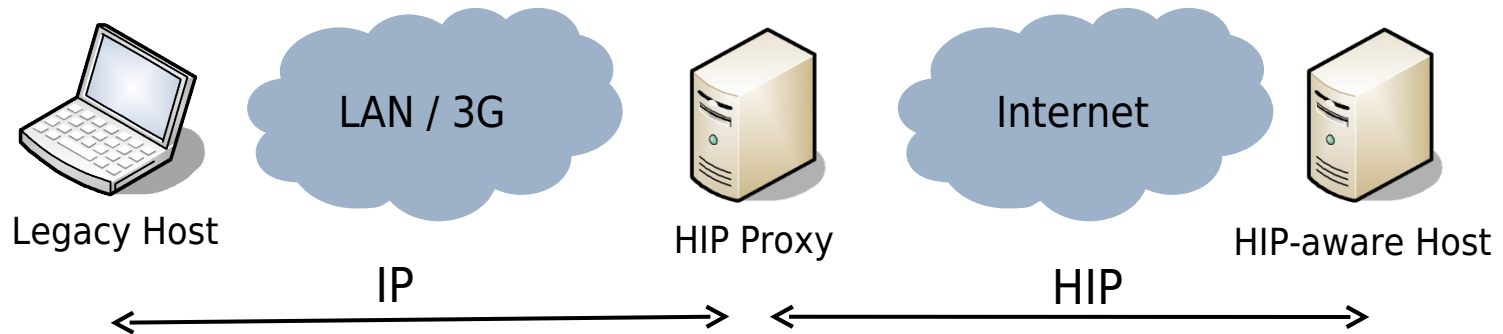
HIP Proxy

HIP-support for Legacy Clients

- HIP requires both end-hosts to support it
- HIP does not need support of core network
 - Unlike IPv6
- Additional infrastructure element to support legacy clients (HIP Proxy)
 - HIP Proxy poses as end-point of the HIP connection
 - One end-hosts of a connection can remain HIP-unaware
 - Allows incremental deployment of HIP
- HIP Proxy helps to obtain HIP benefits in 2 scenarios
 - Legacy host contacting a HIP-aware peer
 - HIP-aware host contacting a legacy peer

Legacy Contacting Host

- HIP Proxy terminates HIP-connection with HIP-aware host
 - Provides secure data transmission in the Internet
 - Still unprotected traffic between legacy host and HIP Proxy
- Setup does not add mobility and multi homing support
 - Requires upgrade to end-to-end HIP awareness



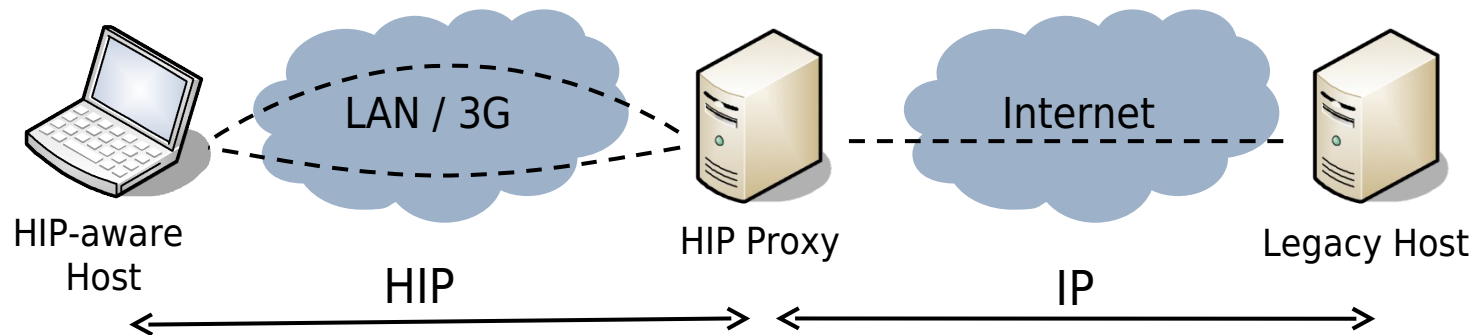
Legacy Peer Host

- Advantages of using a HIP Proxy

- Traffic protection on the path between the mobile hosts and the proxy
- Mobility and Multi homing for HIP-aware host is provided in LAN

- Example

- Mobile host uses WLAN for Internet access
- Mobile host can connect to a HIP proxy in the network
- All air traffic is encrypted
- Moving between different WLAN networks behind proxy is possible



Certificates

Certificates Exchange on HIP Control Channel

- HIP namespace builds upon public keys
 - Certificates allow to leverage this namespace further
 - Centralized control over end-hosts
 - Offline verification possible
- End-hosts send certificates in a special parameter during HIP BEX or in UPDATE messages
 - Certificate follows the Simple Public Key Infrastructure (SPKI) format

Scenario: HIP-aware Firewall

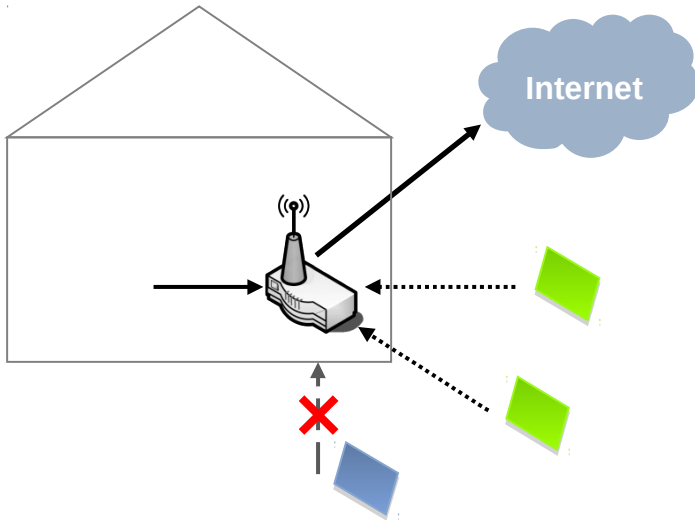
- HIP allows firewalls to permit access based on host ID
- Without certificates the firewall ACL has to include the HI or HIT of each permitted end-host
- Certificates simplify the authentication process for HIP-aware firewalls
 - Firewall trusts and stores the public key of Certificate Authority (CA)
 - Authentication of HIP hosts during BEX and UPDATE based on HI and corresponding certificate transferred on control channel
 - HIs of end-hosts can be purged from the memory after connection teardown

Certificate Lifetimes

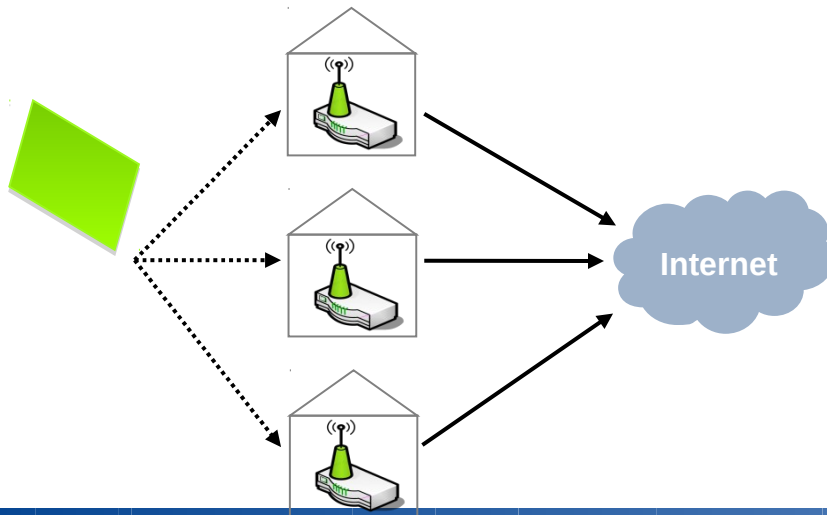
- HIs can be compromised or hosts can behave maliciously
- The lifetime of a certificate determines the validity time span
 - No further authorization when lifetime has ended
- Tradeoffs for short lifetimes
 - HIP host has to frequently apply for new certificates
 - Reduction of the need for a Certificate Revocation Lists (CRL)

PISA

Concept of Wi-Fi Sharing Communities

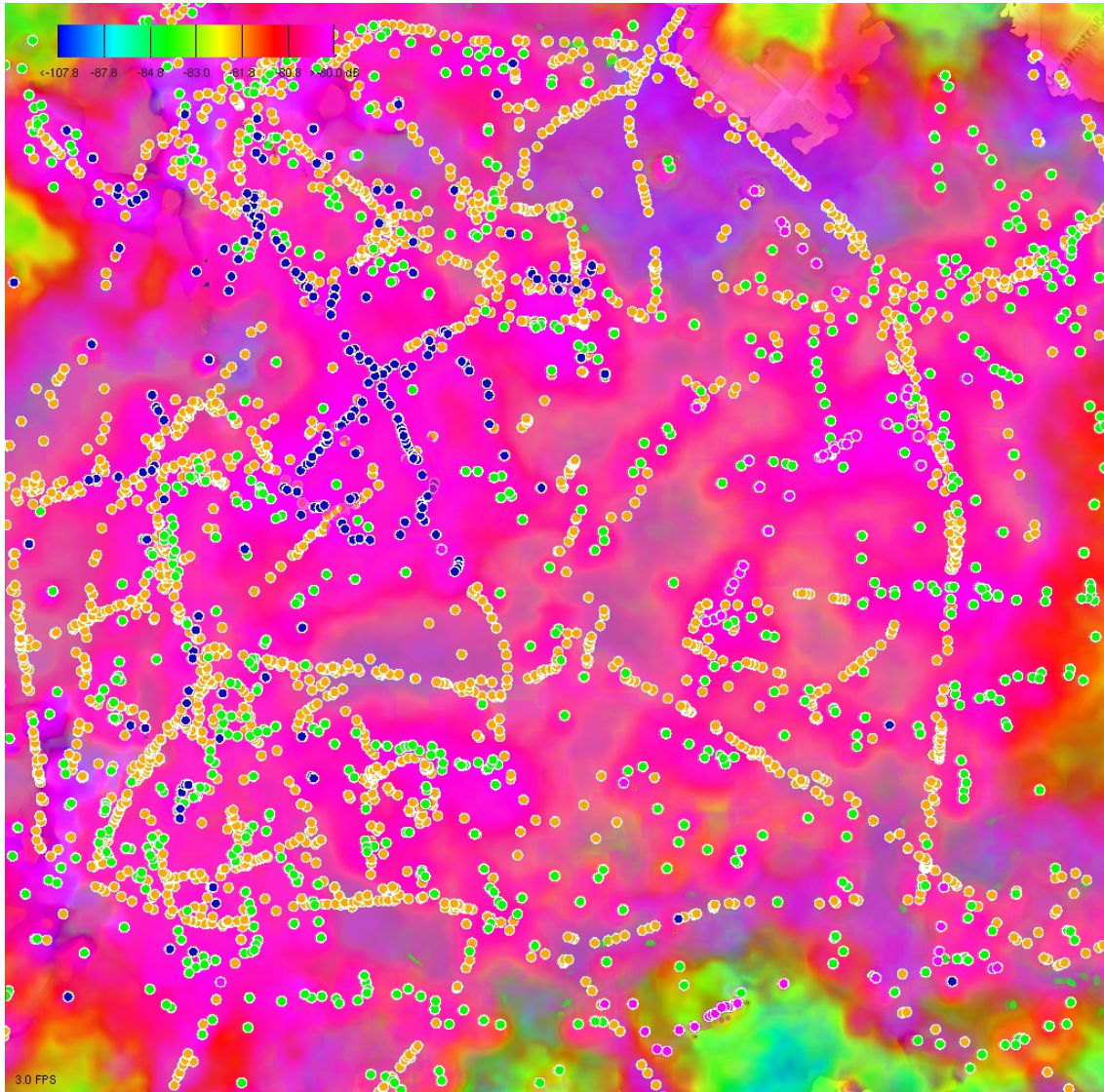


Users open and share their own Internet connection to community members







A community member can use all other Wi-Fi access points of the community

Wi-Fi Access Points in Aachen (city center)



#Access Points (estimated)

 T-Mobile:	67
 RWTH:	300
 ÖcherWLAN:	900
 Individuals:	2400

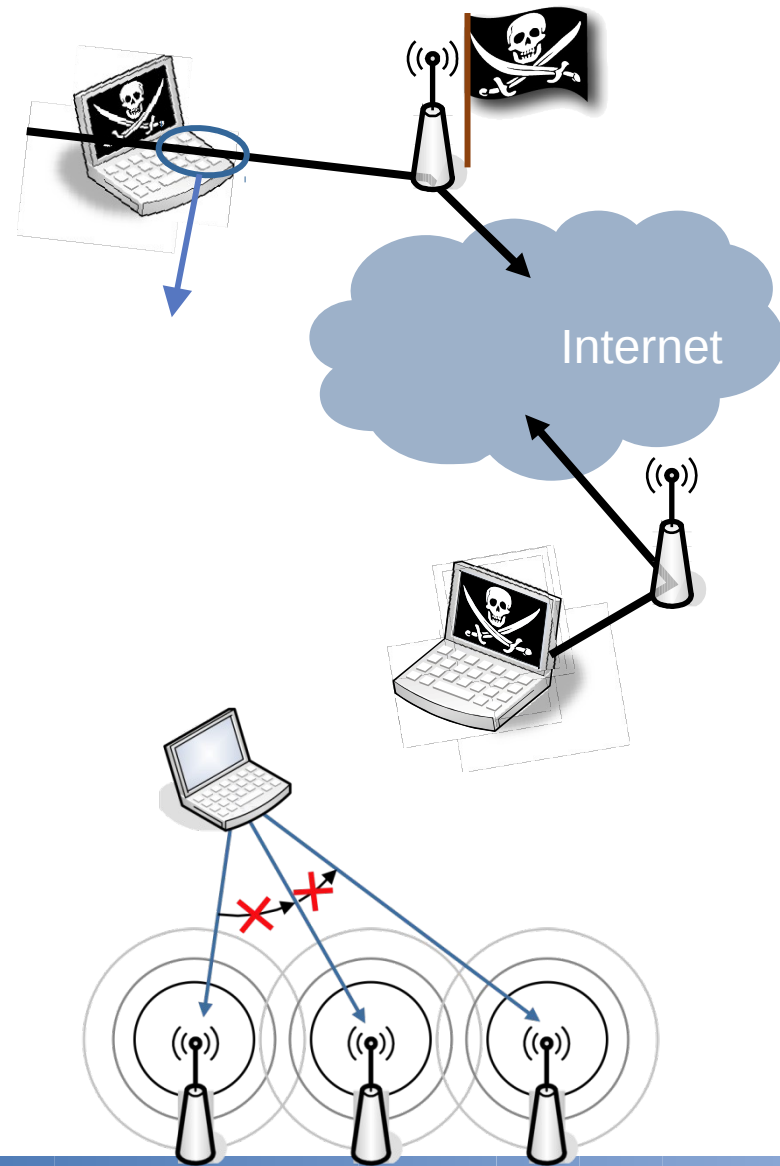
3.600

APs / 3 km²

(Sources: T-Mobile, Fon, BITKOM, OECD Report)

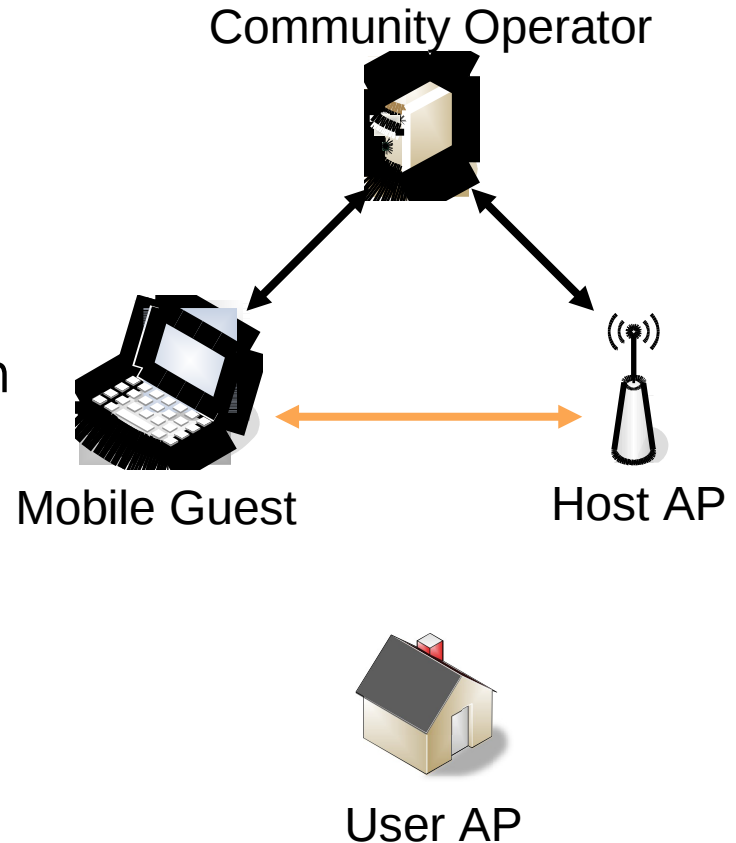
Drawbacks of Established Wi-Fi Sharing Communities

- Threats to mobile node
 - Compromised Community AP
 - Man-in-the-middle attacks
 - Eavesdropping
 - WiFi-link usually NOT encrypted
- Threats to AP provider
 - Misbehaving mobile node
 - Illegal actions relate to Host AP
 - Legal liability of access point owner (Landgericht HH 308 O 407 / 06)
 - Mobile node may access AP provider's LAN
- No roaming between Access Points

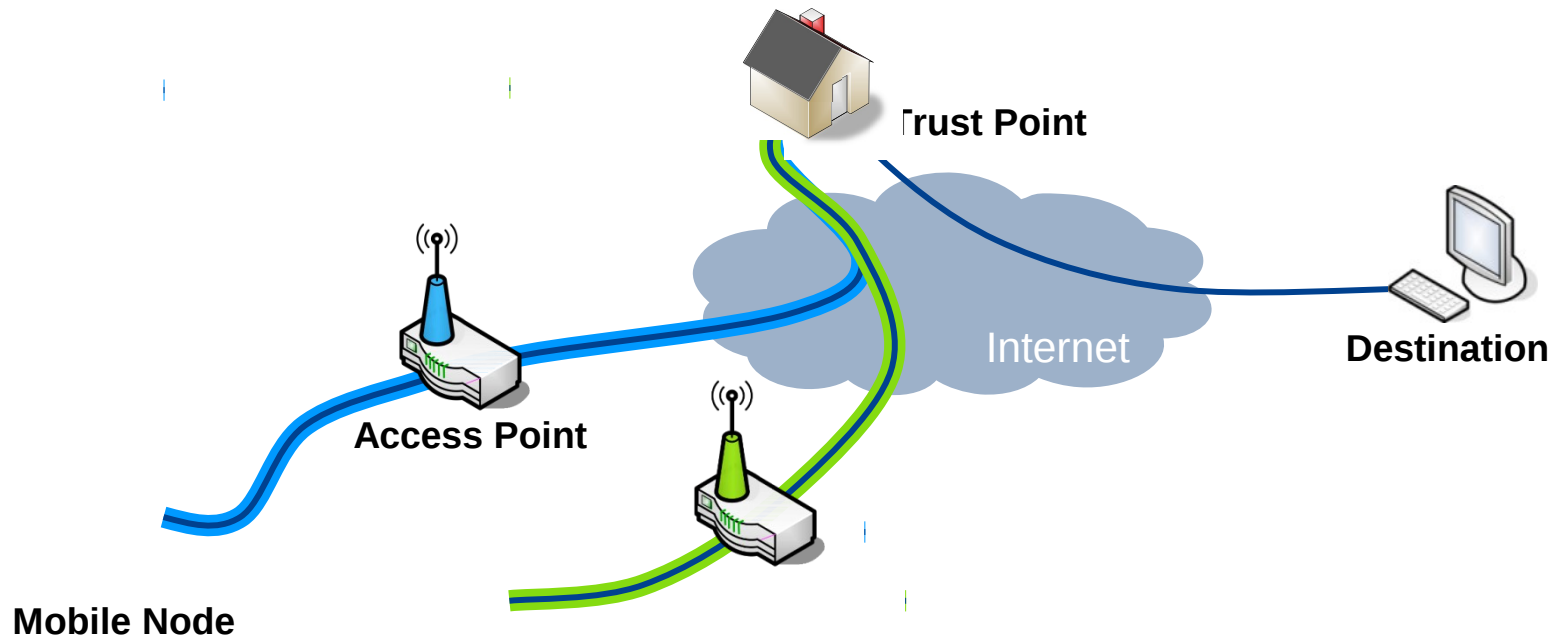


Trust Relations in Wi-Fi Sharing

- Trust relationship between Community and Mobile Guest
 - Mobile Guest is community member
- Trust relation between Host AP and Community
 - Host AP serves the community
- Assumption of a transitive trust relation
 - No technical enforcement
 - Questionable legal enforcement
 - High risk for Host AP and Mobile Guest
- User AP is not part of the model



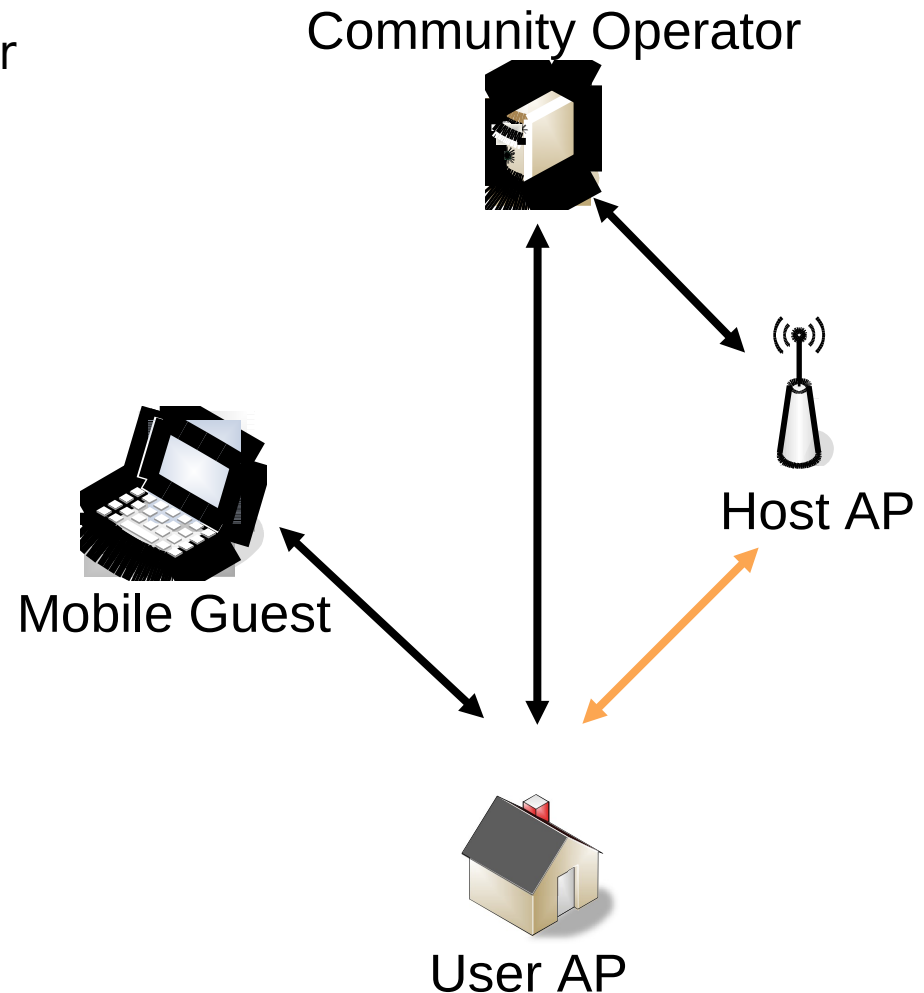
Simple Approach: Securely relocate Internet access to the MN's home



- Encrypted tunnel serves the AP provider and the mobile guest
 - Security and privacy for all parties
 - Solves legal liability problem
 - Transparent mobility support
 - Scalable, decentralized solution

Trust Relationships in PISA

- Community attests membership to User AP
- Host AP trusts the Community Operator
- Transitive trust relationship only ensures membership of User AP
- Strong trust relationship between User AP and Mobile Guest



Cryptographic Techniques Required in PISA

- Mutual authentication between Mobile Guest and User AP
 - Cryptographic identities → HIP
- Integrity protection and encryption between Mobile Guest and User AP
 - Secure tunnel → HIP
- Proof of community membership of User AP
 - Certificates → HIP Certificates
- Authentication of User AP towards Host AP
 - Cryptographic identities → HIP

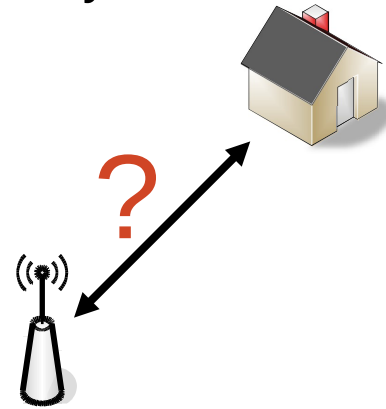
Authentication of the Trust Point

- HIP handshake

- Mutual authentication between Initiator and Responder only
 - Authenticated Diffie-Hellman key exchange
- User AP authentication towards Host AP is missing

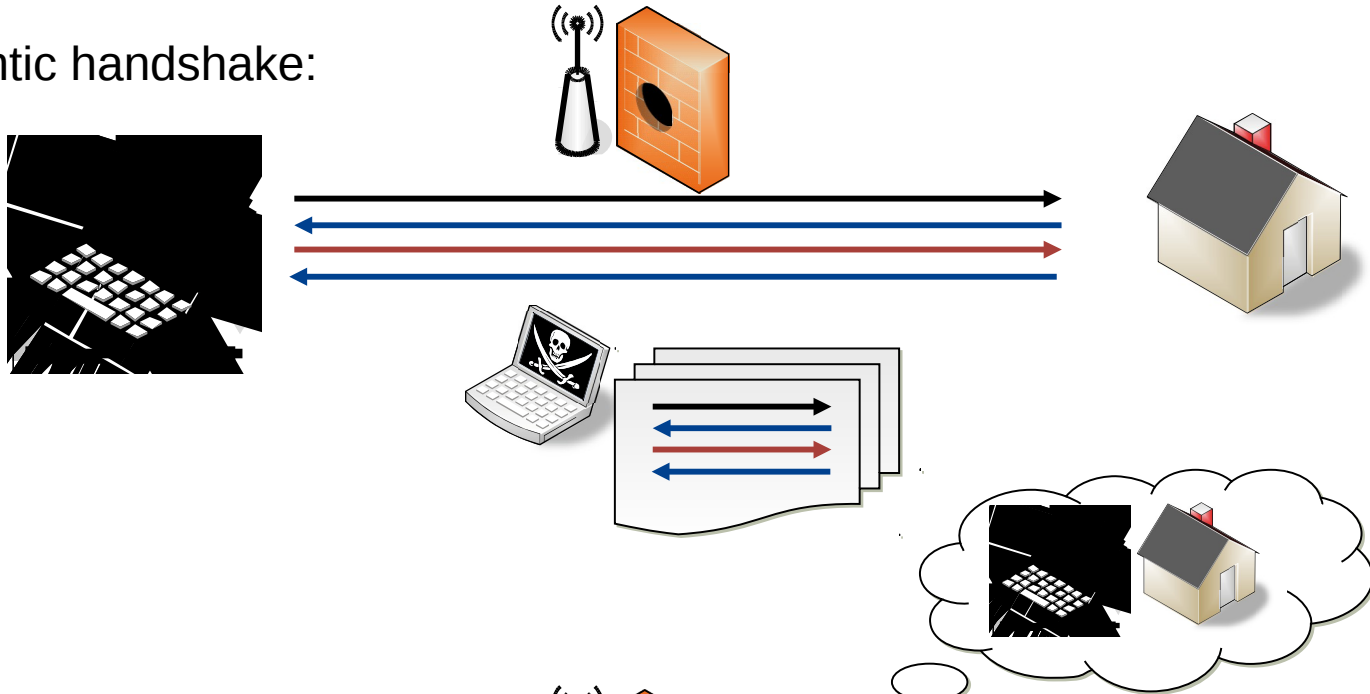
- PISA extends HIP handshake

- No separate handshake required
- Fewer packets: lower latency
- Lower protocol complexity

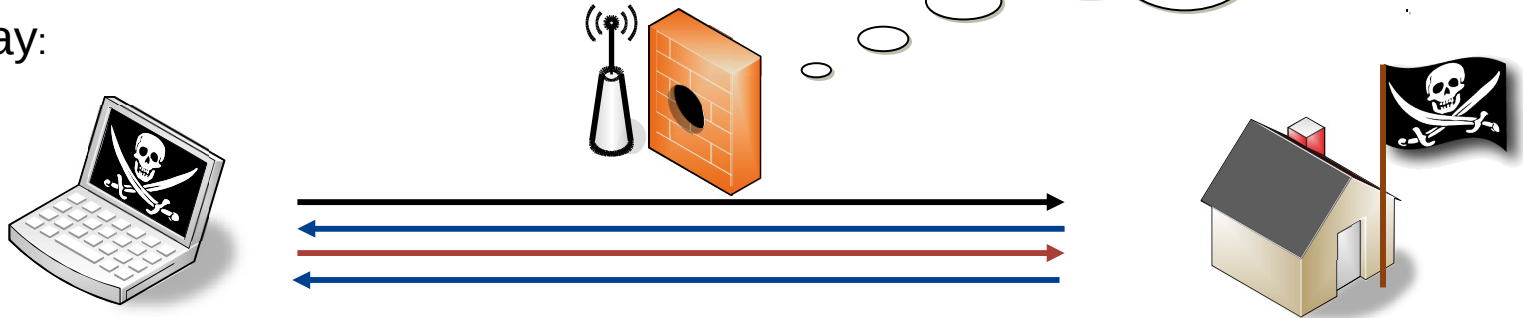


Authentication towards the Trust Point

1.) Authentic handshake:



2.) Replay:



PISA Extension for HIP



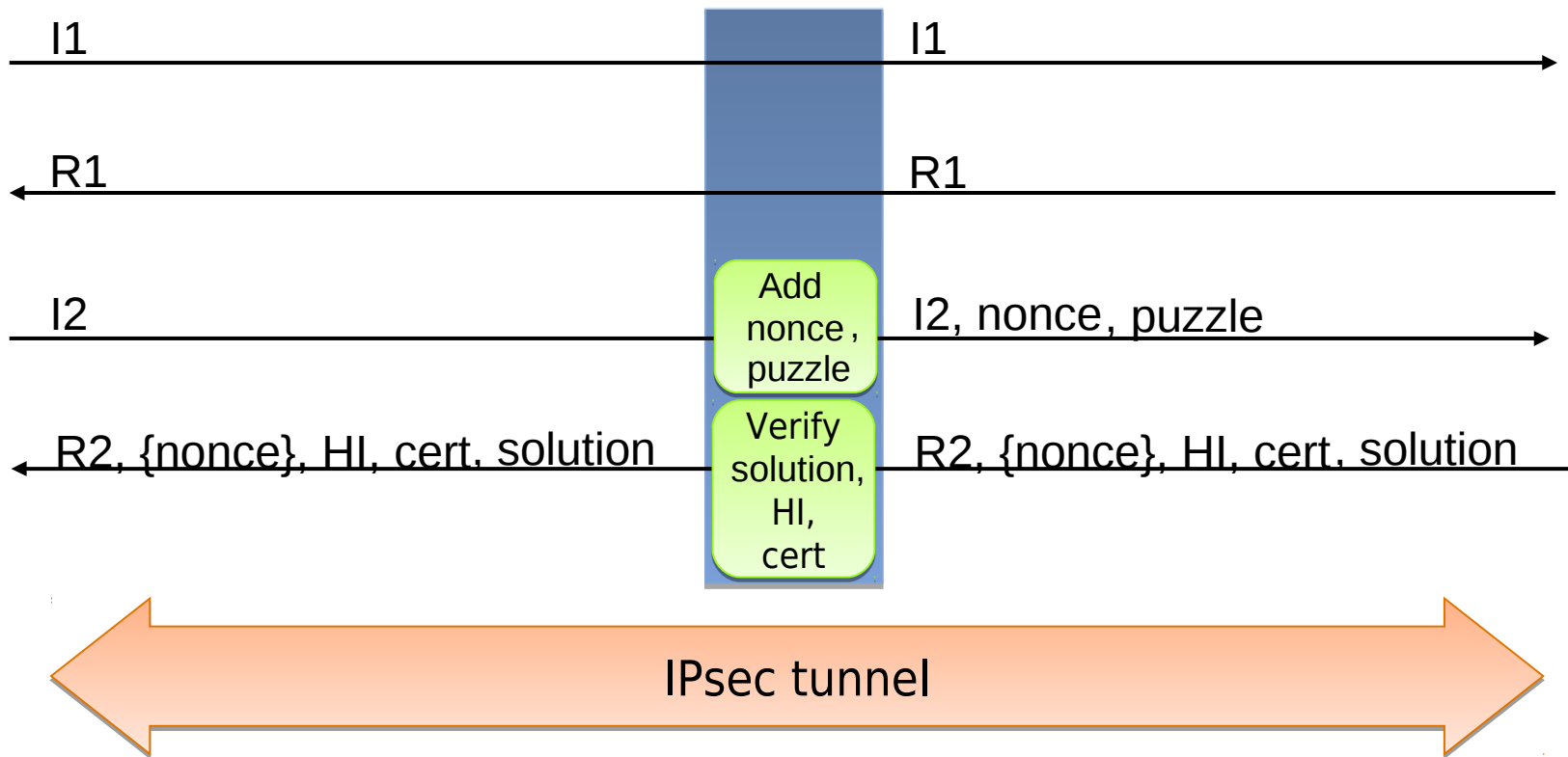
Mobile Guest (Initiator)



Host AP (Observer)

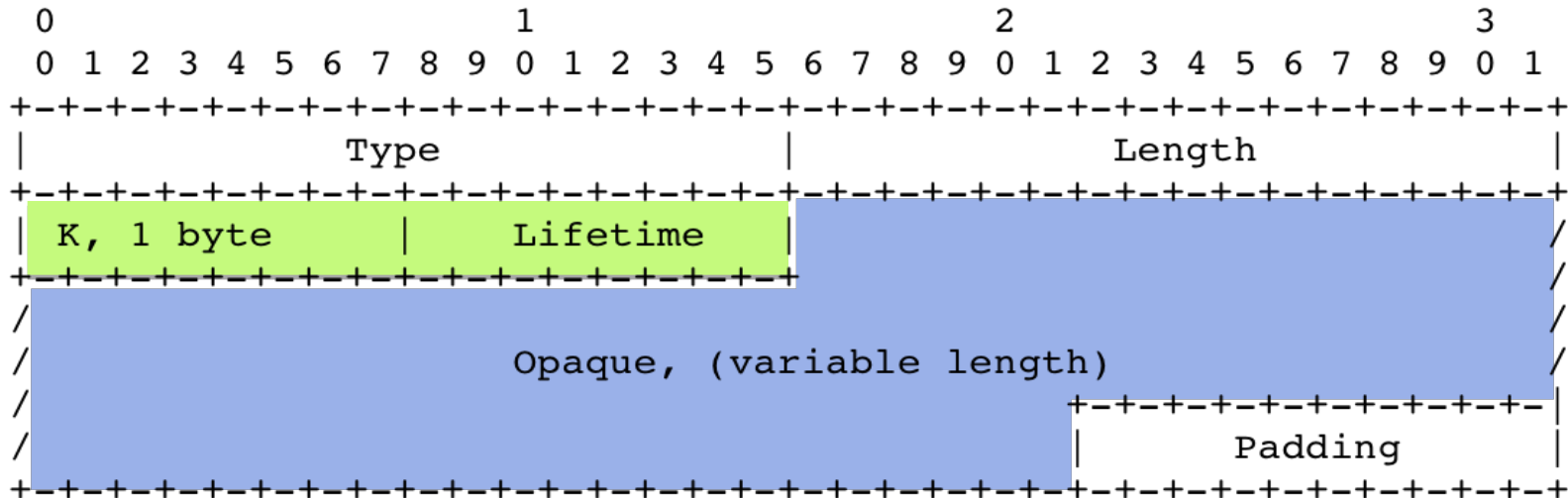


Trust Point (Responder)



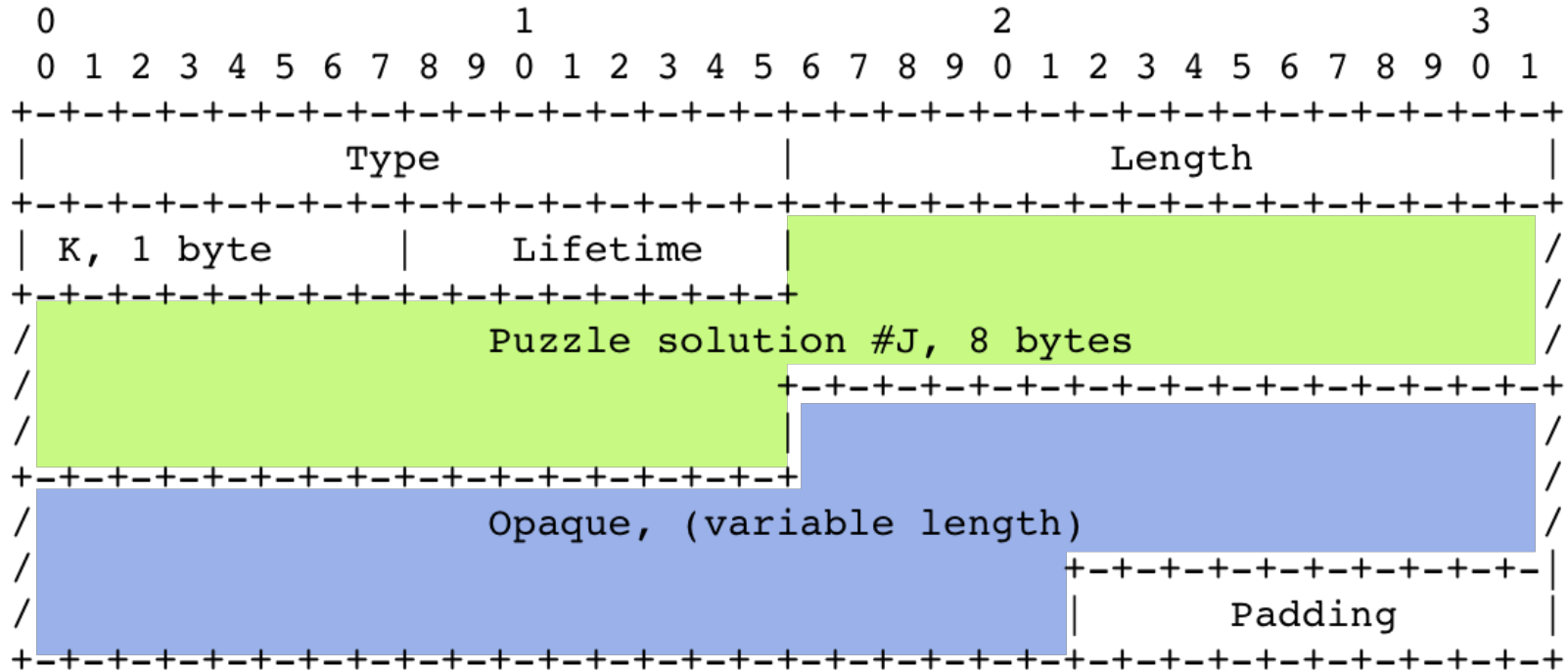
Combined Puzzle + Nonce Parameter

- CHALLENGE_REQUEST parameter
- One parameter for both functions
 - Simplified handling for cascaded middleboxes
- Opaque data serves as puzzle seed
 - Smaller size
 - Randomness can be added if needed



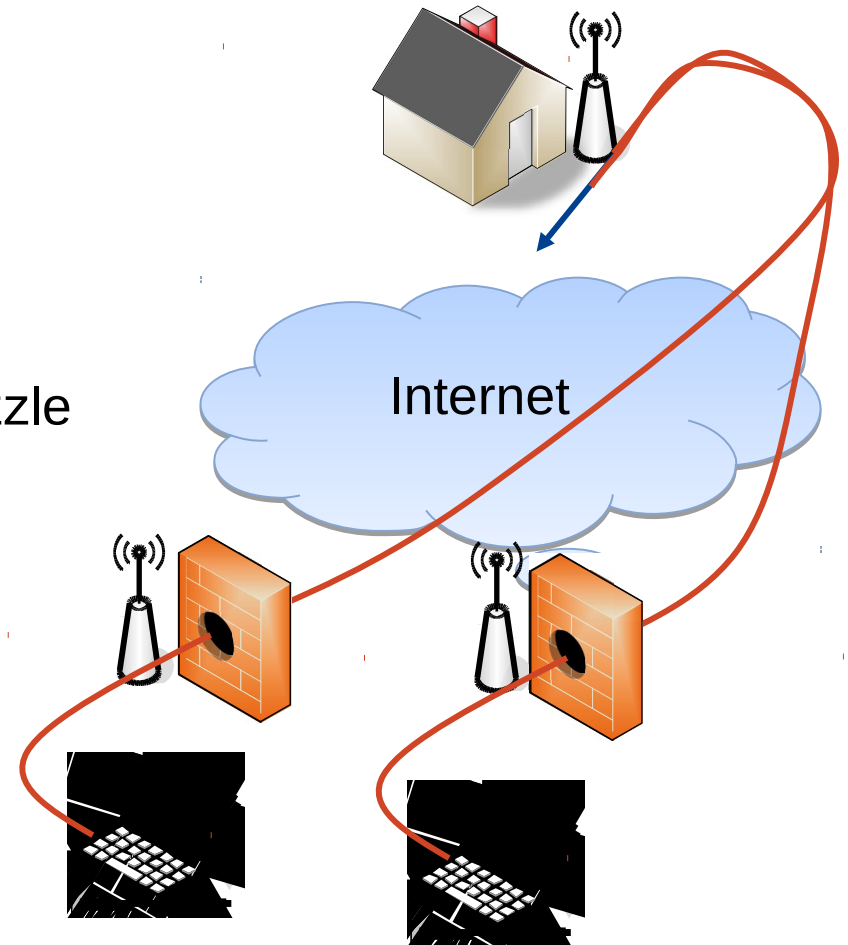
Solution and Response

- CHALLENGE_RESPONSE



Mobility with PISA

- HIP uses 3-way update process
 - Inform peer about new IP address
 - Adjust IPsec tunnel
- PISA extends HIP mobility
 - Authentication towards new TP
 - Host AP can inject nonce and puzzle







Anonymity and Traceability

- PISA combines authentication and mobility
 - Users become traceable
 - Careful design to prevent privacy issues
- Some identities can be concealed
 - Technical way: encrypt identity
- Distinction: real-world ID and digital ID



Authentication towards

	Mobile Guest 	User AP 	Host AP 	Community Operator 
Mobile Guest		Real ID	None	None
User AP	Real ID		Digital ID	Digital ID
Host AP	None	None		None

HIP Implementations

GNU-licensed Implementation

- HIP for Linux (HIPL)

- Developed by Helsinki Institute for Information Technology (HIIT)
- Generally licensed under GNU/GPLv2
- Platform-specific implementation for Linux
 - Uses IPsec BEET mode (included in Linux kernel > 2.6.27)
 - Additionally provides basic userspace IPsec implementation
- Includes firewall capabilities
- Supports OpenWRT Linux distribution for embedded devices
- Probably most complete implementation

- OpenHIP

- Developed by Boeing under GNU/GPLv2 license
- Multi-platform implementation: Linux, Windows, BSD, and Mac OS

- HIP for inter.net
 - Developed by Ericsson NomadicLab
 - New releases under BSD license
 - Primary platform: FreeBSD

HIP References

HIP References

- Moskowitz R and Nikander P (2006) „Host Identity Protocol Architecture“ RFC 4423, IETF
- Moskowitz R, Nikander P, Jokela P and Henderson T (2008) „Host Identity Protocol“ RFC 5201, IETF
- Jokela P, Moskowitz R and Nikander P (2008) „Using the Encapsulating Security Payload (ESP) Transport Format with the Host Identity Protocol“ RFC 5202, IETF
- Laganier J, Koponen T and Eggert L (2008) „Host Identity Protocol Registration Extension“ RFC 5203
- Laganier J and Eggert L (2008) „Host Identity Protocol Rendezvous Extension“ RFC 5204, IETF

HIP References

- Nikander P and Laganier J. (2008) „Host Identity Protocol Domain Name System (DNS) Extensions“ RFC 5205, IETF
- Nikander P, Henderson T, Vogt C and Arkko J (2008) „End-Host Mobility and Multihoming with the Host Identity Protocol“ RFC 5206, IETF
- Stiemerling M, Quittek J and Eggert L (2008) „NAT and Firewall Traversal Issues of Host Identity Protocol Communication“ RFC 5207, IETF
- A. Gurtov „Host Identity Protocol (HIP): Towards the Secure Mobile Internet“, ISBN 978-0-470-99790-1, Wiley and Sons, June 2008 (Hardcover, 332 p)

HIP References

- Nikander P, Laganier J and Dupont F (2007) „An IPv6 prefix for overlay routable cryptographic hash identifiers (ORCHID)“ RFC 4843, IETF
- Heer T, Wehrle K, Komu M (2009) „End-Host Authentication for HIP Middleboxes“ Draft, IETF
- Varjonen S, Heer T (2008) „HIP Certificates“ Draft, IETF
- Heer T, Goetz S, Weingaertner E and Wehrle K (2008) „Secure Wi-Fi Sharing on Global Scales“ IEEE
- Heer T, Hummen R, Komu M, Götz S and Wehrle K (2009) „End-host Authentication and Authorization for Middleboxes based on a Cryptographic Namespace“ IEEE