Securing Ad Hoc Networks and Vehicular Communications

Tutorial at SBSEG 2007

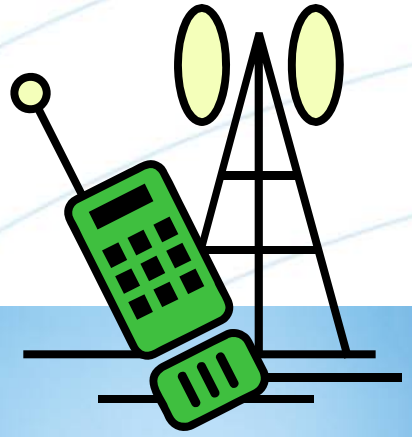
Panos Papadimitratos
panos.papadimitratos@epfl.ch

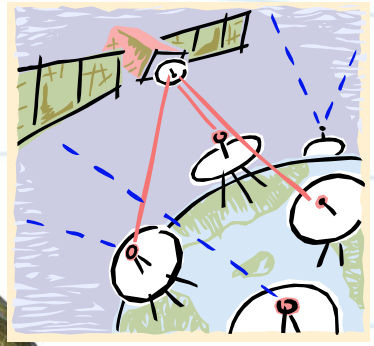
Securing Ad Hoc Networks and Vehicular Communications

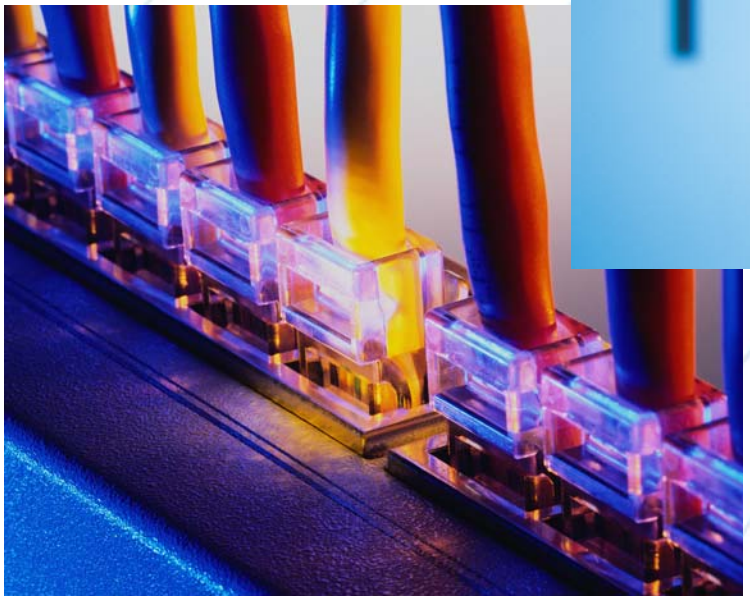
Part 1: Securing Ad Hoc Networks

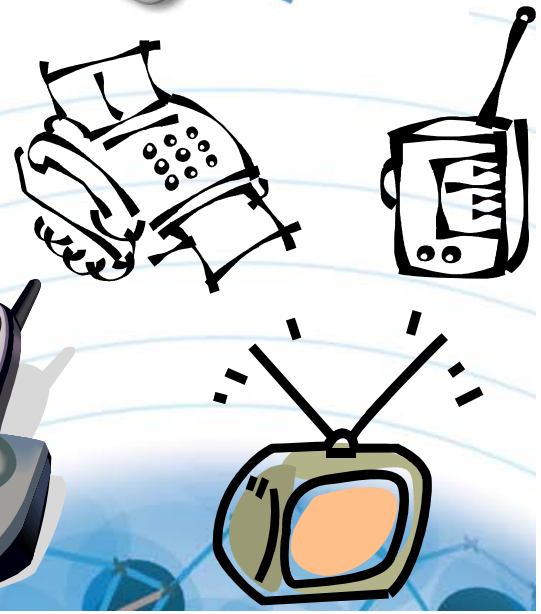
The Internet

- Exchanging Information









The Internet (cont'd)

- Applications and protocols

Search Engines

E-Commerce

File Sharing

Email

Voice-over-IP

Video Streaming

TCP/IP

HTTP

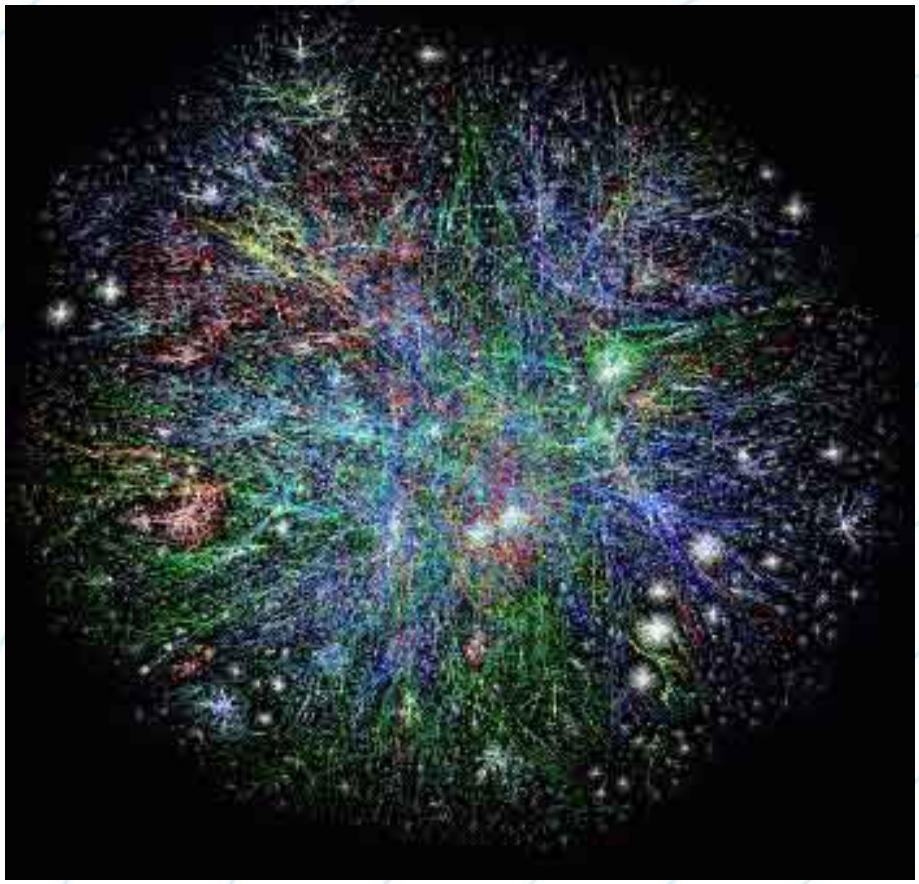
SMTP

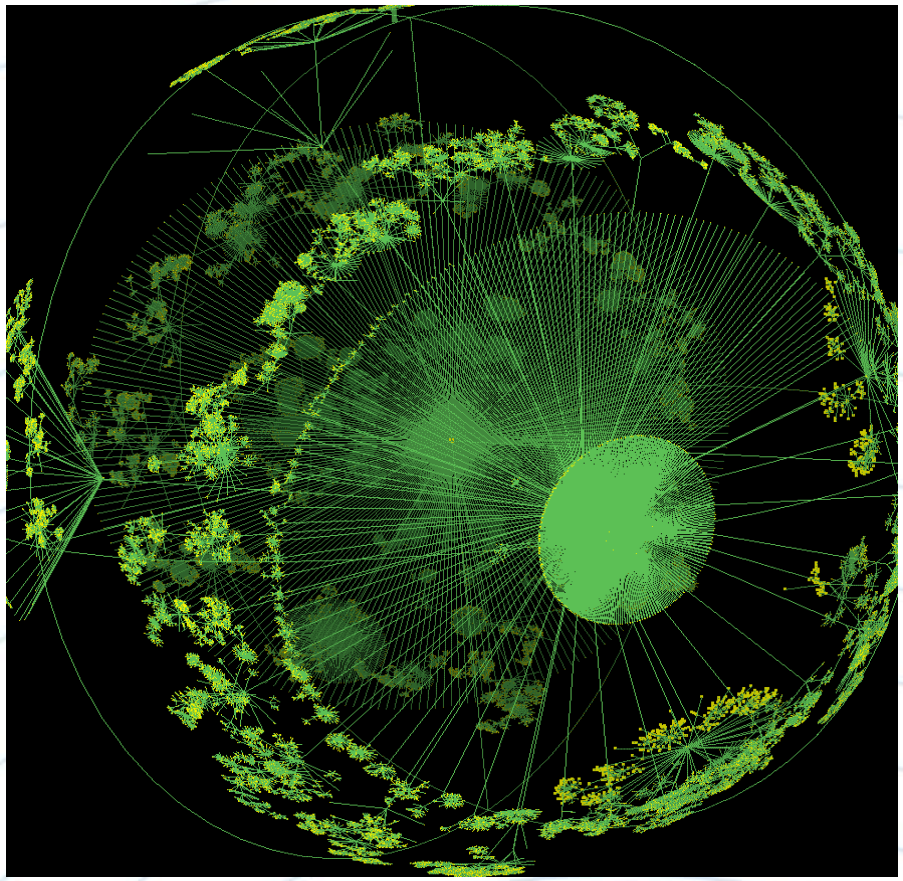
FTP

SSL

The Internet (cont'd)

- Large number of interconnected systems

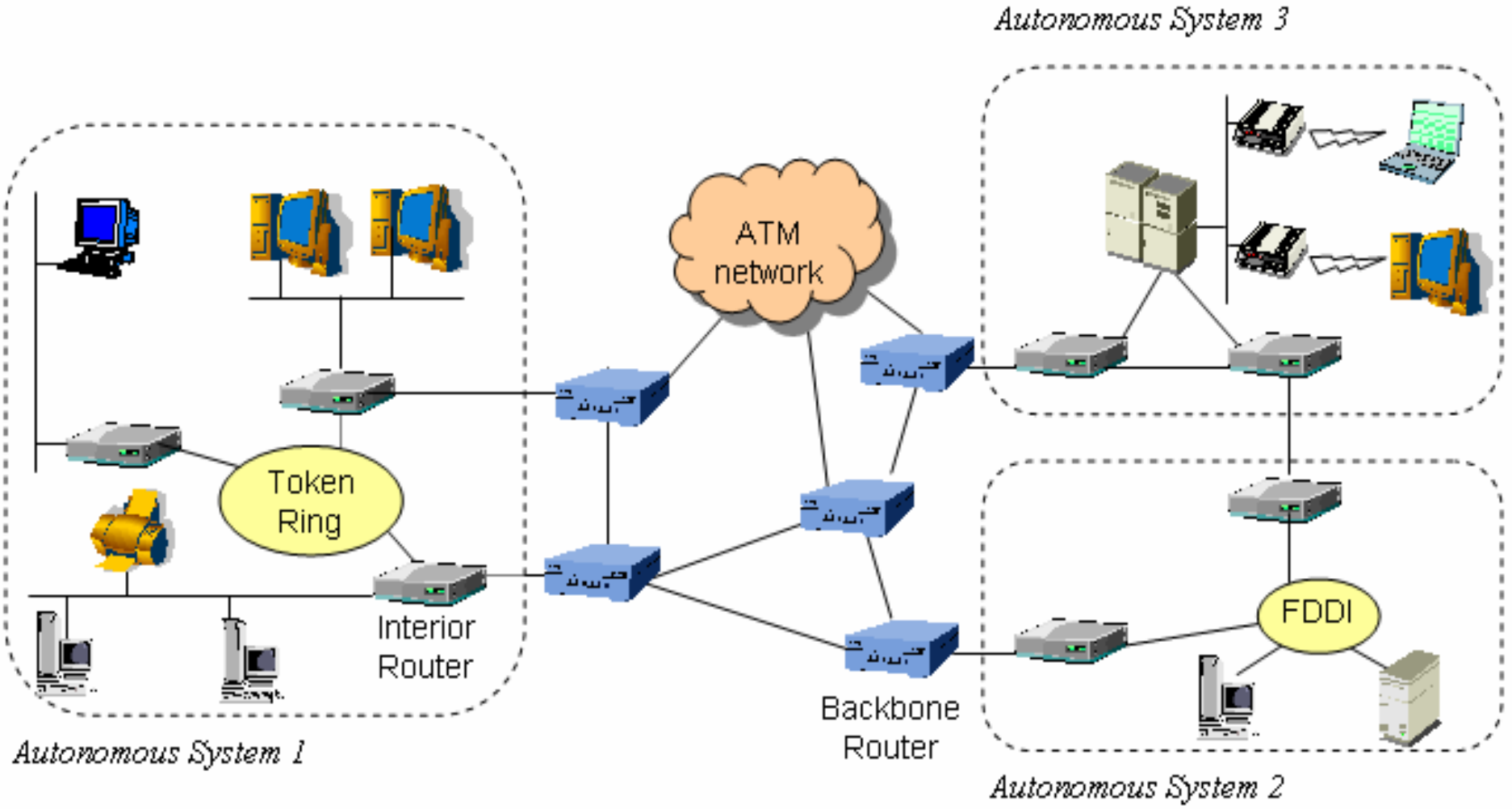




Graphics by CAIDA

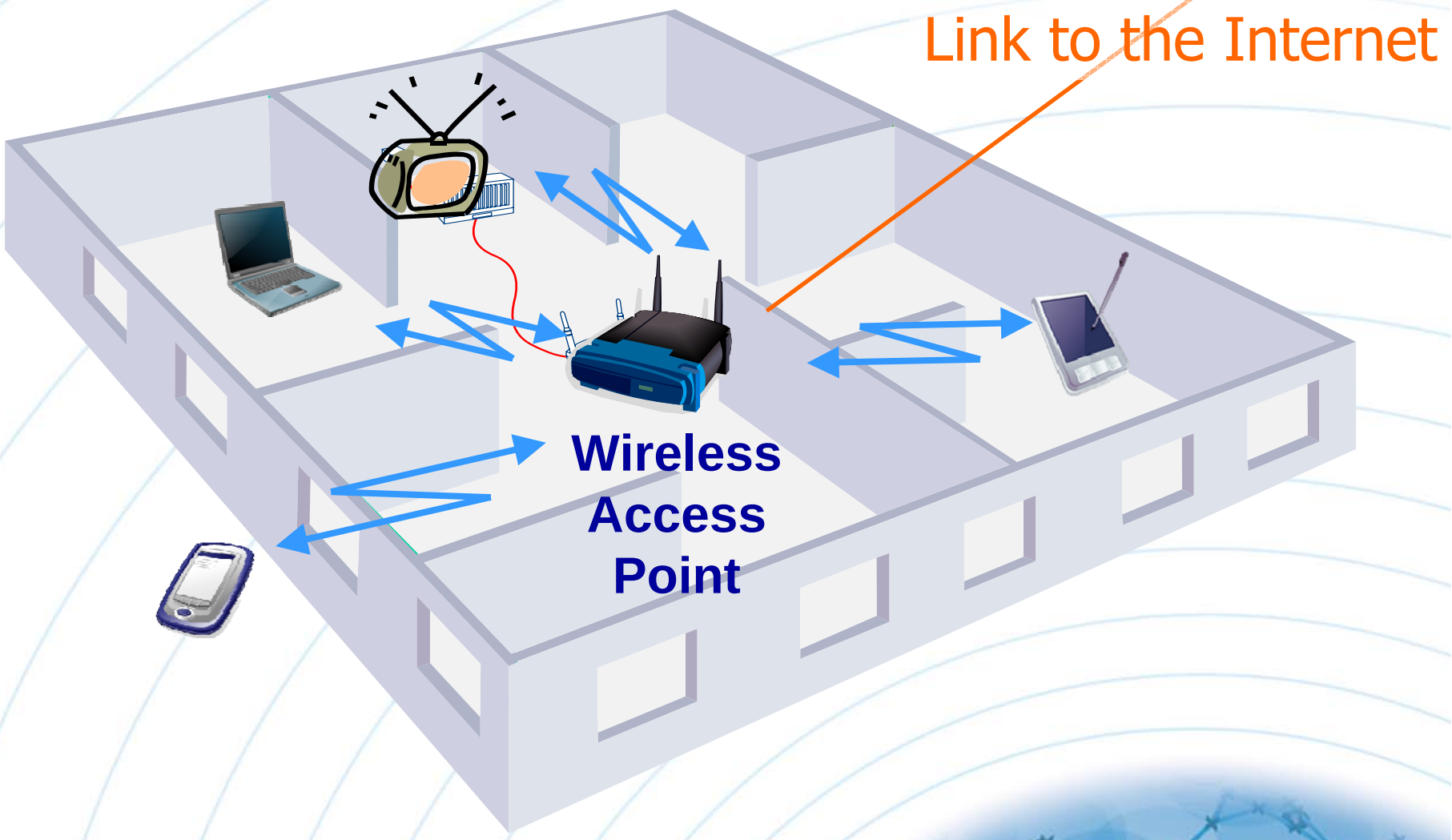
The Internet (cont'd)

- Interconnected Packet Switching Networks



Ad hoc networks: towards a pervasive Internet

- Wireless local area networks (WLANs)



Ad hoc networks (cont'd)

- WLANs and Personal Area Networks (PANs)

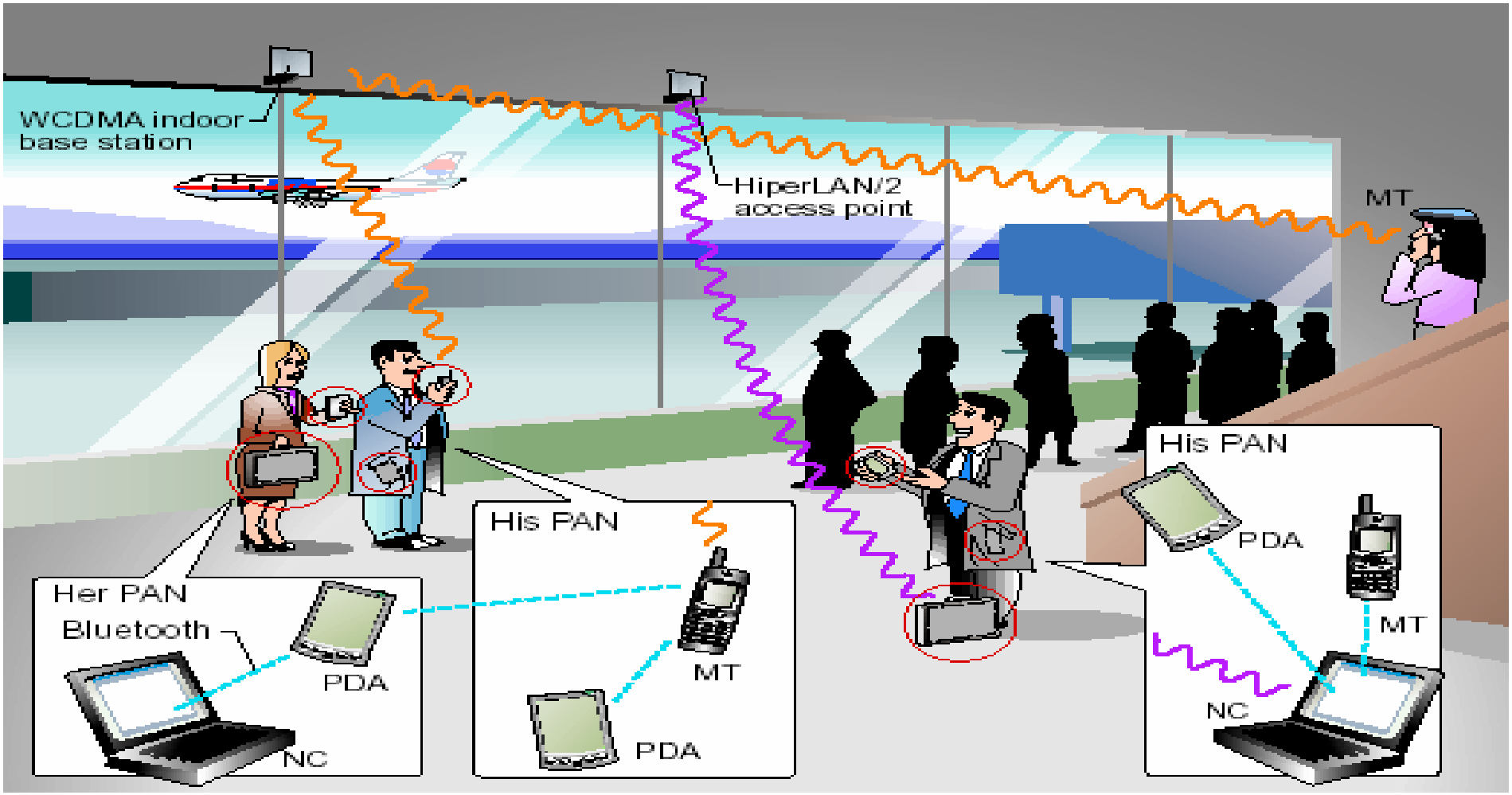


Illustration by Ericsson

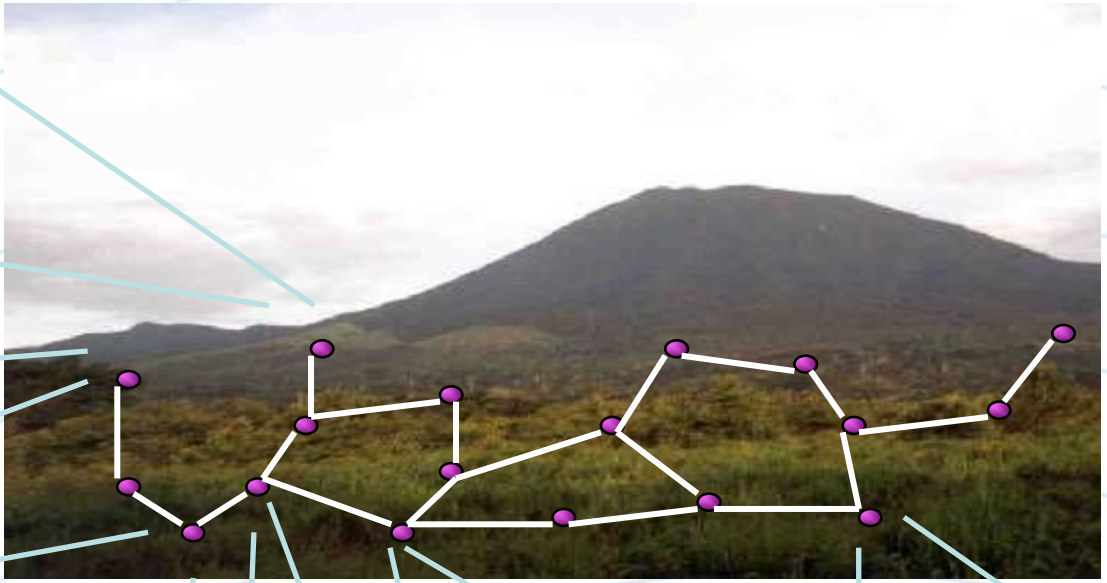
Ad hoc networks (cont'd)

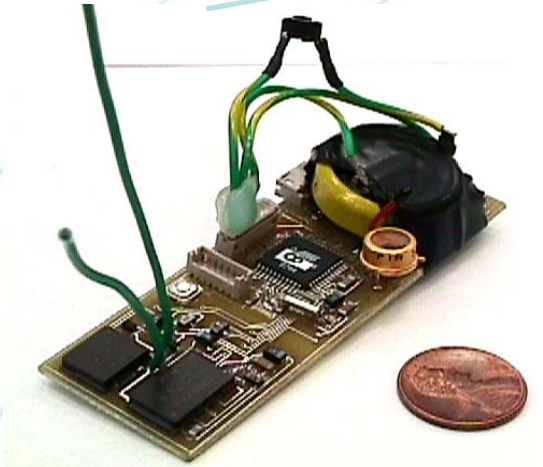
- Vehicular ad hoc networks (VANETs)

Illustration by the Car-to-Car Communication Consortium

Ad hoc networks (cont'd)

- Sensor networks





Photos by XBow

Ad hoc networks (cont'd)

- The Interplanetary Internet

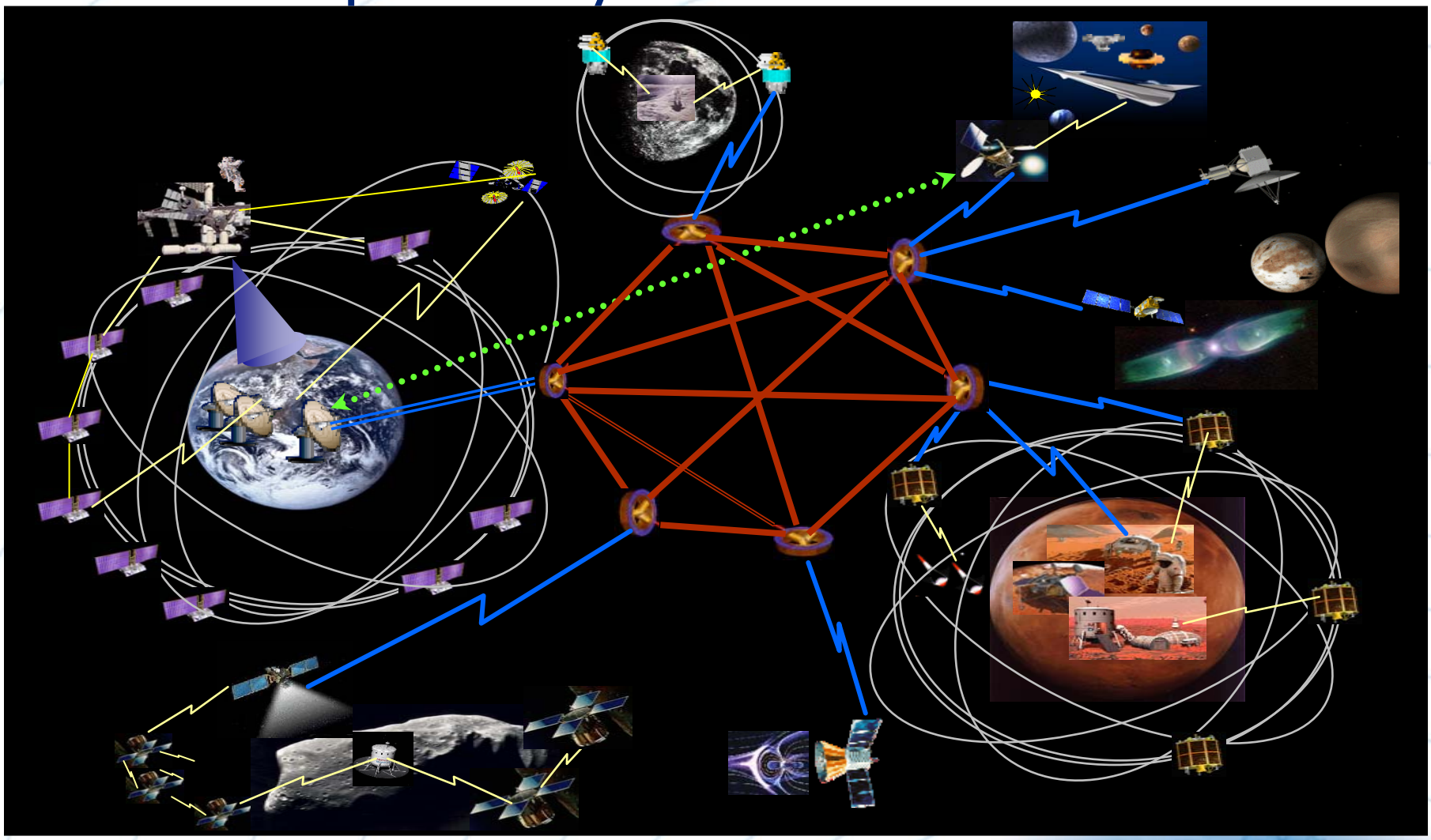


Illustration by JPL

Ad hoc networks (cont'd)

- On other planets

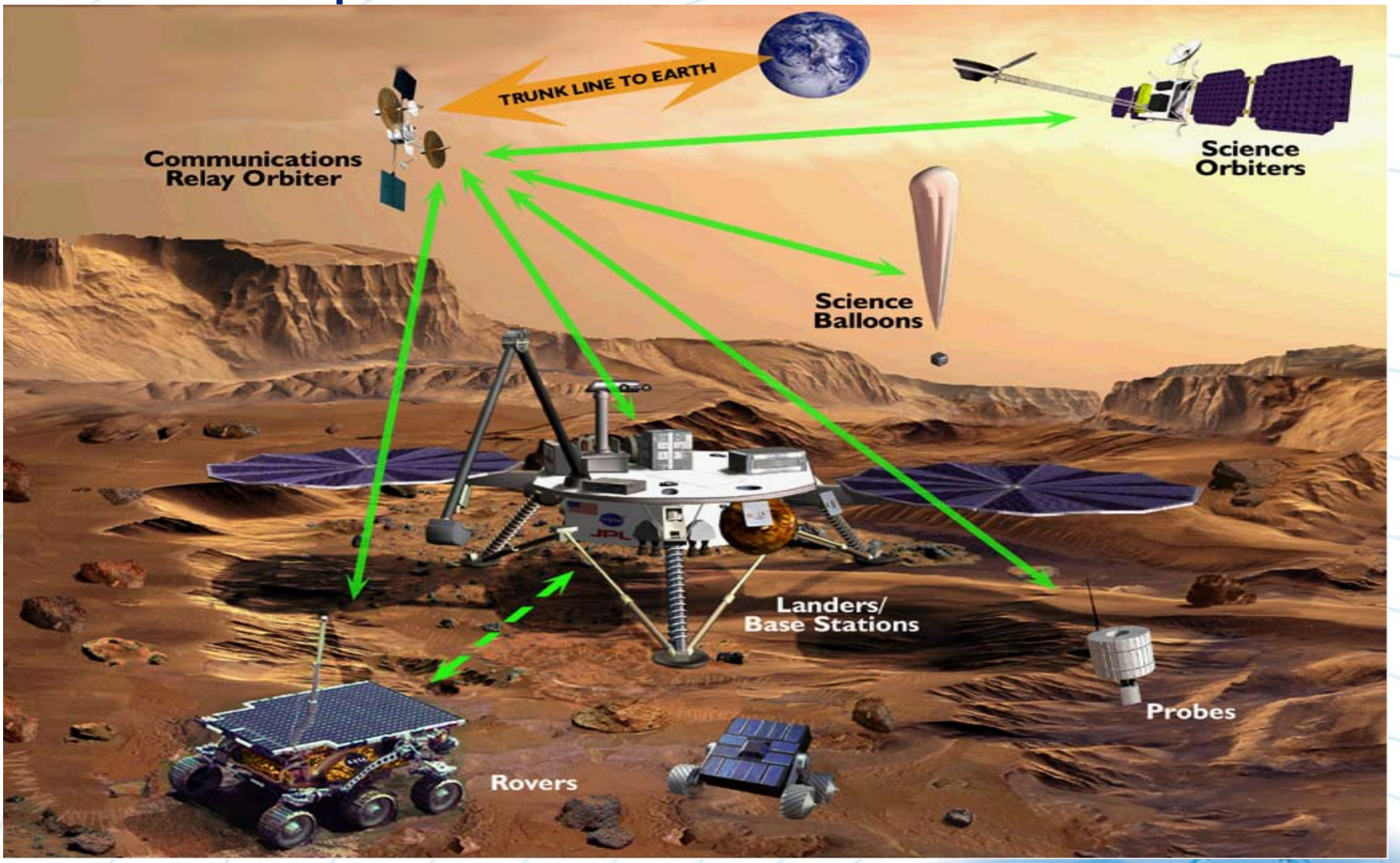
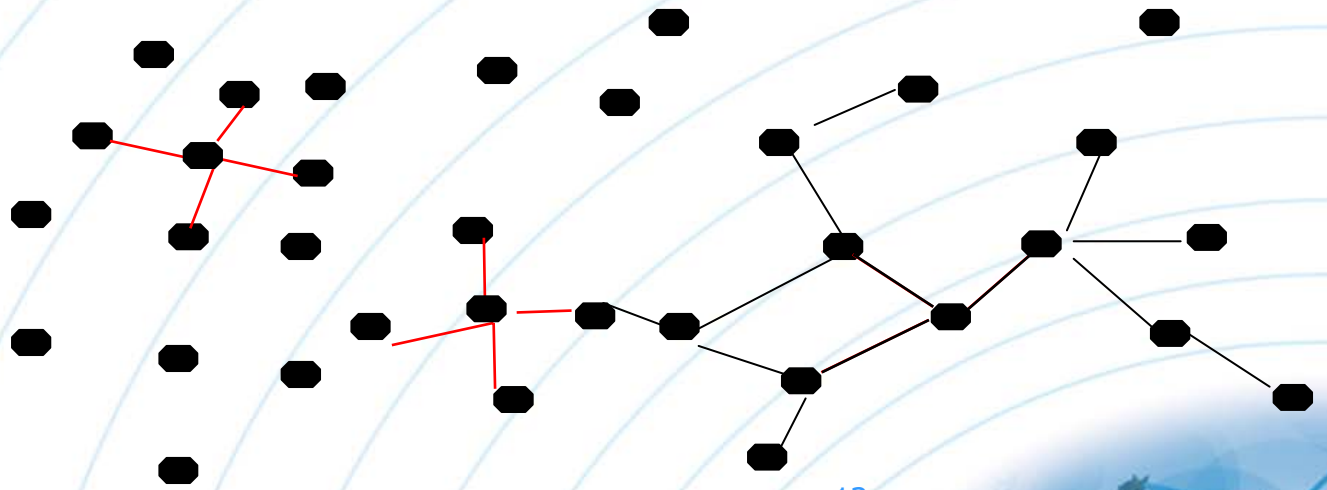


Illustration by JPL

Ad hoc networks (cont'd)

- No fixed infrastructure
- Collaborative support of the network operation
- Peer-to-peer interaction
- Transient associations
- No administrative boundaries



Security challenges

- Ad hoc networks can operate in hostile environments, e.g., tactical networks
- Ad hoc networks cannot comprise only collaborative and correct (i.e., not faulty) nodes
  - HARD LESSON LEARNED BY THE WIRELINE INTERNET
- Wireless communication makes eavesdropping and message injection easy
- Each and every node can disrupt the network operation
- Difficult or impossible to distinguish between benign and malicious faults
- No central authority and monitoring facility
- Frequent network changes

Overall objective

- Design networking protocols that manage and tolerate malicious and selfish nodes, a.k.a adversaries or attackers
  - Leave as little space as possible for attackers to deviate from the protocols and disrupt the network operation
  - Build fault-tolerance features to mitigate the impact of misbehavior
- Secure communication and maintain end-to-end connectivity in the presence of adversaries

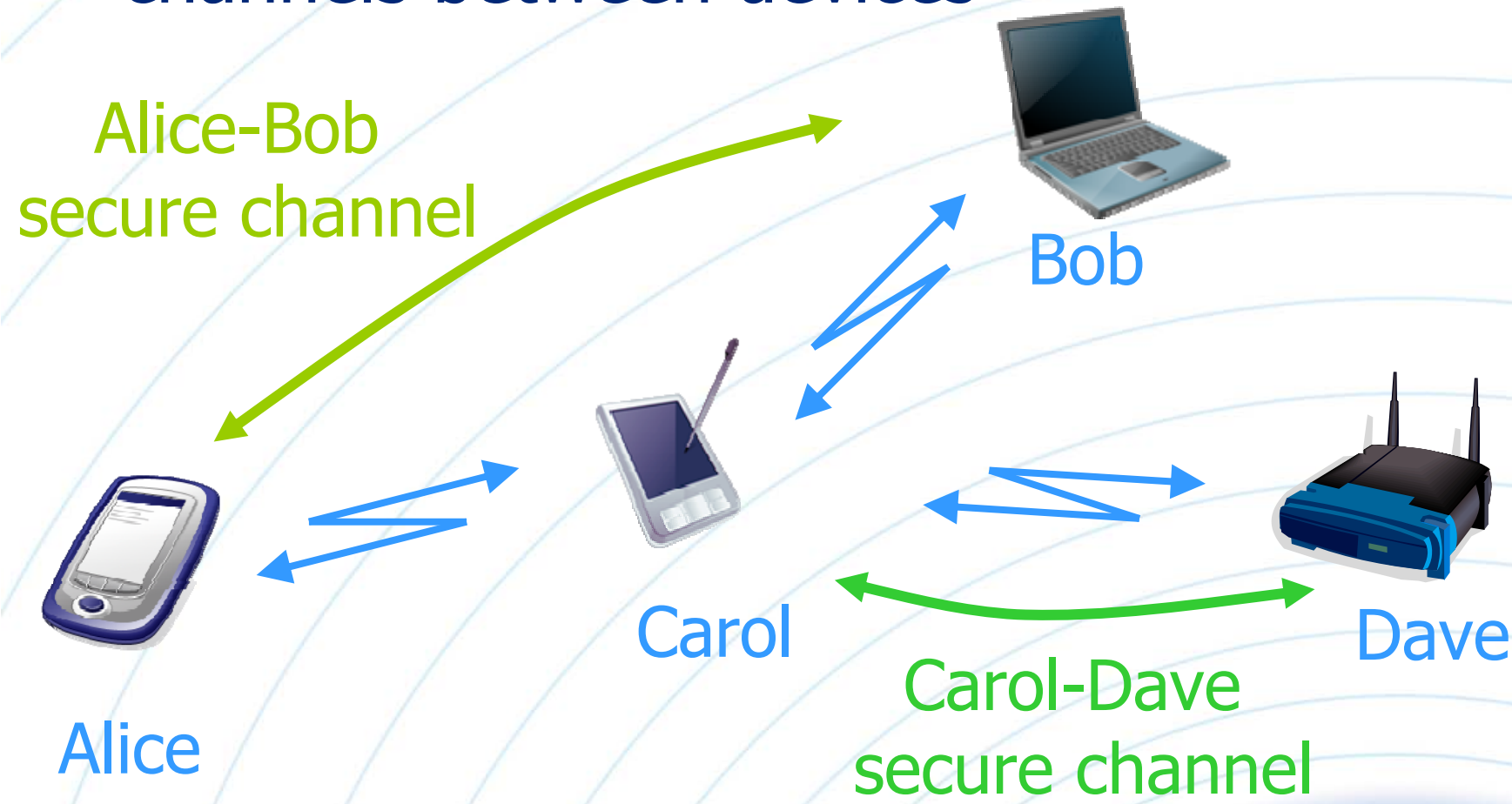
Outline

- Part 1 topics
  - Security Association Establishment
  - Secure Neighbor Discovery
  - Secure Route Discovery
  - Secure Data Communication

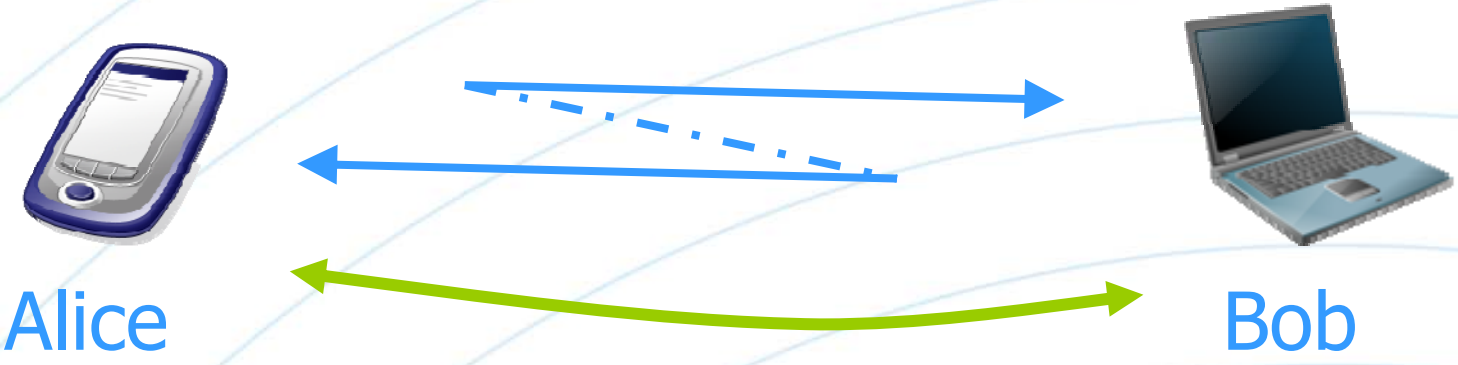
Security Association Establishment

Problem statement

- Establishing secure communication channels between devices

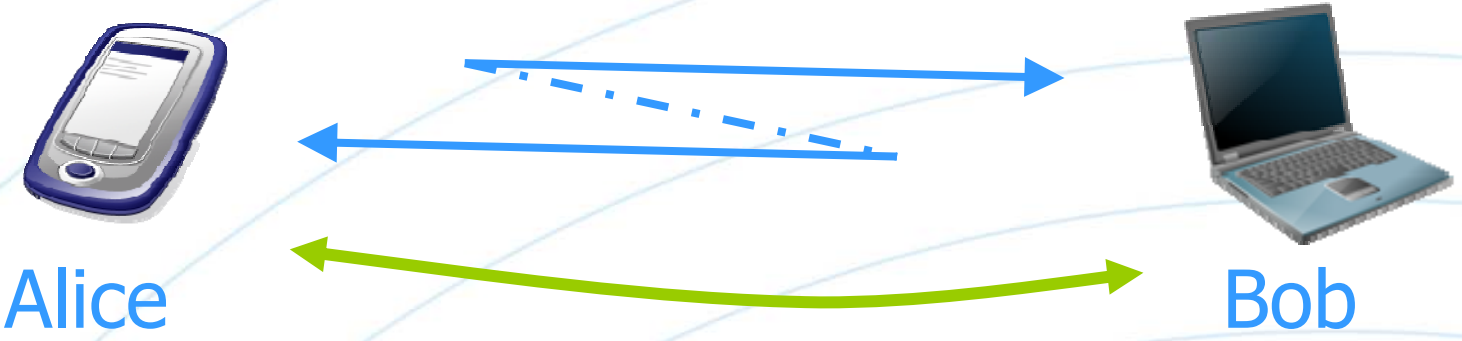


Problem statement (cont'd)



- Security requirements
  - Authentication
  - Integrity
  - Confidentiality
  - Non-repudiation
  - …

Problem statement (cont'd)

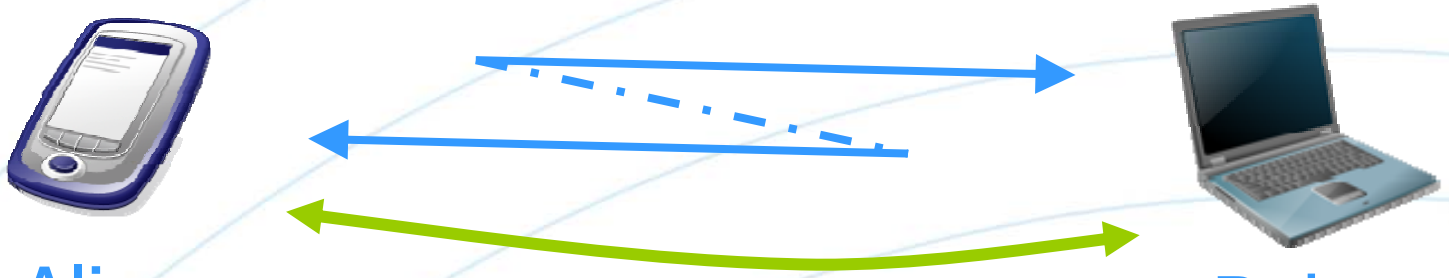


- Security mechanisms
  - Message Authentication Codes (MACs)
  - Digital signatures
  - Encryption/decryption
  - Passwords
  - …
- Cryptography
  - Asymmetric key
  - Symmetric key

Problem statement (cont'd)

- Enable secure communication
  - Uni-directional
  - Bi-directional
- Issues to consider
  - Long- or short- term?
  - What fraction of the system nodes?
  - Is there a trusted third party?
  - …

Public-key approach



Alice

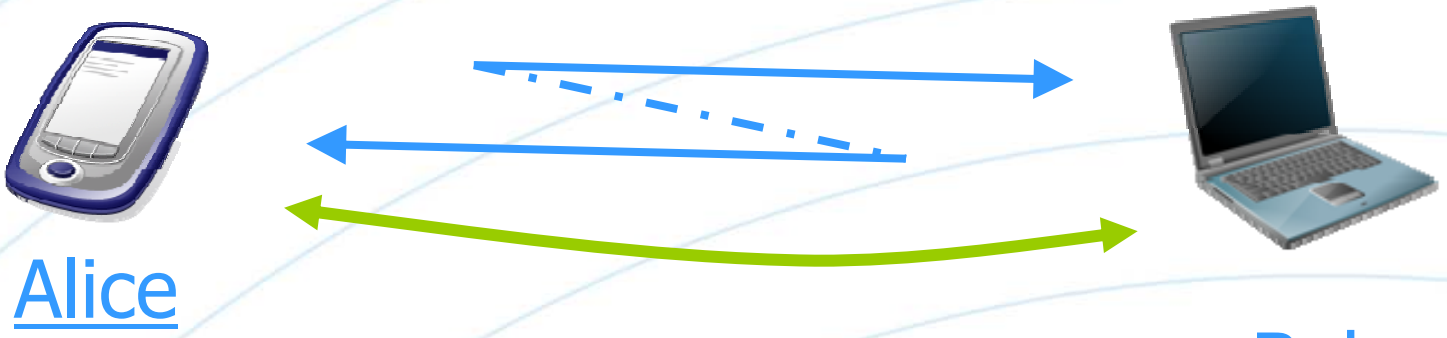
Identity: A
Public key: K_A
Private key: k_A

Bob

Identity: B
Public key: K_B
Private key: k_B

- Pro: Any-to-any secure communication
- Con: Need to bind public keys to identities

Public-key approach (cont'd)



- Secure communication example
  - Message (1): signed with k_A; n_A is a nonce
  - Message (2): sec-text and B encrypted with K_A; A, n_A, and ciphertext signed with k_B
  - Note: In practice, different keys are used for signing/verifying and encrypting/decrypting

Public-key approach (cont'd)

Alice

Identity: A
Public key: K_A
Private key: k_A
$Cert_{CA}\{A,K_A\}$

Bob

Identity: B
Public key: K_B
Private key: k_B
$Cert_{CA}\{B,K_B\}$

- Certification Authority(CA)
  - Trusted Third Party
  - Known K_{CA}
  - $Cert_{CA}$: CA signature on the identity, public key, and other information (e.g., lifetime)

Using a CA

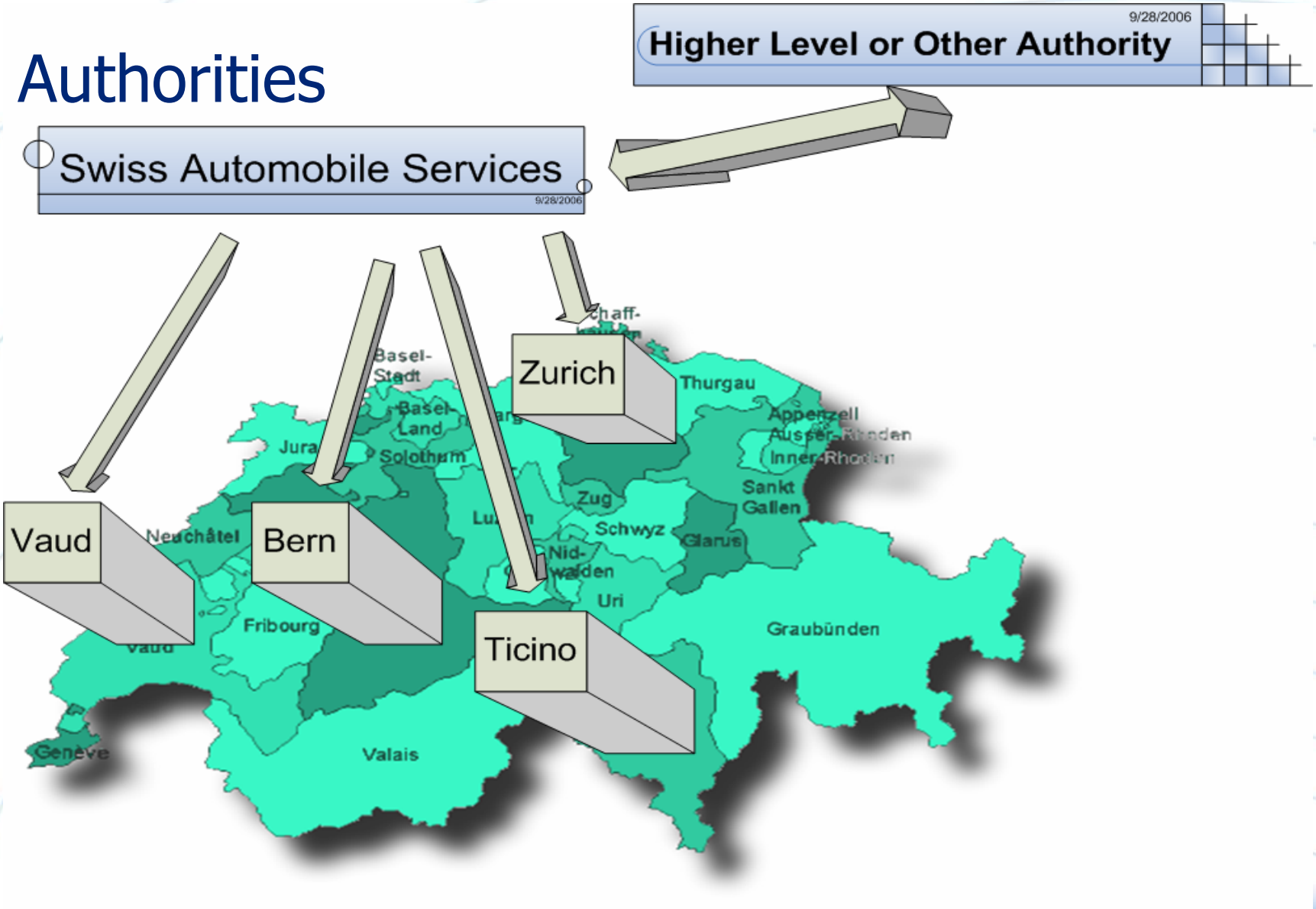
- Largely independent of communication
  - Users obtain certificates over the wire-line network
  - Certificates are installed at wireless devices and the corresponding keys used to secure wireless communication
- Examples specific to wireless networks

Using a CA (cont'd)

- Wireless local-area (e.g., campus-wide) networks
  - CA locally administered
  - IEEE 802.11 devices communicate securely with access points
- Tactical networks
  - CA operated by the corresponding government department
  - Keys and certificates installed at wireless-enabled devices
  - Hierarchical network organization
- Vehicular Communication (VC) Systems
    - More detailed look next

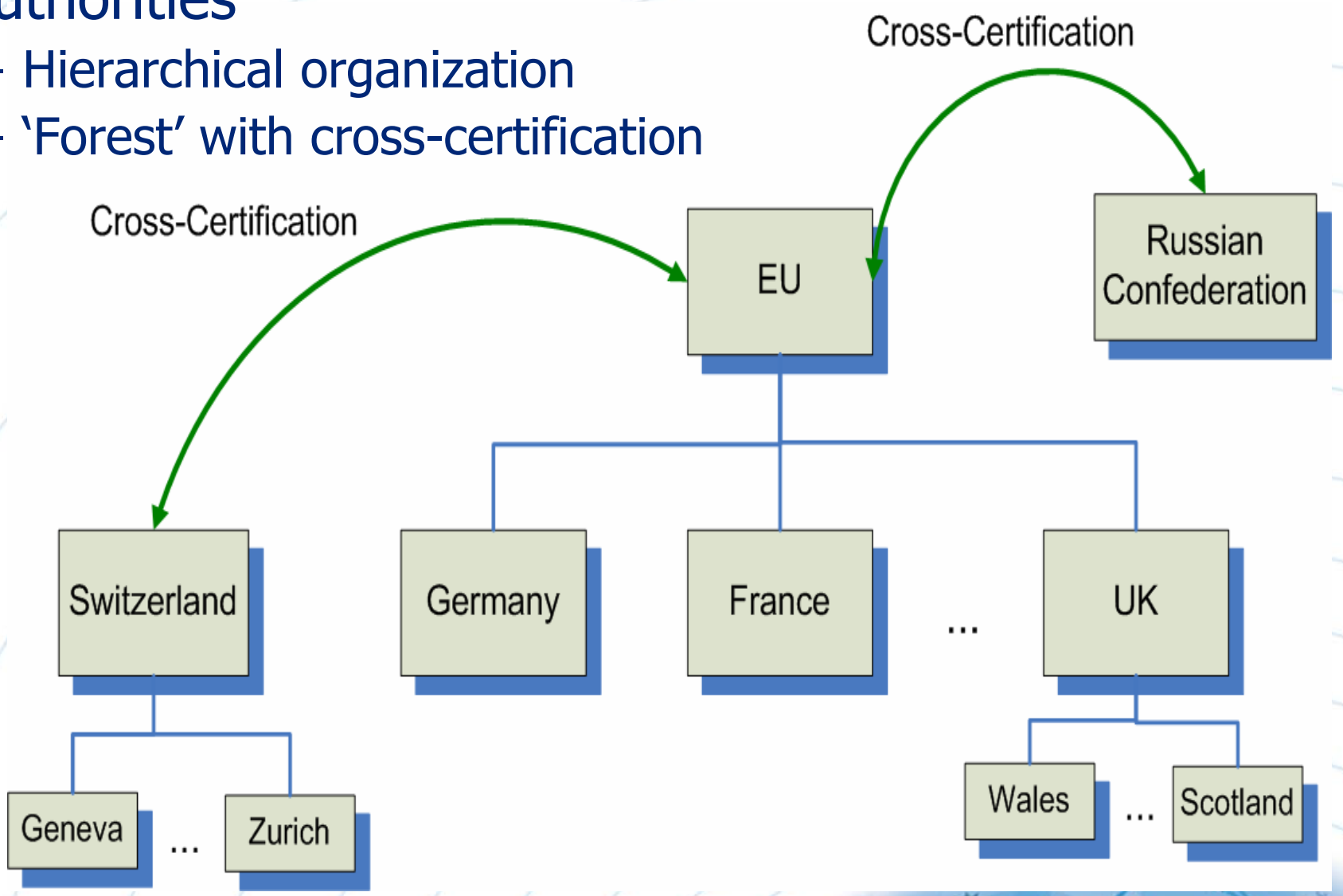
CA example: Vehicular Networks

- Authorities



CA example: Vehicular networks (cont'd)

- Authorities
  - Hierarchical organization
  - 'Forest' with cross-certification



Public key cryptography - Practical aspects

- There is no single trusted authority
  - Nodes belonging to different administrative domains will in general be associated and execute security protocols
- PK cryptography is feasible even in low-end mobile platforms, but it is costly
  - Processing
  - Energy consumption
  - Delays
  - Transmission overhead

Symmetric key establishment

- PK cryptography
  - Moderated use recommended
  - Examples:
    - Session keys
    - Shared symmetric key establishment
- Key agreement
  - Both nodes contribute to the shared symmetric key
- Key transport
  - One of the nodes 'chooses' the shared symmetric key

Key agreement

- Authenticated Diffie-Hellman protocol
  - *g* publicly known parameter; *G a* multiplicative group
  - *A* selects a random number r_A in *G;* it calculates $t_A = g^{r_A}$
  - *B* selects a random number r_B in *G;* it calculates $t_B = g^{r_B}$

Alice

Bob

(1) t_A, $E_B(A, n_A)$

$K_0 = h(n_A, n_B)$

(2) t_B, $E_A(n_B)$, $MAC(K_0, t_A, t_B, B, A)$

(3) $MAC(K_0, t_A, t_B, A, B)$

$$K_{AB} = (t_B)^{r_A} = g^{r_B r_A} = (t_A)^{r_B}$$

H. Krawczyk, "SKEME: A versatile secure key exchange mechanism for Internet," NDSS'96

Key transport

Alice

Bob

(1) n_A, B, $E_B(A, K_{AB})$, $Sig_A(n_A, B, E_B(A, K_{AB}))$

(2) n_B, A, n_A, $E_A(B, K_{BA})$, $Sig_B(n_B, A, n_A, E_A(B, K_{BA}))$

(3) n_B, B, $Sig_A(n_B, B)$

$$K_{AB} = f(K_{AB}, K_{BA})$$

X.509 three-pass key transport protocol

Hash chains

- Cryptographic *hash* or *one-way* function
  - h : $\{0,1\}^* \rightarrow \{0,1\}^n$
  - Input: Arbitrary length
  - Output: Fixed length *n*
- Required properties
  - *Collision resistance*: it is computationally infeasible to find two distinct inputs, *x*, *y*, which hash to a common value $h(x)=h(y)$
  - *Pre-image resistance*: given a specific hash-value *z*, it is computationally infeasible to find an input *x* such that $h(x)=z$
  - *2nd pre-image resistance*: given *x* and $h(x)$ it is computationally infeasible to find a second input $y \neq x$ such that $h(y)=h(x)$
  - *Low computational cost*: given *h* and an input *x*, $h(x)$ is easy to compute.

Hash Chains (cont'd)

- Pick a random number *r*
- Generate *k* elements by hashing *r* successively *k* times

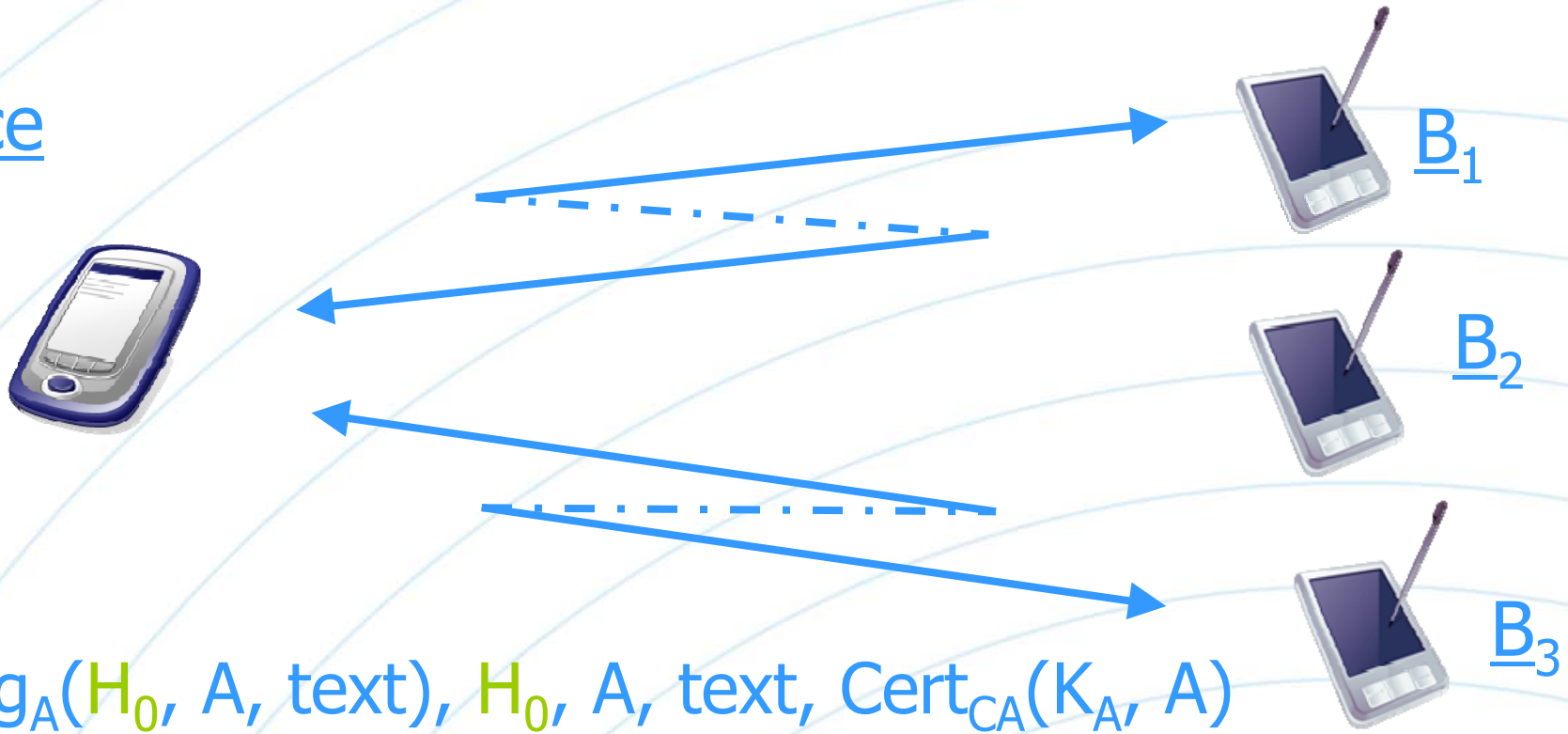
$$h^k(r) \leftarrow h^{k-1}(r) \leftarrow \cdots \leftarrow h^3(r) \leftarrow h^2(r) = h(h(r)) \leftarrow h(r)$$

$$\| \qquad \| \qquad \| \qquad \| \qquad \|$$

$$H_0 \leftarrow H_1 \leftarrow \cdots \leftarrow H_{k-3} \leftarrow H_{k-2} \leftarrow H_{k-1}$$

- H_0 is the hash chain *anchor*
- The remaining *k-1* elements can be used for authentication

Bootstrapping a hash chain



- Alice must 'commit' to the hash chain anchor
- Each B_i node validates the commitment (signature) and stores H_0
- Alice can then utilize the hash chain elements

Using a hash chain

- Chain elements as authenticators, e.g., to transmit "yes" / "no"
  - "Yes" chain

$$H_0 \leftarrow H_1 \leftarrow \cdots \leftarrow H_{k-3} \leftarrow H_{k-2} \leftarrow H_{k-1}$$

  - "No" chain

$$G_0 \leftarrow G_1 \leftarrow \cdots \leftarrow G_{k-3} \leftarrow G_{k-2} \leftarrow G_{k-1}$$

- Sender : 'Reveal' elements in this order
  - Use G_i or H_i to authenticate a "no" or "yes"
- Receiver: For the i – th message from Alice, verify that $h^i(H_i) = H_0$ *or* $h^i(G_i) = G_0$

R. Hauser, A. Przygienda, and G. Tsudik, "Reducing the Cost of Security in Link-State Routing," NDSS'96

Using a hash chain (cont'd)

- Chain elements as symmetric keys

$$H_0 \leftarrow H_1 \leftarrow \cdots \leftarrow H_{k-3} \leftarrow H_{k-2} \leftarrow H_{k-1}$$

Time T_i: m_i = A, $text_i$, MAC(H_i, A, $text_i$)

Time T_{i+j}: Release H_i

  - Synchronized clocks at sender and receiver
  - Sender release keys (e.g., flooding them across the network) at specific intervals
  - *A posteriori* validation at the receiver: reject messages not generated sufficiently close to the release time

S. Cheung, "An Efficient Message Authentication Scheme for Link State Routing," Comp. Sec. App. Conf. '97

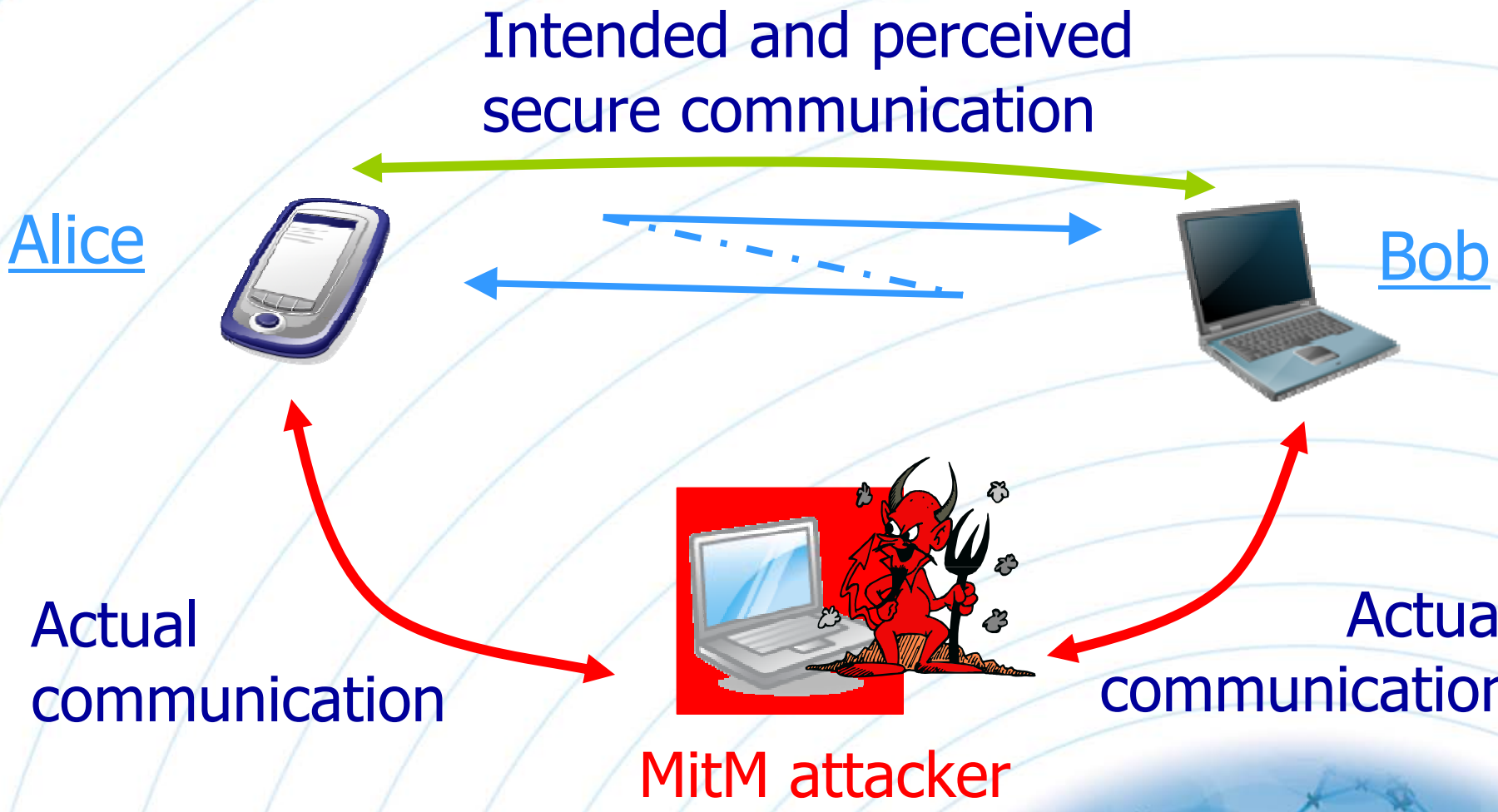
A. Perrig et al., "Efficient and secure source authentication for multicast," NDSS '01

Recap: Public key enabled security

- Advantages
  - Any-to-any secure communication
  - Basis for bootstrapping symmetric key primitives
- Disadvantages
  - Processing and communication overhead
  - Setting up a certification authority
- Comment
  - Methods discussed so far are rather 'agnostic' to the underlying network technology

What if no CA is available?

- **Main challenge:** *Man-in-the-Middle* attacks



What if no CA is available? (cont'd)

- Can we leverage on characteristics of the network or the mobile application?
- **Observation 1:** Wireless, mobile devices are used by human beings, who can assist the security association establishment
- **Observation 2:** Wireless communication possible only within a very short range or within a line of sight can imply that no other device is present (**caution!**)

Leveraging on the users

- Password-based key establishment
  - π : shared password
  - $g_\pi = \left(h(\pi)\right)^2 \bmod q$
  - *q* publicly known parameter
  - *A, B* select random numbers *x, y respectively*

Alice

Bob

$w_A = g_\pi^x \bmod q$

(1) w_A

$w_B = g_\pi^y \bmod q$

(2) w_B

$z = w_B^x \bmod q$

$z = w_A^y \bmod q$

Abort if z is not in range [2, q–2]
Otherwise, z is a large shared secret

D. Jablon, "Extended Password Key Exchange Protocols Immune to Dictionary Attacks," WET-ICE '97

Leveraging on the users (cont'd)

- Password-based key establishment
  - h : hash function
  - Once *z* is established, A and B prove to each other they know the same *z*
  - A and B can then derive a session key from *z*

Alice

Bob

$o_A = h(04 \| w_A \| w_B \| z \| g_\pi)$ $o_B = h(03 \| w_A \| w_B \| z \| g_\pi)$

(3) o_B

(4) o_A

$o_3 = h(03 \| w_A \| w_B \| z \| g_\pi)$ $o_4 = h(04 \| w_A \| w_B \| z \| g_\pi)$

Abort if $o_B \neq o_3$ or $o_A \neq o_4$

Leveraging on the user (cont'd)

- The user verifies that the keys generated at the two devices are identical
- Visual and audible hashes

J. McCune, A. Perrig, and M. Reiter, "Seeing-is-Believing: Using Camera Phones for Human-Verifiable Authentiction." S&P'05

M. Goodrich, M. Sirivianos, J. Solis, G. Tsudik, and E. Uzun, "Loud And Clear Human-Verifiable Authentication Based on Audio", ICDCS'06

Leveraging on the wireless link

- 'Off-line' local channels
  - One example: infra-red
    - D. Balfanz, D. Smetters, P. Stewart, and H. Wong, "Talking to strangers: Authentication in ad-hoc wireless networks," NDSS'02
  - Exchange information over the local channel that allows you to authenticate over the wireless radio channel
- Caution: System and protocol design must ensure that it is indeed impossible for the attacker to interfere actively with the communication over the local channel
  - For example, the attacker must be unable to act as an 'invisible' relay

Leveraging on the network

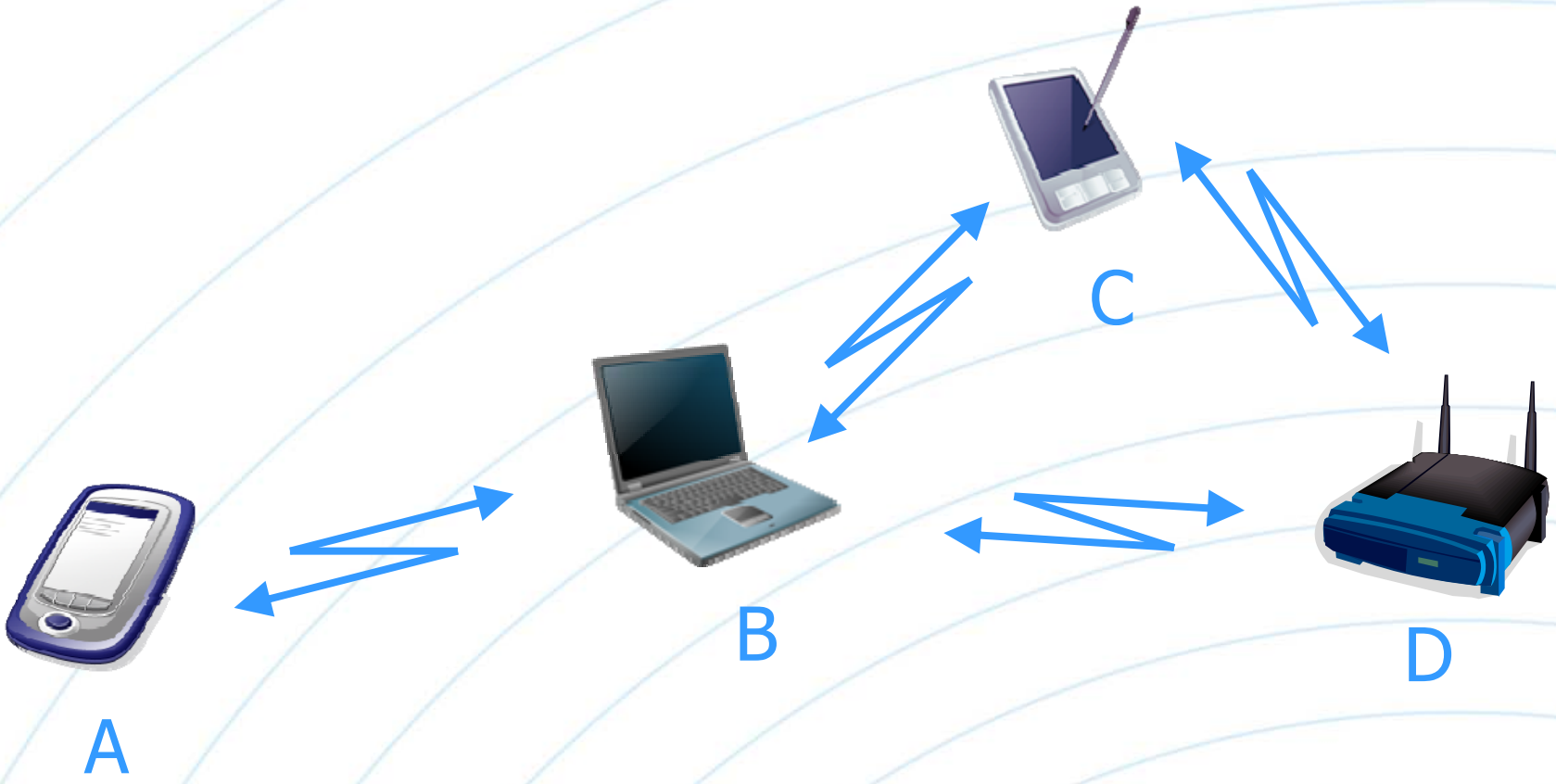
- Mobility
  - Users meeting each other, e.g., at a conference, can set up symmetric keys or exchange public keys
    - S. Capkun, J-P. Hubaux, and L. Buttyan, "Mobility helps security," ACM Mobihoc'03
  - More generally, a mobile device can be interested in obtaining public keys of other devices in proximity, e.g., within a few hops
    - Example later in secure routing
  - Point of caution: communication pattern

Summary

- One-to-one, one-to-many, one-to-all, any-to-any secure communication
- Need for protocols that allow dynamic establishment of security associations
  - Public-key cryptography
  - Symmetric-key cryptography
  - Leveraging on the communication and computing environment characteristics
- Various communication patterns
  - Duration, number of communicating devices, direction of communication
- Additional readings
  - Sensor network key distribution

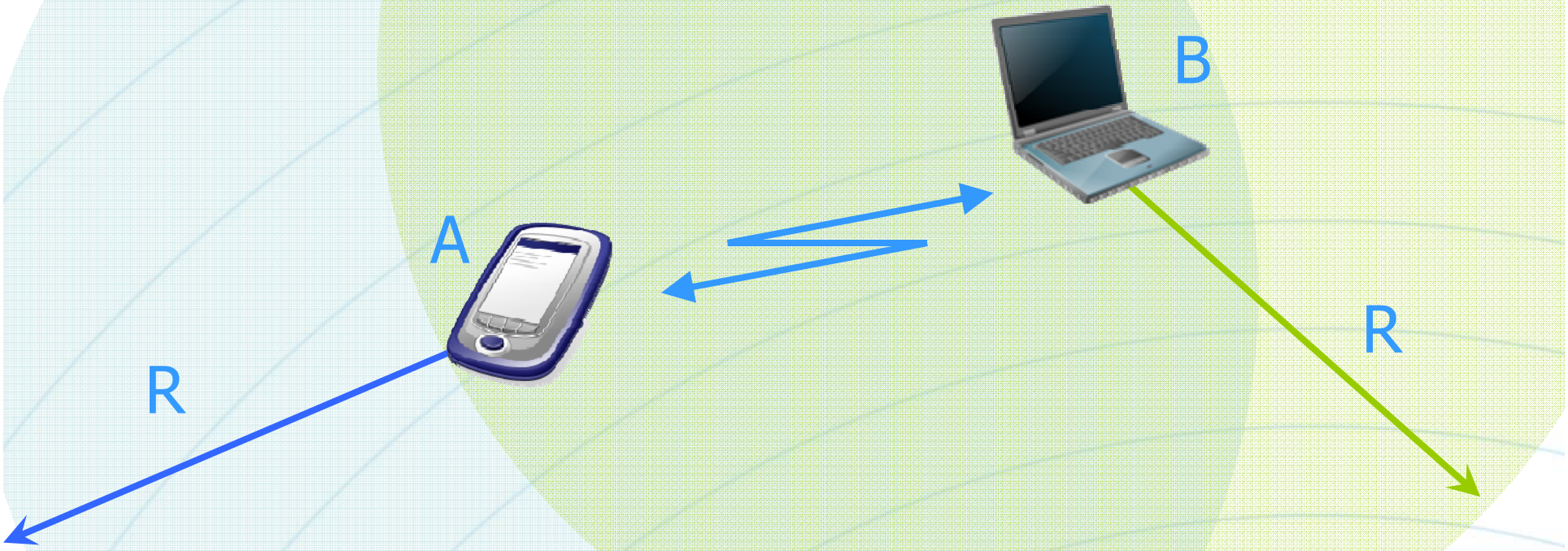
Secure Neighbor Discovery

Problem statement



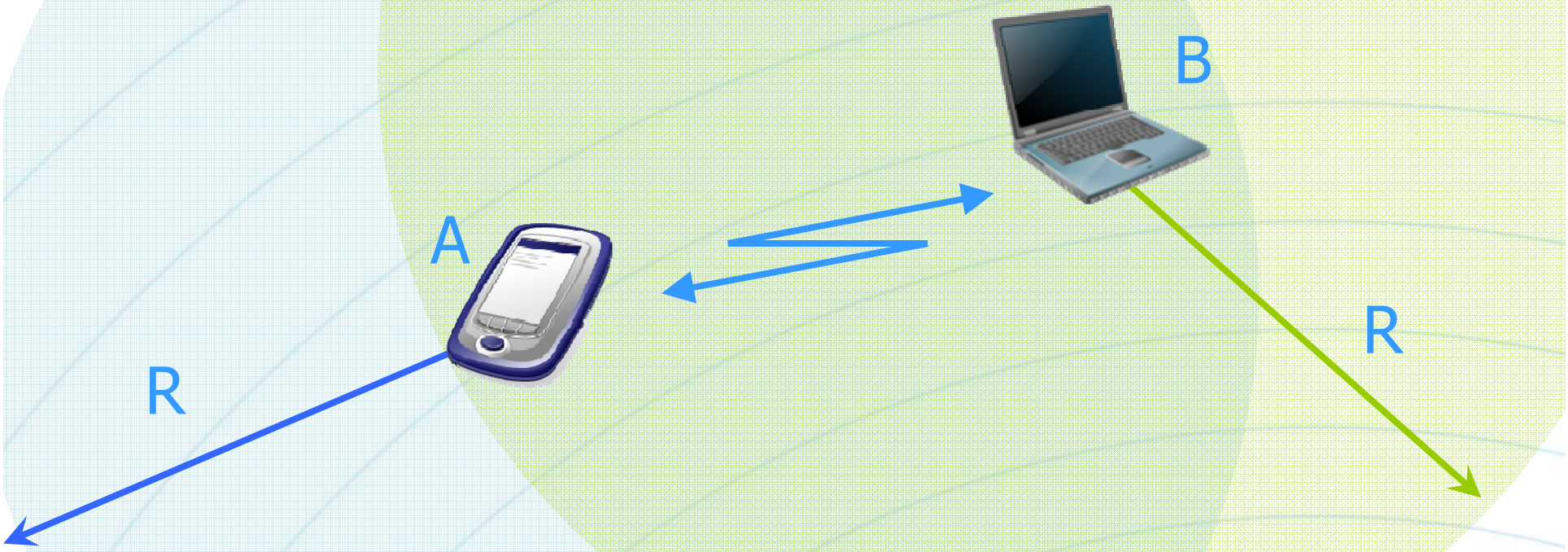
- Node discovery
  - A node discovers other nodes it can *directly* communicate with

Problem statement (cont'd)



- R: nominal communication range
- Caution: A, B are neighbors *if and only if* they can communicate *directly*

Problem statement (cont'd)

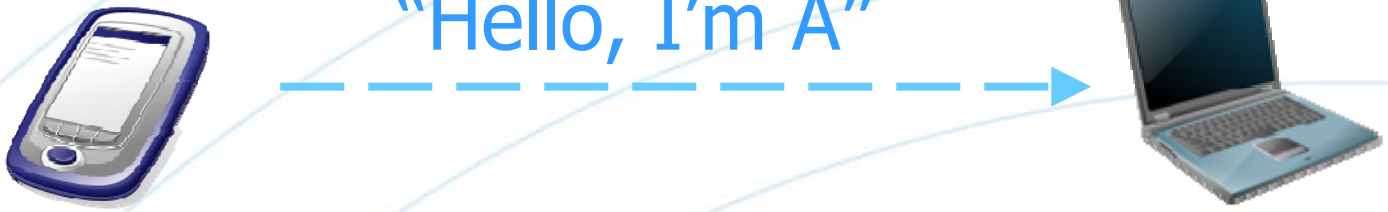


- B is neighbor of A if and only if it can receive directly from A
- Link (A,B) is *up* ⇔ B is neighbor of A
- Consider the case with different nominal communication ranges, e.g., R_A, R_B; then (A,B) may be *up* while (B,A) is *down*

Neighbor discovery

- Neighbor discovery is a building block for other system functionality
  - Communication
  - Access control
  - Physical access control
- Examples
  - First step before routing
  - Connection to an wireless LAN access point
  - Radio Frequency Identification (RFID) reader controlling a door

Neighbor discovery (cont'd)



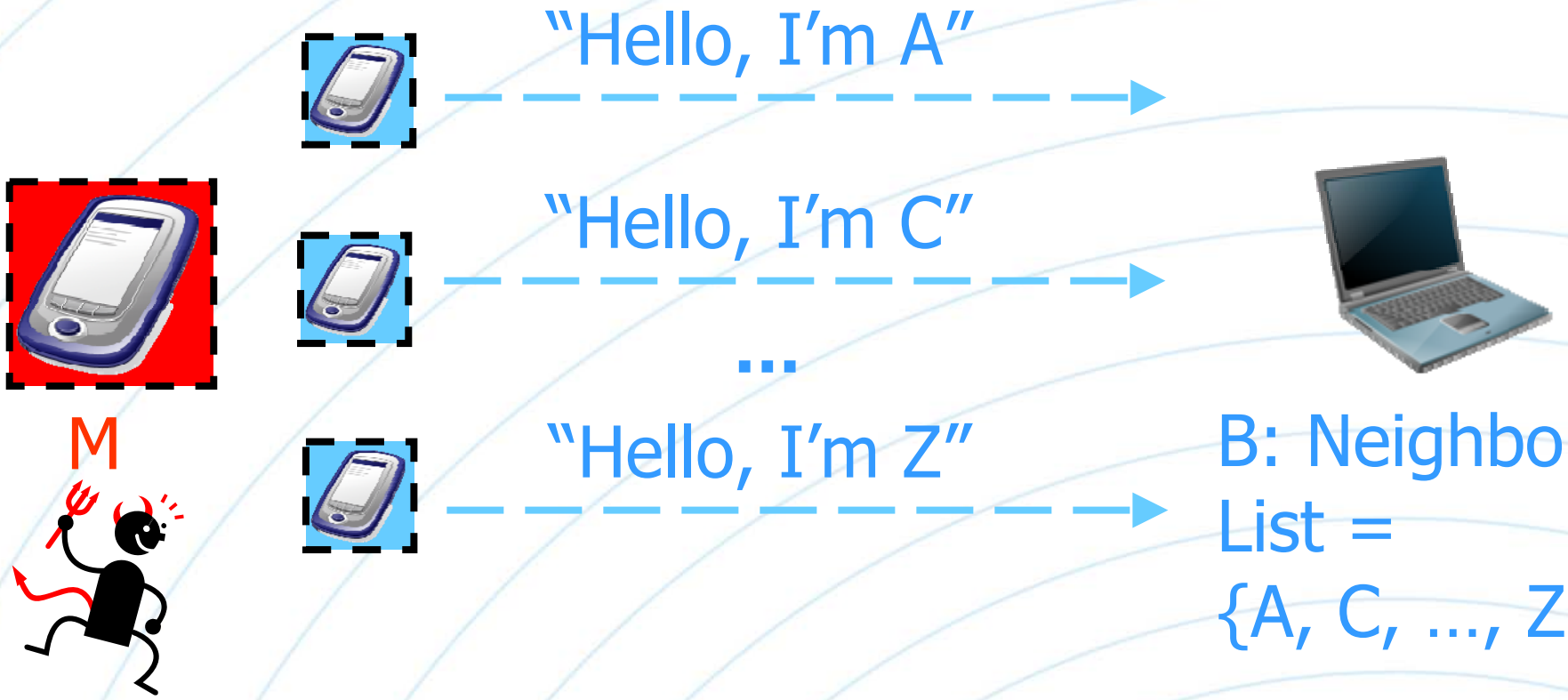
A

B

B: "A is my neighbor"; "A is added in my Neighbor List"

- Simple, widely used solution, but **not** secure
- Easy to mislead B that A is its neighbor when this is not the case

Attacking neighbor discovery



- Single adversary appears as multiple neighbors

Securing neighbor discovery

A

"Hello, I'm A", Sig_A("Hello, I'm A"), $Cert_{CA}(K_A,A)$

B

(1) Validate $Cert_{CA}()$
(2) Validate $Sig_A()$
(3) Add A to neighbor iff (1), (2) are successful

- A first attempt
  - Authenticate "Hello" messages
- The adversary can record signed "Hello" messages and transmit (replay) them later

Securing neighbor discovery (cont'd)

A

(1) n_B, B

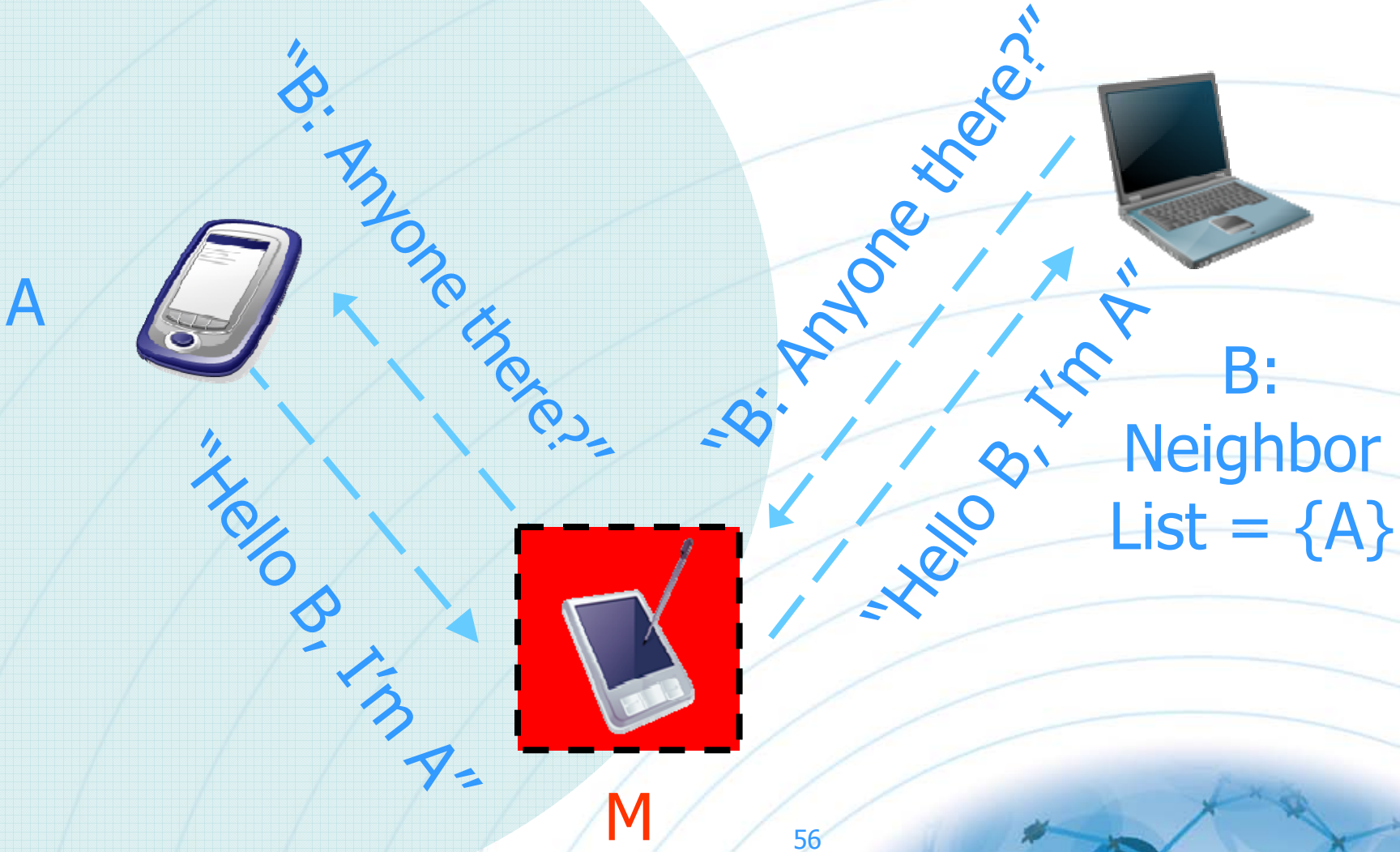
B

(2) A, n_A, n_B, B, $Sig_A(A, n_A, n_B, B)$, $Cert_{CA}(K_A, A)$

- A second attempt
  - Message authenticity and replay protection
    - n_A, n_B are nonces
  - Bob essentially 'challenges' Alice to provide a 'hello' message

Attacking neighbor discovery (cont'd)

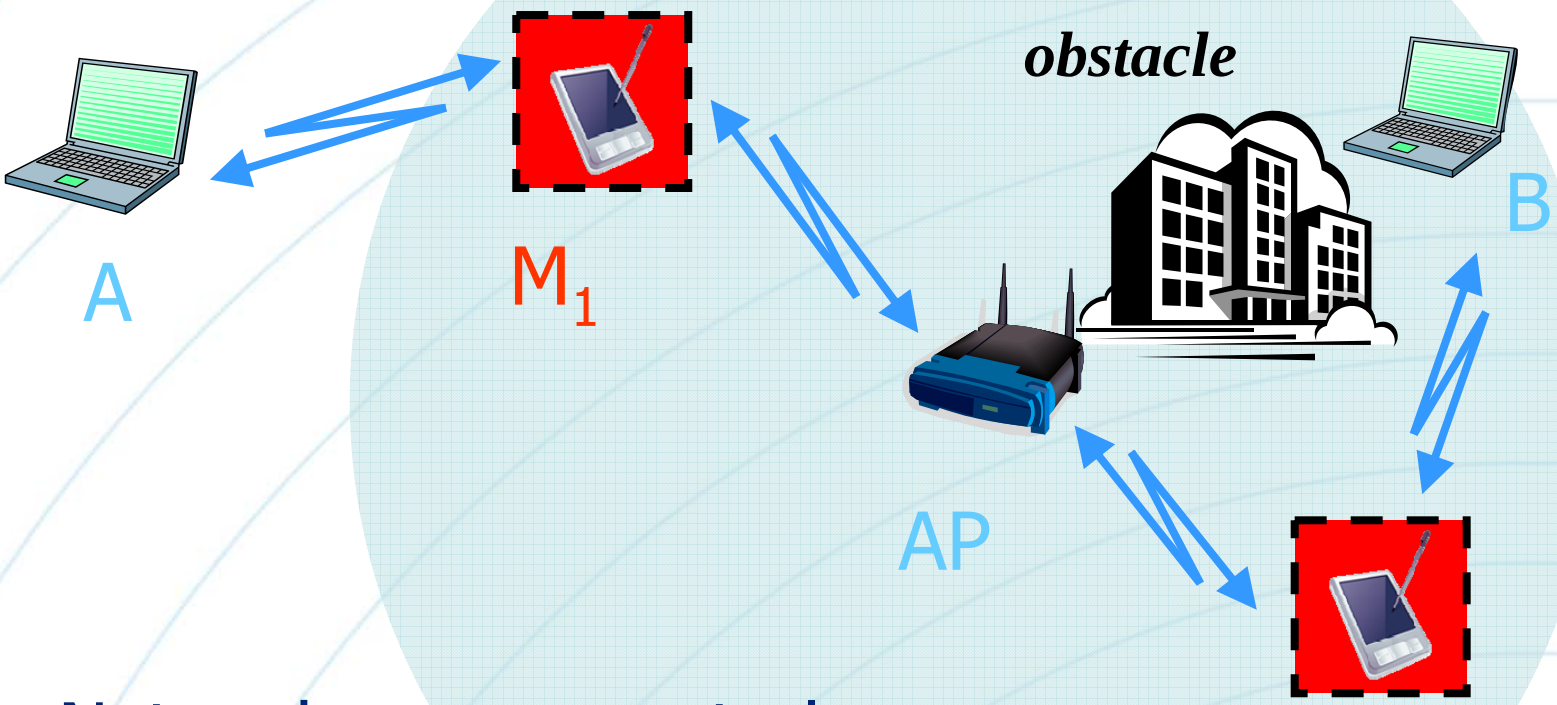
- "Relay" or "Wormhole" Attack
  - Simply relay any message, without any modification



Attacking neighbor discovery (cont'd)

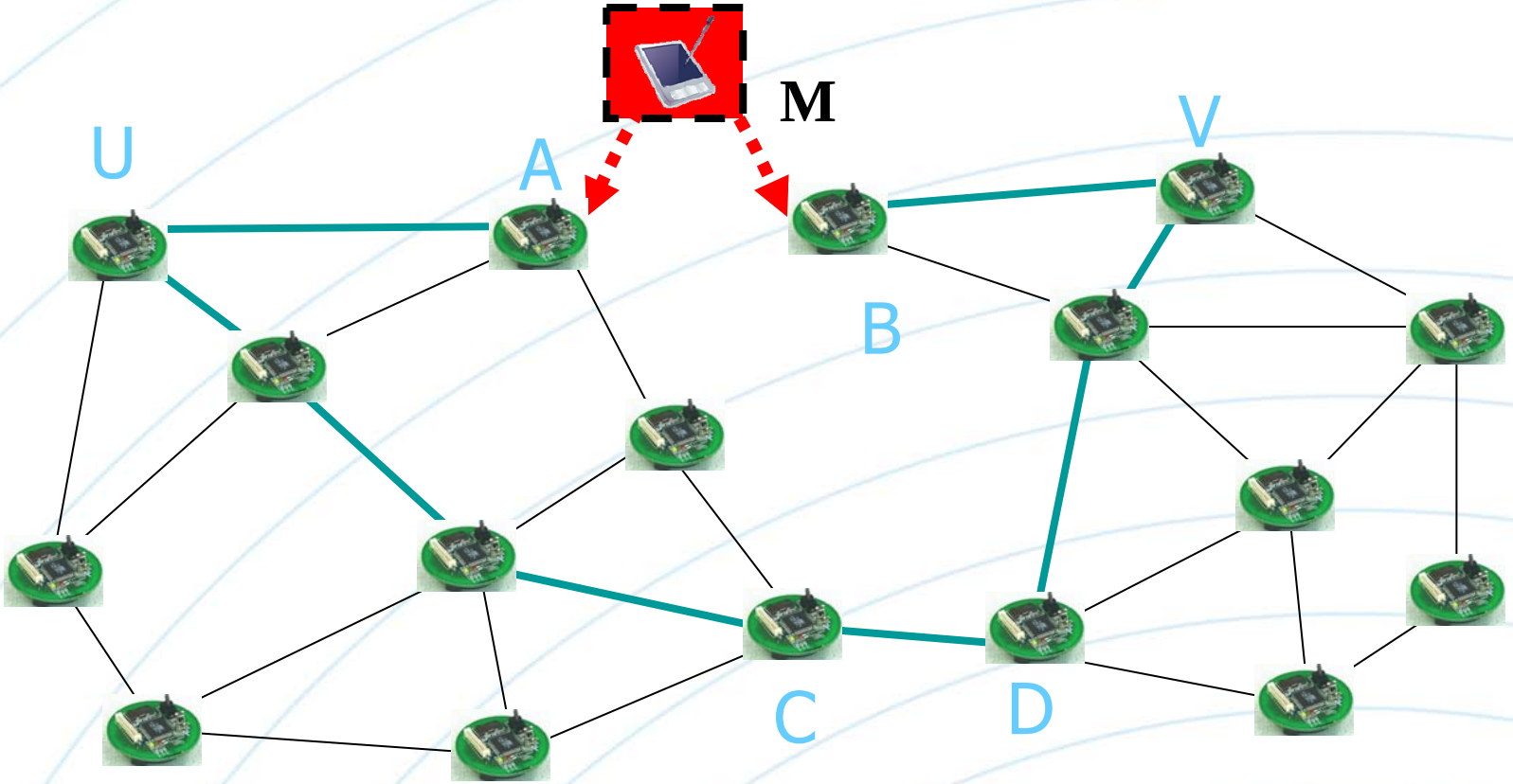
- Long-range relay / wormhole
  - The attacker relays messages across large distances

Attack implications



- Network access control
  - The attacker 'assists' access
  - But it has control over the nodes' communication

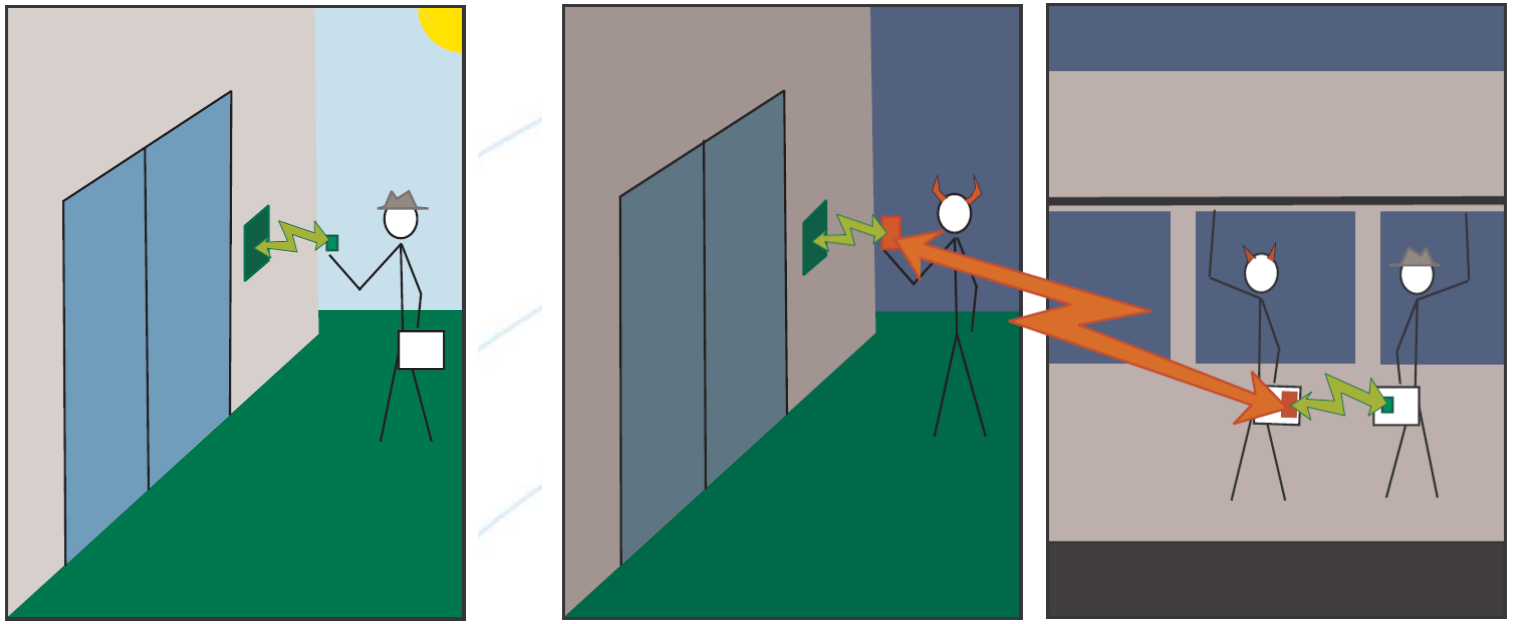
Attack implications (cont'd)



- Routing
  - The attacker creates a 'link' and 'provides' shortest routes
  - Attracted traffic is under the control of the adversary

Attack implications (cont'd)

Illustration by M. Poturalski



- Physical access control
  - RFID based access control
  - Attacker close to the owner of the access-granting RFID tag; relays signals from and to her accomplice, who obtains access

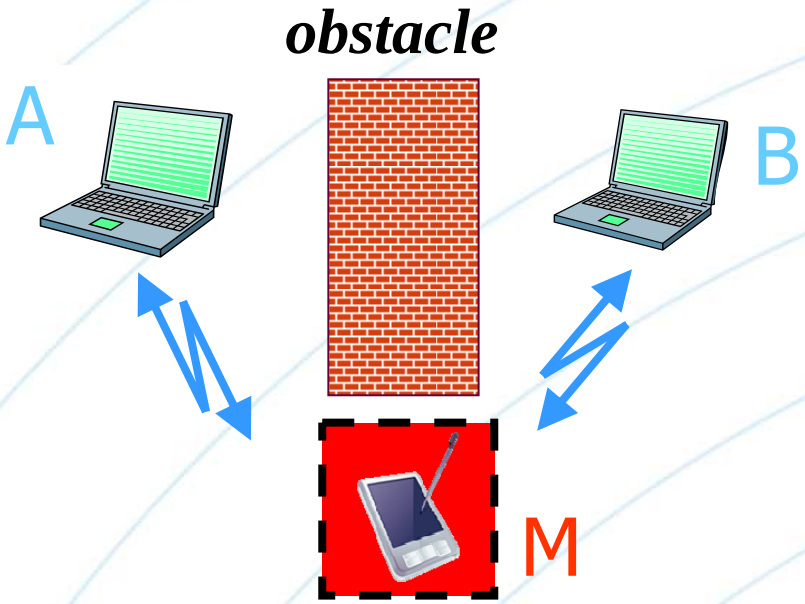
Z. Kfir and A. Wool, "Picking virtual pockets using relay attacks on contact-less smartcard," SECURECOMM '05

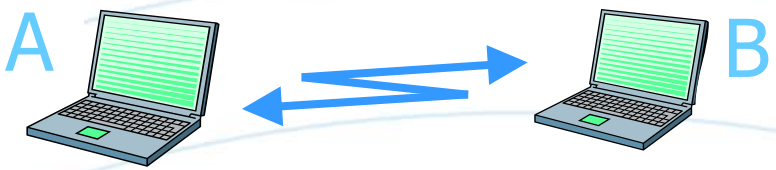
Securing neighbor discovery (cont'd)

- A third attempt
  - Geographical packet leashes
    - Nodes are aware of their location in a secure manner
    - Loosely synchronized clocks
    - Sender adds coordinates to each packet
    - Receiver checks if sender is within range
  - Temporal packet leashes
    - Nodes have tightly synchronized clocks
    - Sender (A) adds a timestamp to each packet
    - Receiver (B) estimates its distance from the sender based on the elapsed time, $t_{prop} = t_{receiveB} - t_{sendA}$
    - $Dist(A,B) < ct_{prop}$
      - c is the speed of light
      - 'Ignore' the clock drift

Y-C. Hu, A. Perrig, and D. Johnson, "Packet Leashes: A Defense against Wormhole Attacks in Wireless Ad Hoc Networks," Infocom'03

Attacking neighbor discovery (cont'd)



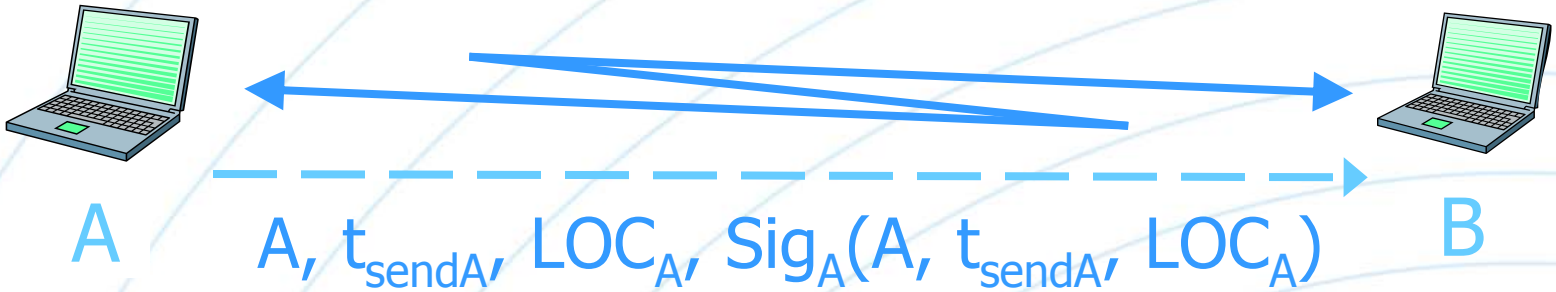


- **Observation:** *Physical proximity* does **not** necessarily imply correct nodes are *able to communicate directly*
- **No** protocol using time-of-flight measurements can distinguish the two situations

M. Poturalski, P. Papadimitratos, and J-P. Hubaux, "Secure Neighbor Discovery: Is it Possible?" LCA-REPORT-2007-004, 2007

Securing neighbor discovery (cont'd)

- Location-aware nodes (securely)
- Estimate neighbor distance in two ways
  - Based on the time-of-flight (ToF)
  - Based on the location information (LOC)
- Compare the two distance estimates

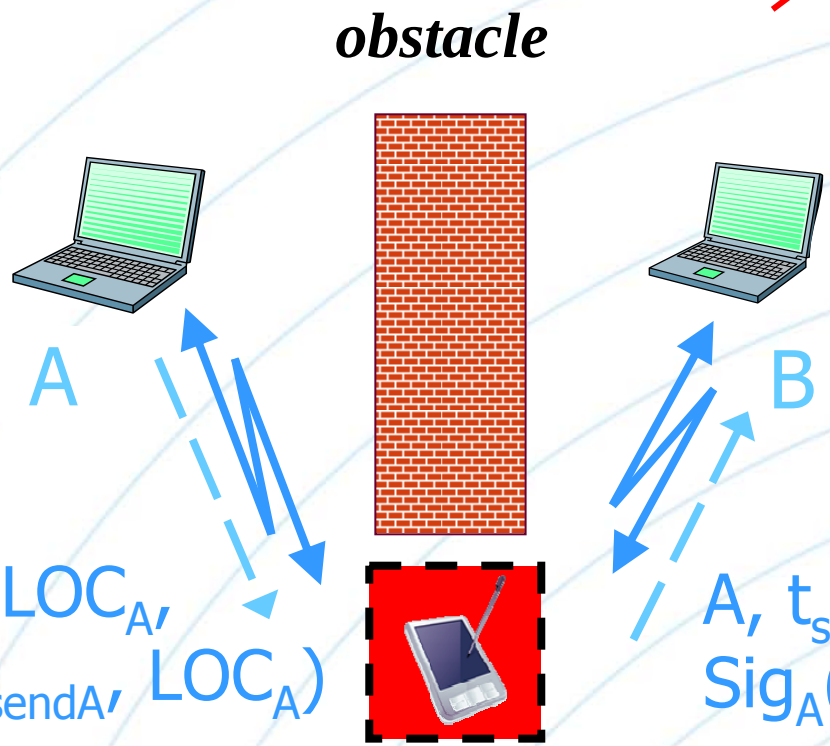


B: Dist(LOC_A, LOC_B) = $Dist_{Estimate}(t_{sendA}, t_{receiveB})$

B: "Add A to neighbor list"

Securing neighbor discovery (cont'd)

- *Secure Neighbor Discovery*: exchange location information, and compare ToF and LOC based distance estimates



✗ B: $Dist(LOC_A, LOC_B) < Dist_{Estimate}(TS_A, t_{receiveB})$

B: "Do NOT add A to neighbor list"

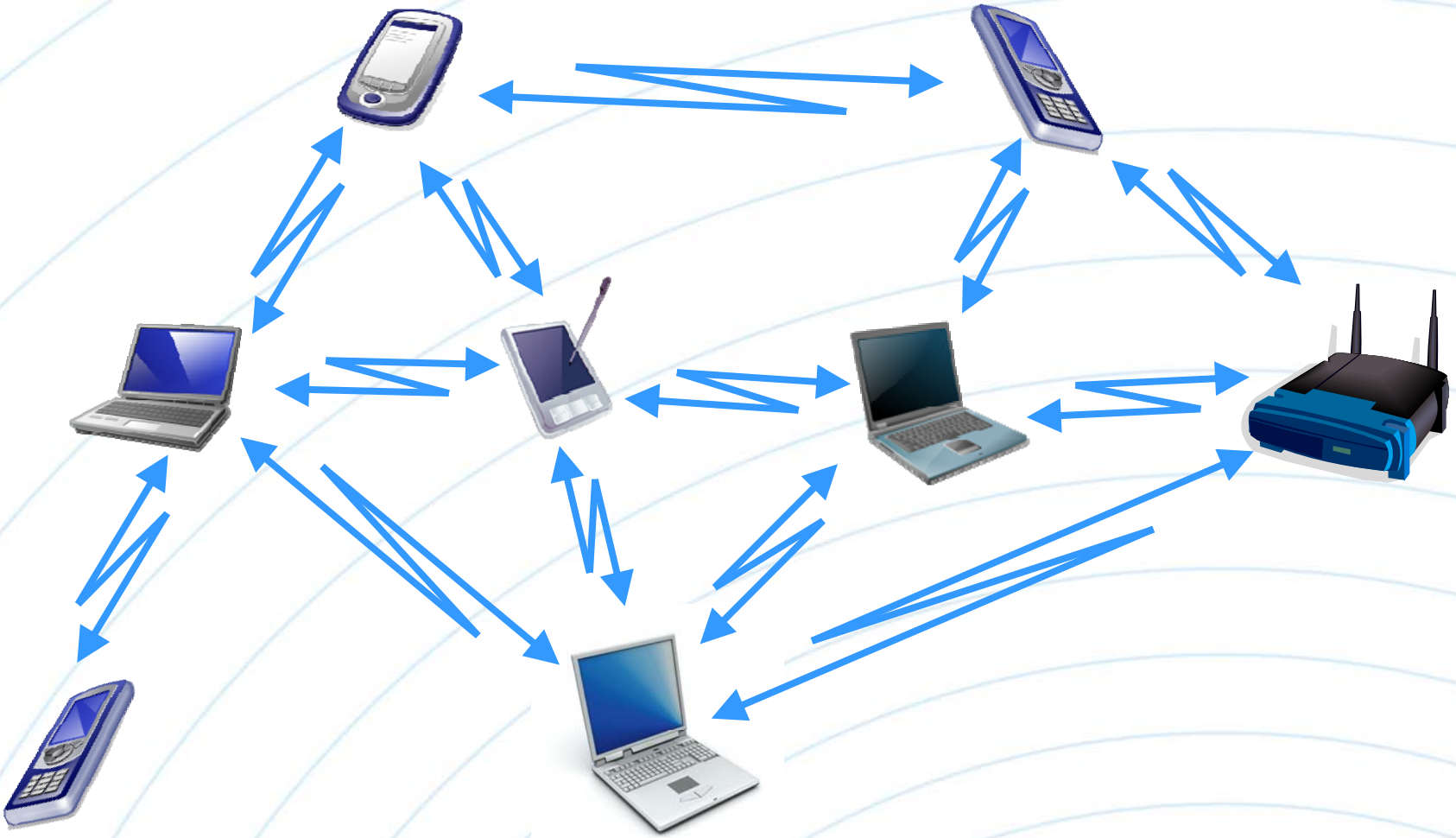
M. Poturalski, P. Papadimitratos, and J-P. Hubaux, "Secure Neighbor Discovery: Is it Possible?" LCA-REPORT-2007-004, 2007

Summary

- Secure Neighbor Discovery
  - Solution for
  - Hard problem; solution is not easy to implement in practice
  - Prerequisite for secure networking protocols and system security
  - Additional reading
    - Other methods, surveyed in [Poturalski-Papadimitratos-Hubaux] report: Using distance bounding, directional antennas, knowledge of topology, properties of the radio signal
    - Centralized visual and statistical wormhole detection

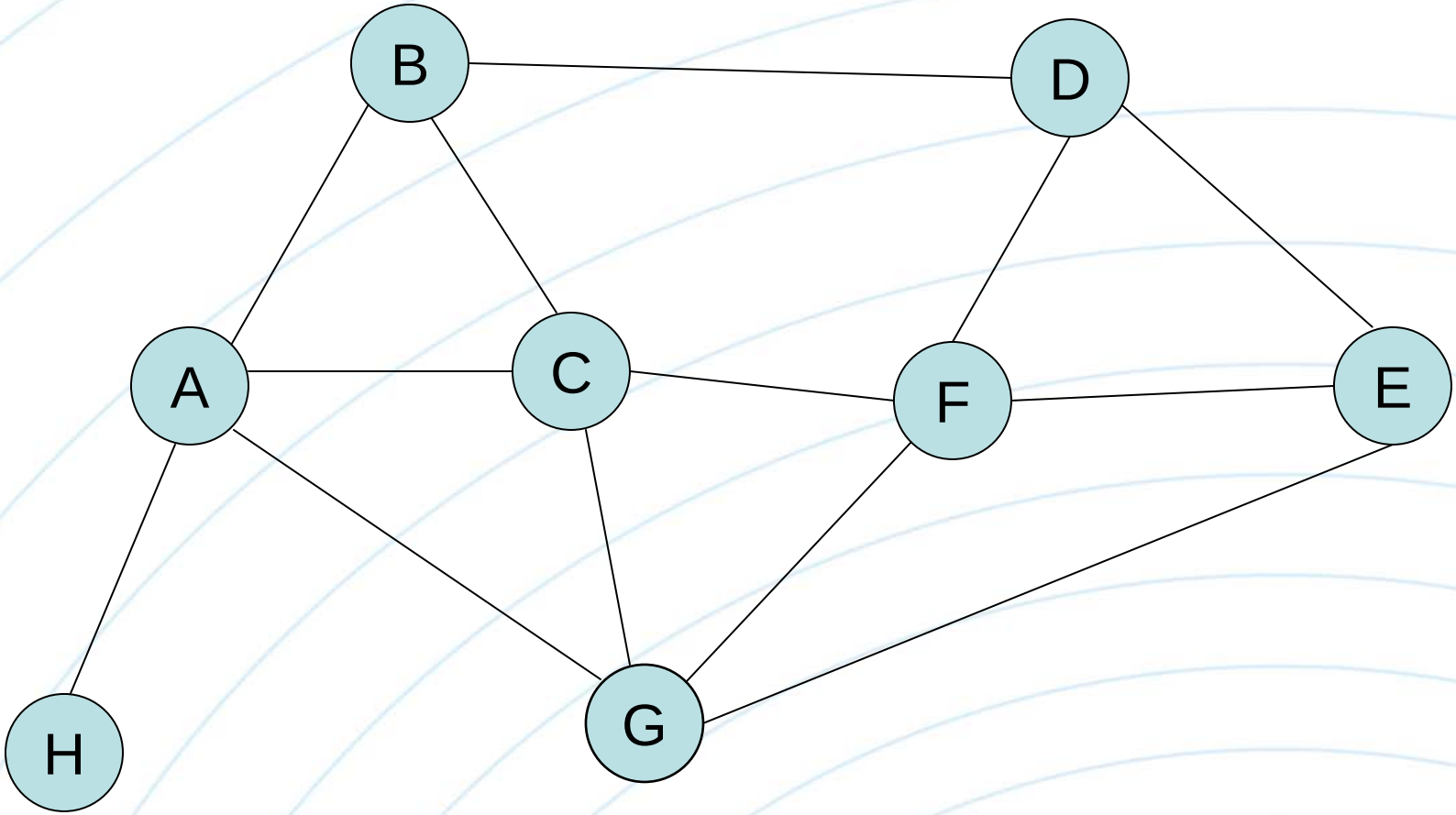
Secure Route Discovery

Multi-hop routing



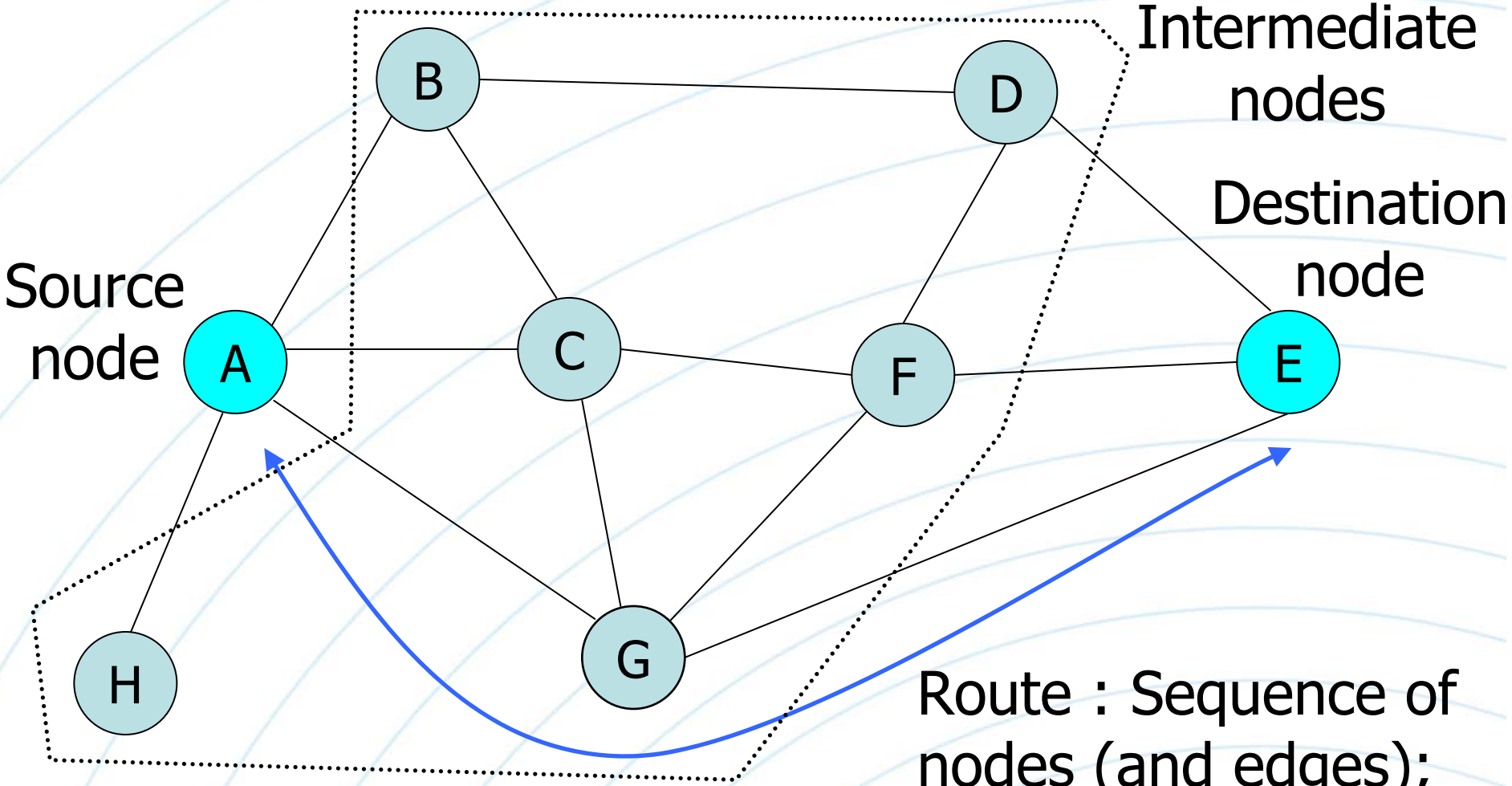
- Wireless multi-hop connectivity

Multi-hop Routing (cont'd)



- (Multi-hop) Connectivity graph

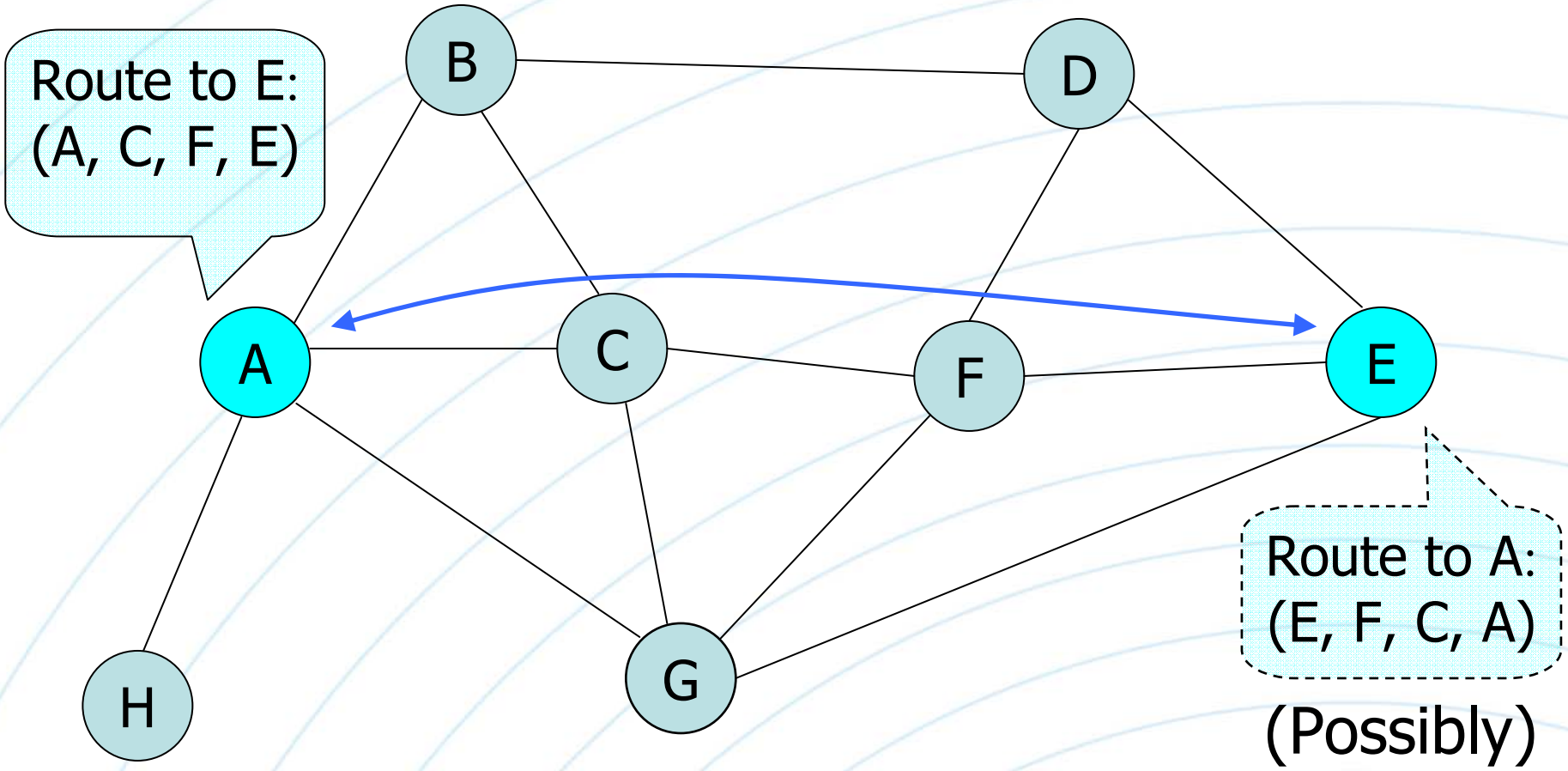
Multi-hop routing (cont'd)



- Stage 0: neighbor discovery
- Stage 1: route discovery

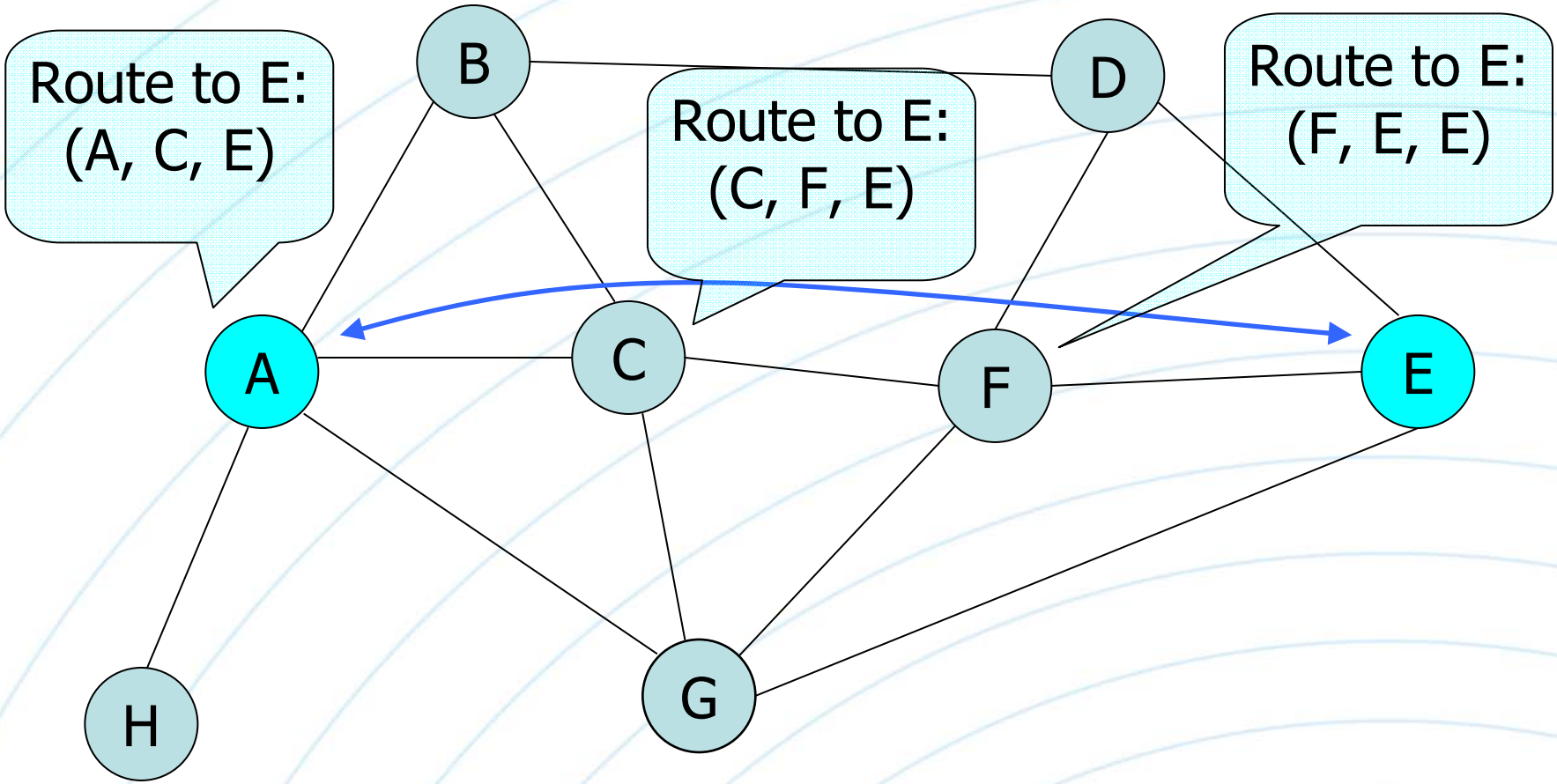
Route : Sequence of nodes (and edges); for simplicity: (A, G, E)

Multi-hop routing (cont'd)



- Explicit route discovery
  - Fully, clearly expressed and readily observable route returned by the routing protocol

Multi-hop routing (cont'd)

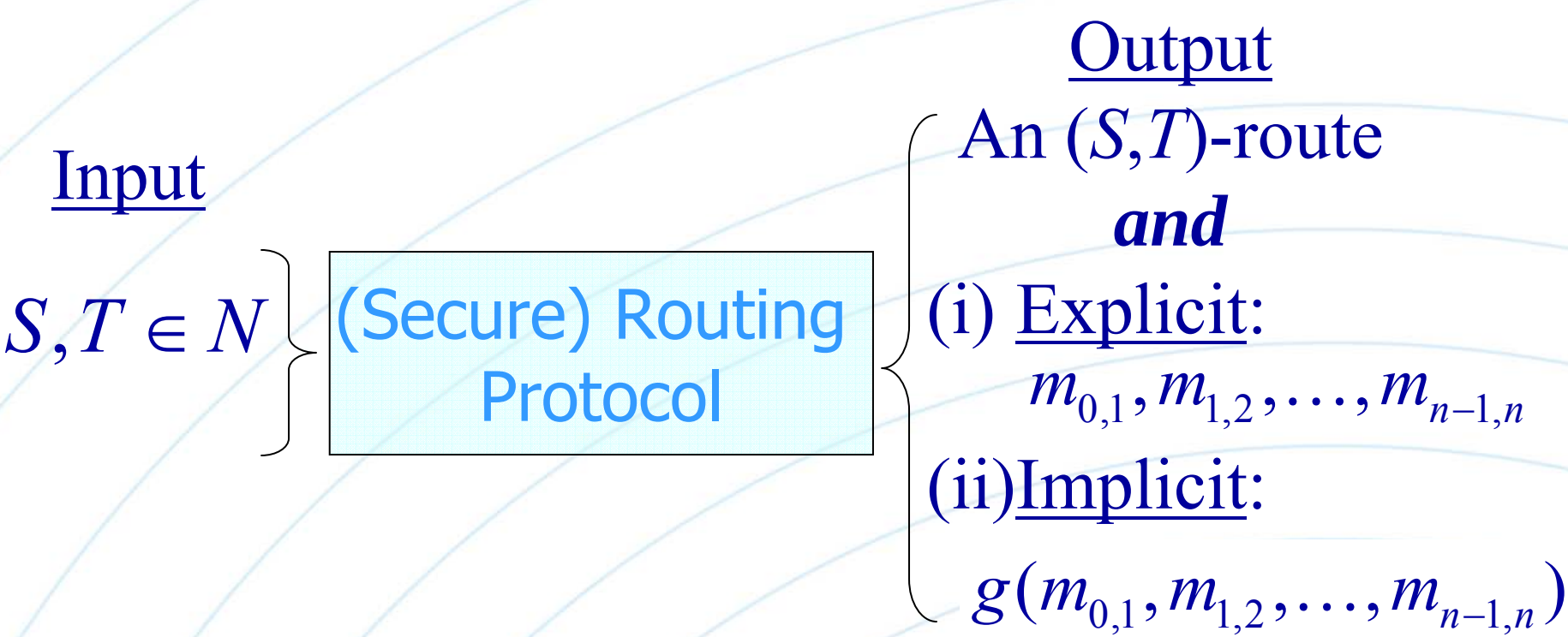


- Implicit route discovery
  - Distributed computation that returns a tuple of the form (current node, relay node, destination node)

Multi-hop routing (cont'd)

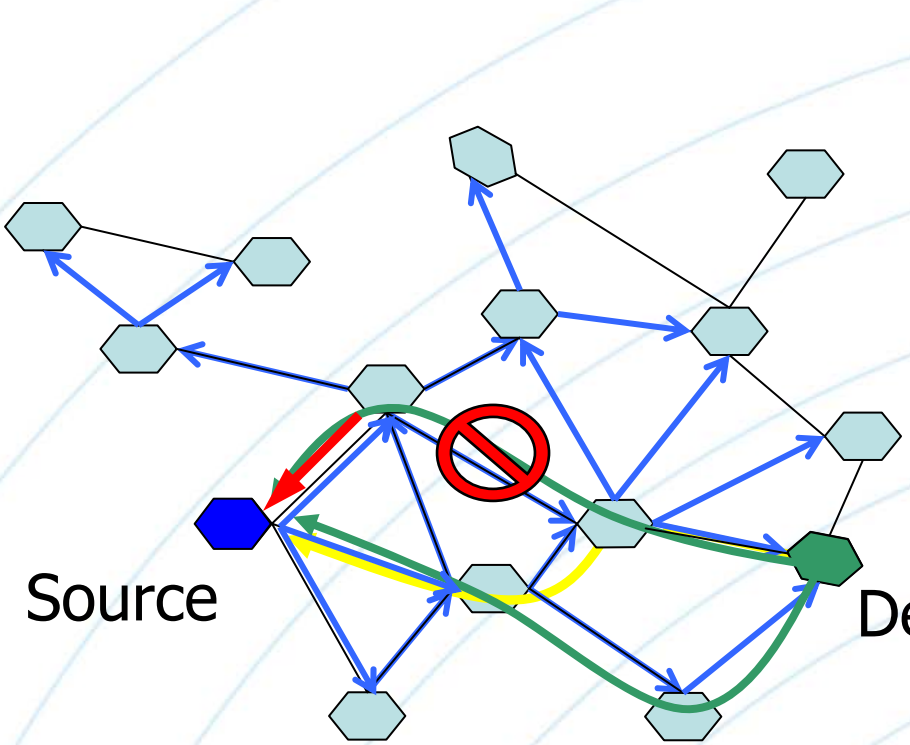
- Basic route discovery
  - Explicit or implicit, providing only the structure of the route
- Augmented route discovery
  - Need a function that assigns labels to links, denoted as link metrics
    - For a link (V_1,V_2), metric $m_{1,2}$
  - Route metric: a function that is the aggregate of the route link metrics
    - For a route $(V_0, V_1, \ldots, V_n)$, route metric $g(m_{0,1}, m_{1,2},\ldots, m_{n-1,n})$

Multi-hop routing (cont'd)



- Input: source, S, and destination, T, nodes
- Output: an (*S*-*T*)-route (of *n* links) and
  - The link labels (metrics) or
  - The route metric

Multi-hop routing (cont'd)



Route Request

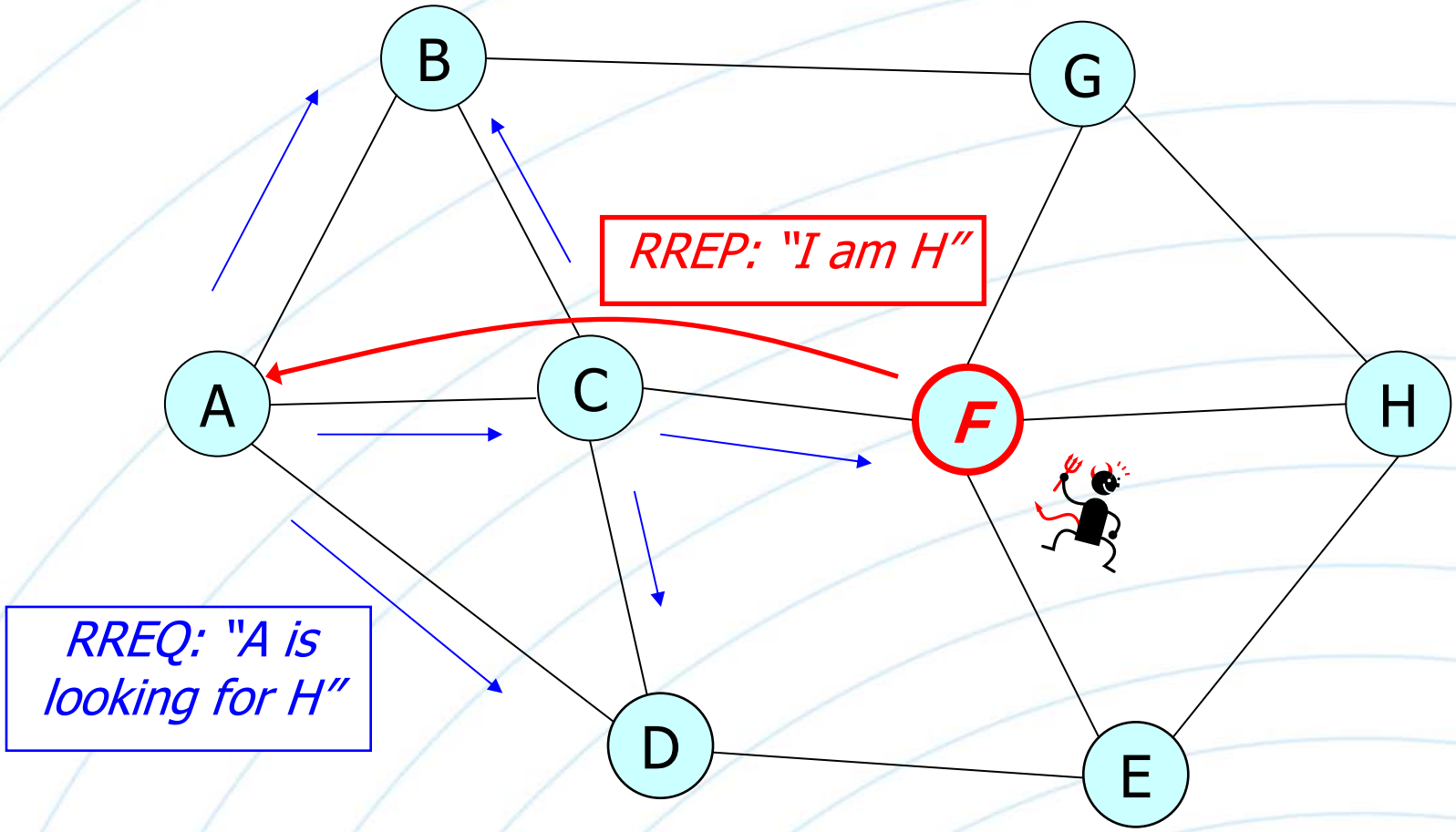
Route Reply I

Route Reply II

Route Error

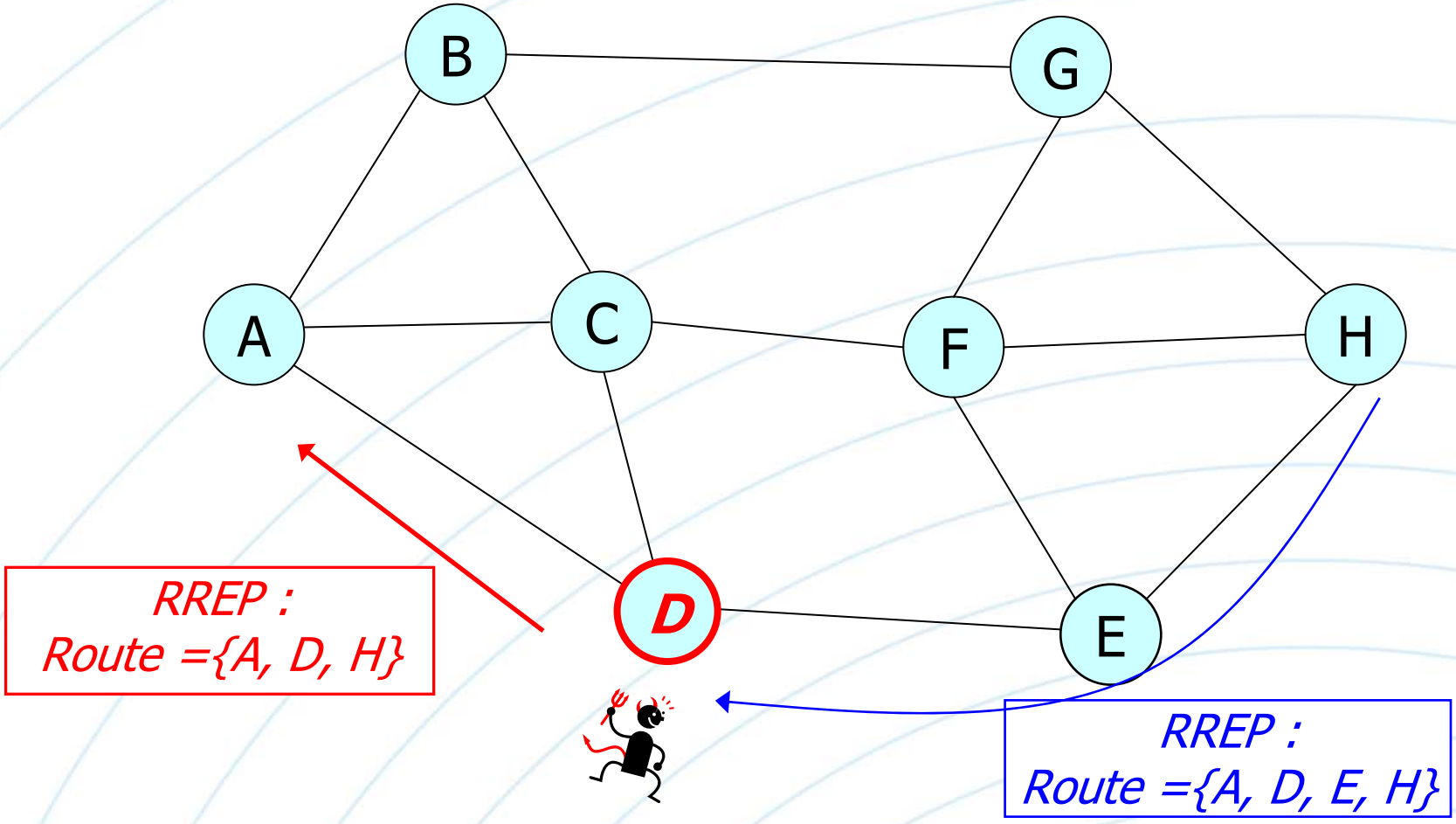
- Example of route discovery
  - Reactive routing protocol

Attacking route discovery (cont'd)



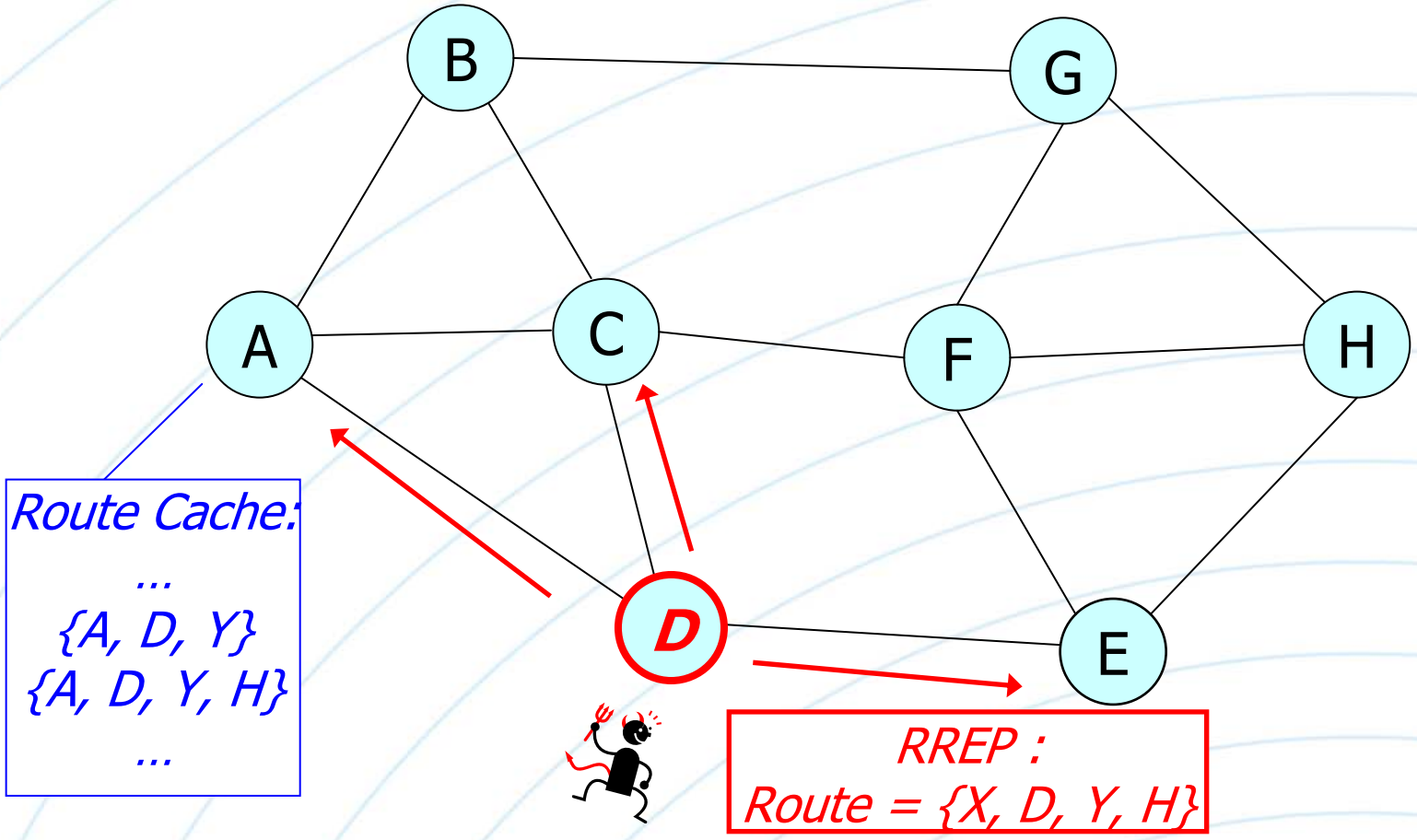
- Impersonation of the destination, for example, in any reactive routing protocol

Attacking route discovery (cont'd)



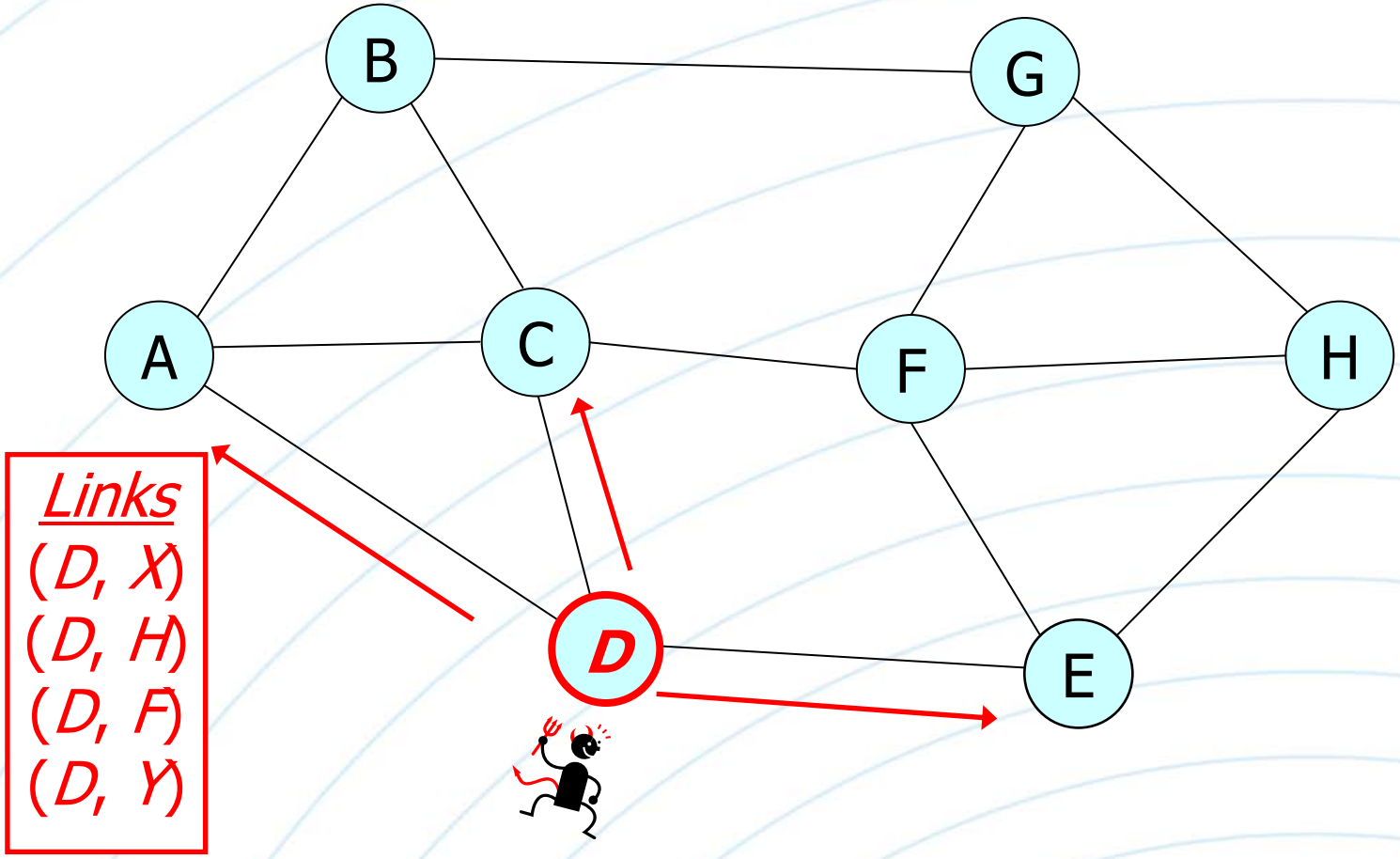
- Modification of the route links, for example, in DSR

Attacking route discovery (cont'd)



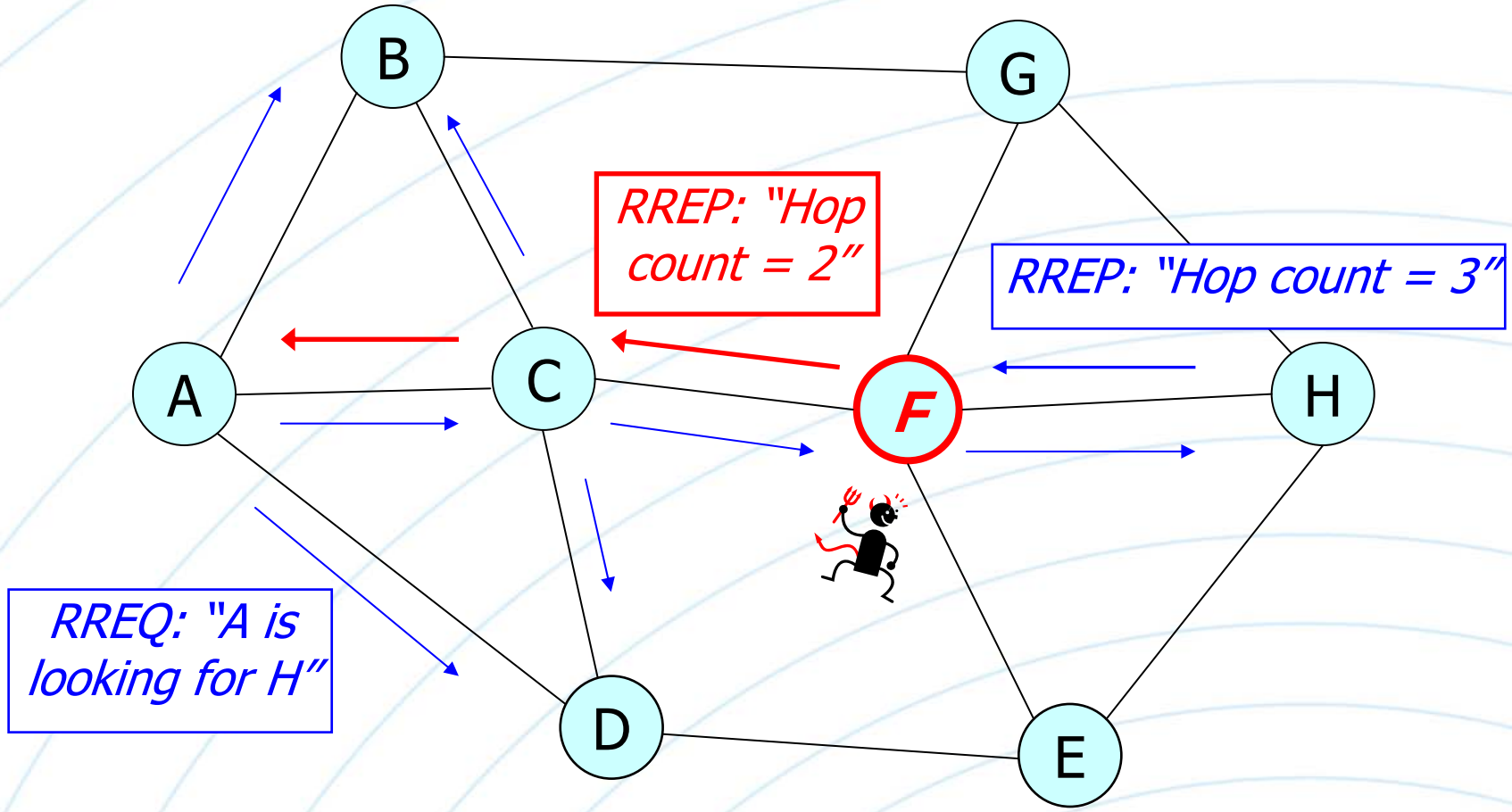
- Abuse of the routing caching mechanism, for example, in DSR

Attacking route discovery



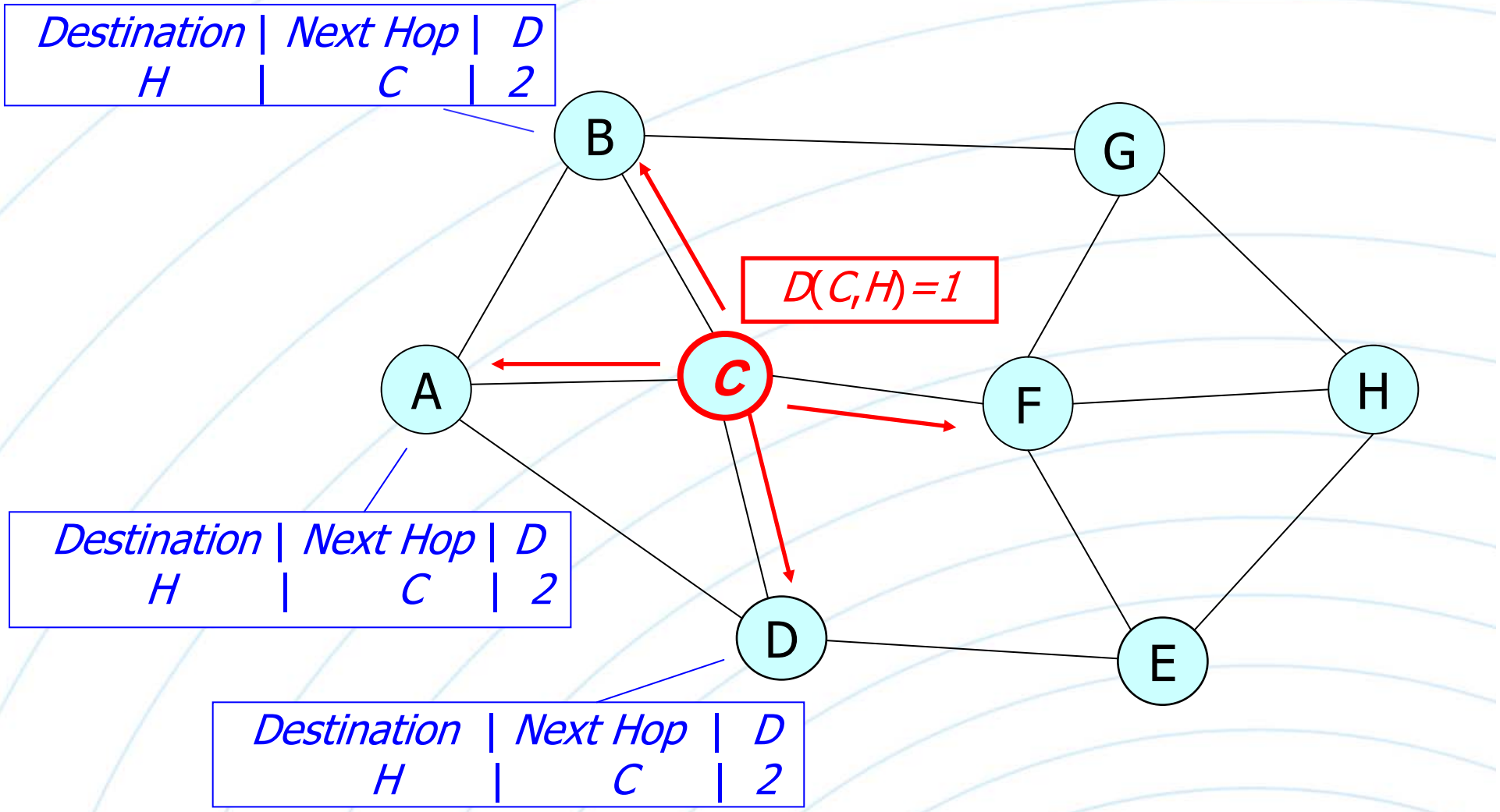
- Disrupting a link state routing protocol, for example, in *OLSR*

Attacking route discovery (cont'd)



- Disrupting distance vector routing, for example, in *AODV*

Attacking route discovery (cont'd)



- Disrupting distance vector routing, for example, in *DSDV*

Attacking route discovery (cont'd)

- Caution: none of the above-mentioned protocols (DSR, AODV, DSDV, OLSR) was designed with security in mind
- Many possible ways to attack the route discovery
- Outcome of attacks
  - Control communication
    - Become part of utilized routes
    - Monopolize resources
  - Disrupt communication
    - Degrade or deny

Secure route discovery requirements

- What do we need a secure routing protocol to do?
- Network model
  - Capture the system characteristics
    - For example, dynamically changing topology
- Specification
  - Define the properties of any candidate secure routing protocol independently of its functionality

Requirements

- We are interested in protocols that discover routes with the following two properties:

  (1) ***Loop-freedom***: an (S,T)-route is loop-free when it has no repetitions of nodes

  (2) ***Freshness***: an (S,T)-route is fresh with respect to a (t_1,t_2) interval if each of the route's constituent links is up at some point during the (t_1,t_2)

- Loop-freedom and freshness are relevant for both explicit and implicit route discovery, and both basic and augmented protocols

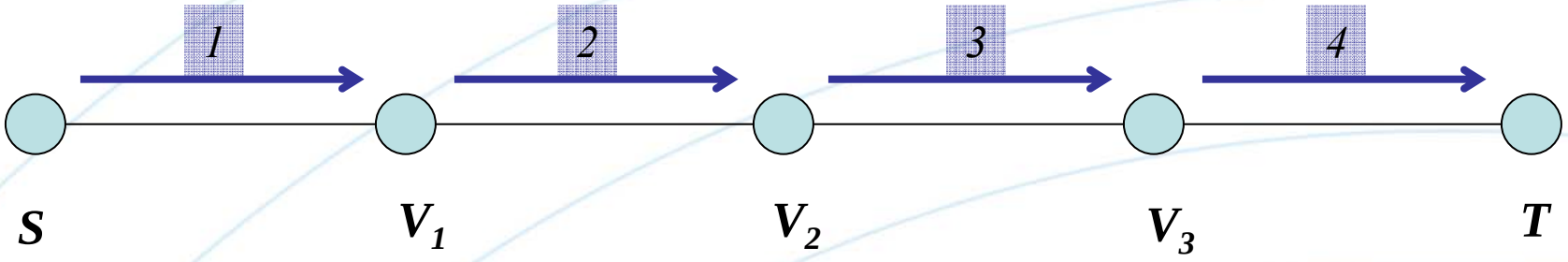
P. Papadimitratos, Z.J. Haas, and J.-P. Hubaux, "How to Specify and How to Prove Correctness of Secure Routing Protocols for MANET," BroadNets'06

Secure Routing Protocol (SRP)

- Explicit basic route discovery
- Observation
  - It is hard to 'know' all nodes in the network, i.e., establish associations with all of them
  - Often infeasible and very costly
  - Especially in 'open' networks
- SRP assumptions
  - Secure neighbor discovery
  - Hop-by-hop authentication of all control traffic
  - End nodes (source, destination) 'know' each other
    - Can set up security associations

P. Papadimitratos and Z.J. Haas, "Secure Routing for Mobile Ad Hoc Networks," CNDS 2002

SRP (cont'd)

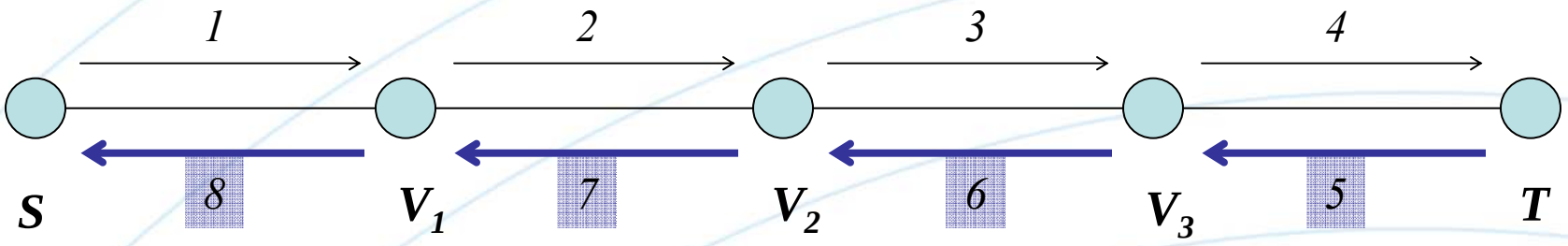


Route Request (*RREQ*):

$$S,\ T,\ Q_{SEQ},\ Q_{ID},\ MAC(K_{S,T},\ S,\ T,\ Q_{SEQ},\ Q_{ID})$$

1. S broadcasts *RREQ*;
2. V_1 broadcasts *RREQ*, $\{V_1\}$;
3. V_2 broadcasts *RREQ*, $\{V_1, V_2\}$;
4. V_3 broadcasts *RREQ*, $\{V_1, V_2, V_3\}$;

SRP (cont'd)



Route Reply (RREP):

$Q_{ID}, \{T, V_3, V_2, V_1, S\},$
$MAC(K_{S,T}, Q_{ID}, Q_{SEQ}, T, V_3, V_2, V_1, S)$

5. $T \rightarrow V_3$: *RREP*;
6. $V_3 \rightarrow V_2$: *RREP*;
7. $V_2 \rightarrow V_1$: *RREP*;
8. $V_1 \rightarrow S$: *RREP*;

SRP (cont'd)

- Route requests verifiably reach destination
  - Intermediate node replies disabled
  - Aggressive caching of routing information disabled
- Route replies must trace back the paths traversed by route requests
- Intermediate nodes are not authenticated at the end nodes
- Dual route request identifier
  - Q_{ID}: random, used by the intermediate nodes
  - Q_{SEQ}: sequence number, used by the destination
  - The adversary cannot launch a "sequence number" attack

SRP (cont'd)

- Crucial to operate on top a secure neighbor discovery protocol
- Neighbor Lookup Protocol (NLP)
  - Secure neighbor discovery
  - Establish security associations between neighbors
  - Identify control traffic injected by each neighbor
  - Prevent attacks that misuse network addresses
    - IP spoofing
    - Use of multiple identities
    - MAC spoofing
  - DoS protection
- Efficient mechanisms to discard spurious/ corrupted traffic at intermediate nodes
  - Replies relayed only if neighbors had previously forwarded the corresponding request

SRP (cont'd)

- Routes discovered by SRP in the presence of independent adversaries are fresh
  - t_1 is the point in time at which S transmitted a RREQ for T, and t_2 is the point at which S received the corresponding RREP
- In the presence of colluding adversaries SRP discovers 'weakly fresh' routes
  - A sequence of links, in general different than those in the discovered route were *up* at some point in (t_1,t_2)

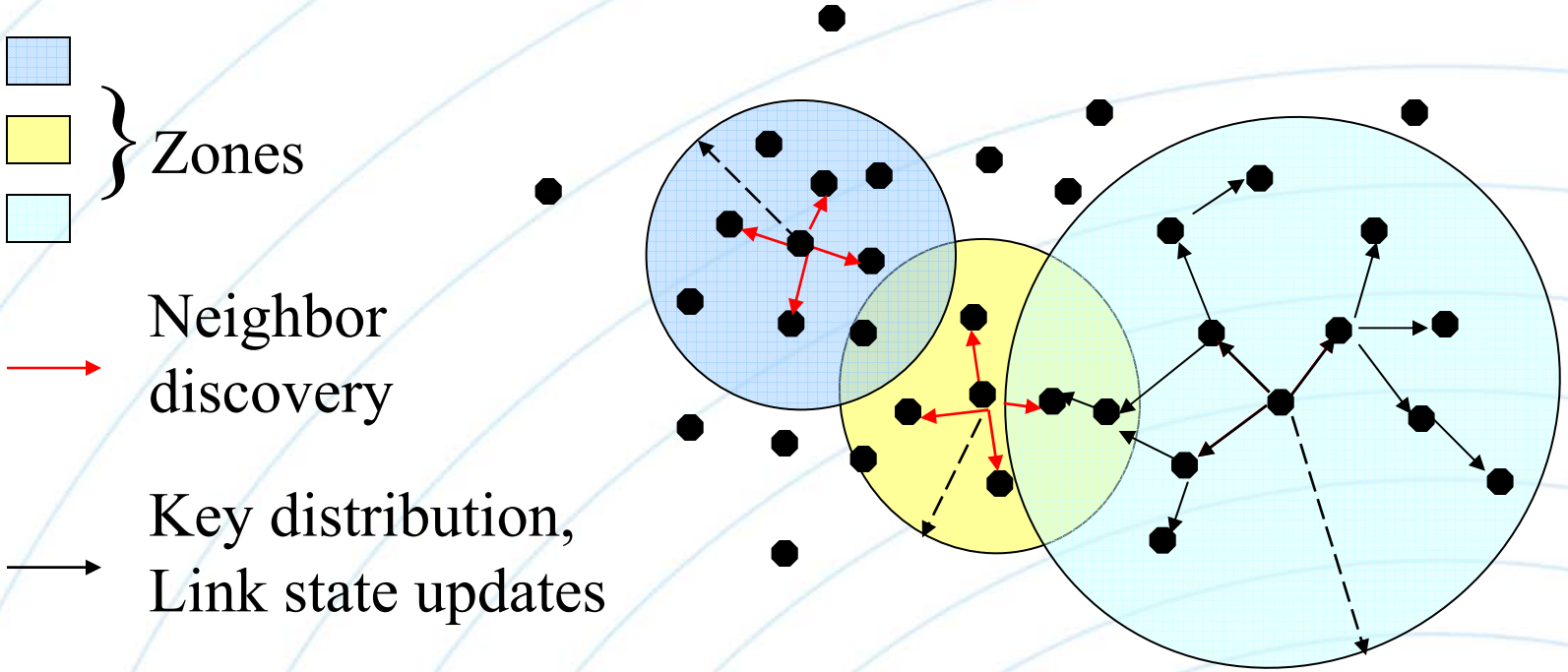
Secure Link State Protocol (SLSP)

- Secure Neighbor Discovery
  - Correct nodes discover only actual neighbors
- Periodic Link State Update (LSU) advertisements
  - Nodes distribute their discovered neighbors within an extended neighborhood, the *zone*
  - LSUs are signed
- Link state accepted *iff* reported by both incident nodes
- Nodes distribute their public key throughout the zone
- SLSP can adjust its scope with different zone radii

P. Papadimitratos and Z.J. Haas, "Secure Link State Routing for Mobile Ad Hoc Networks," WSAAN'03

SLSP (cont'd)

- SLSP can adjust its scope, with different zone radii
- It can operate locally, combined with another global route discovery, or network-wide



SLSP (cont'd)

- Keep the LSU propagation within the zone
  - Use a hash chain mechanism
  - *zone_radius* = $X_R = h^R(x_0)$
  - *hops_traversed* = $X_1 = h(x_0)$, *TTL = R-1*
  - After *i* hops (*i=R-TTL*), relay packet if:
    - $i < R$, and
    - $h^{R-i}(hops\_traversed) == zone\_radius$
  - *hops_traversed* = *H(hops_traversed)*
- Same idea can be applied in reactive routing, to perform an expanding ring search

Authenticating intermediate nodes

- Source knows all nodes in the network
- All nodes know any source and destination node (especially in the case of reactive protocols)
- Overall, all nodes know all nodes, or equivalently have security associations established before any route discovery
- Hard to achieve, yet what if? For example, in small or closed networks

Ariadne

- Secures DSR, adding authentication of RREQ and RREP messages by each intermediate node that relays and modifies them
- All-to-all security associations
- Use of different cryptographic primitives
  - Signatures, Message Authentication Codes, and TESLA

Y.-C. Hu, A. Perrig, and D. Johnson,"Ariadne: A secure on-demand routing protocol for ad hoc networks," Wireless Networks, 2005

Ariadne (cont'd)

- Operation across a route (S, F_1, F_2, D) with MACs
- If TESLA is used, the delayed authentication (for key disclosure) becomes part of the route discovery delay

S	:	$h_S = MAC_{SD}(\text{rreq}, S, D, id)$
$S \rightarrow *$	:	$(\text{rreq}, S, D, id, h_S, [\,], [\,])$
F_1	:	$h_{F_1} = H(F_1, h_S)$
$F_1 \rightarrow *$	:	$(\text{rreq}, S, D, id, h_{F_1}, [F_1], [mac_{F_1}])$
F_2	:	$h_{F_2} = H(F_2, h_{F_1})$
$F_2 \rightarrow *$	:	$(\text{rreq}, S, D, id, h_{F_2}, [F_1, F_2], [mac_{F_1}, mac_{F_2}])$
$D \rightarrow F_2$	:	$(\text{rrep}, D, S, [F_1, F_2], mac_D)$
$F_2 \rightarrow F_1$	:	$(\text{rrep}, D, S, [F_1, F_2], mac_D)$
$F_1 \rightarrow S$	:	$(\text{rrep}, D, S, [F_1, F_2], mac_D)$

Protocol operation as in Fig. 7.6 (p.202) of SeCoWiNet book

EndairA

- All-to-all security associations, digital signatures
- **Novelty**: intermediate nodes sign only the RREP
- Withstands provably attacks and reduces overhead with respect to Ariadne

$S \rightarrow *$	:	$(\text{rreq}, S, D, id, [\,])$
$F_1 \rightarrow *$	:	$(\text{rreq}, S, D, id, [F_1])$
$F_2 \rightarrow *$	:	$(\text{rreq}, S, D, id, [F_1, F_2])$
$D \rightarrow F_2$	:	$(\text{rrep}, S, D, id, [F_1, F_2], [sig_D])$
$F_2 \rightarrow F_1$	:	$(\text{rrep}, S, D, id, [F_1, F_2], [sig_D, sig_{F_2}])$
$F_1 \rightarrow S$	:	$(\text{rrep}, S, D, id, [F_1, F_2], [sig_D, sig_{F_2}, sig_{F_1}])$

Protocol operation as in Fig. 7.8 (p.206) of SeCoWiNet book

G. Acs, L. Buttyan, and I. Vajda, "Provably secure on-demand source routing in mobile ad hoc networks," TMC, 2006

Augmented Discovery: Requirement

- Let $l_{i,i+1} \in M$ be the actual link metric for each link of a discovered (S,T)-route and $g(l_{0,1},\ldots,l_{n-1,n})$ the actual route metric
- The metric estimated (by a protocol) for link (V_i,V_{i+1}) is $m_{i,i+1}$

(3) Accuracy: an (S,T)-route is accurate with respect to a route metric *g* and a constant $\Delta_{good} \geq 0$ if:

$$| g(m_{0,1},\ldots,m_{n-1,n}) - g(l_{0,1},\ldots,l_{n-1,n}) | < \Delta_{good}$$

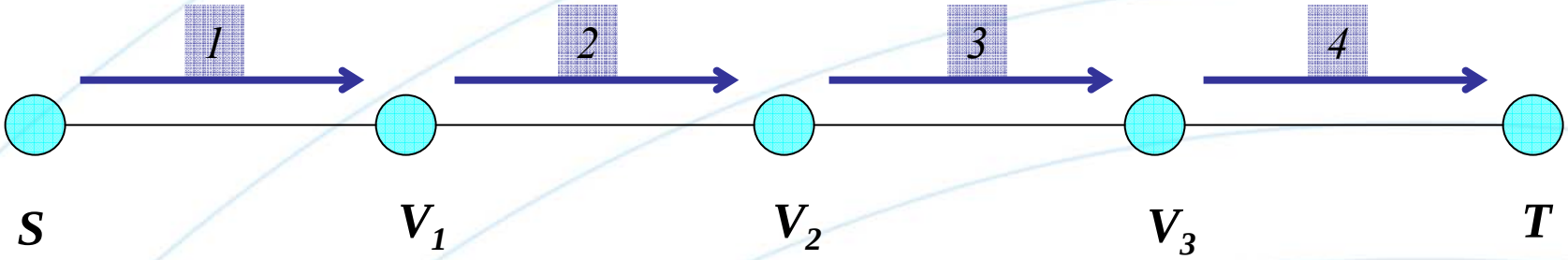
- Accuracy is relevant only to augmented, explicit or implicit, route discovery

Quality-of-Service Aware Discovery

- QoS-SRP: Secure QoS-aware routing
- Nodes estimate metrics for their incident links
  - For link (V_i, V_{i+1}), node V_i calculates $m^{i}_{i,i+1}$ and V_{i+1} calculates $m^{i+1}_{i,i+1}$
  - For some $\varepsilon > 0$, $\left| m^{i}_{i,i+1} - m^{i+1}_{i,i+1} \right| < \varepsilon$
  - ε is a protocol-selectable and metric-specific threshold that allows for metric calculation inaccuracies
  - $\tilde{\delta} \geq 0$ is the maximum metric calculation error by a correct node

P. Papadimitratos and Z.J. Haas, "Secure Route Discovery for QoS-Aware Routing in Ad Hoc Networks," Sarnoff '05

QoS-SRP

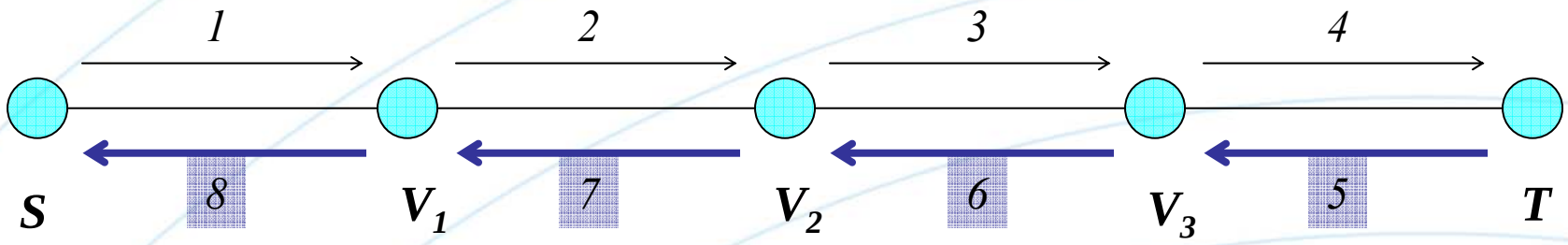


Route Request (*RREQ*):

$$S,\ T,\ Q_{SEQ},\ Q_{ID},\ MAC(K_{S,T},\ S,\ T,\ Q_{SEQ},\ Q_{ID})$$

1. *S* broadcasts *RREQ*;
2. V_1 broadcasts *RREQ*, $\{V_1\}$, $\{ m^1_{S,1} \}$;
3. V_2 broadcasts *RREQ*, $\{V_1, V_2\}$, $\{ m^1_{S,1}, m^2_{1,2} \}$;
4. V_3 broadcasts *RREQ*, $\{V_1, V_2, V_3\}$, $\{ m^1_{S,1}, m^2_{1,2}, m^3_{2,3} \}$;

QoS-SRP (cont'd)



Route Reply (RREP):

Q_{ID}, {T, V_3, V_2, V_1, S}, { $m^{T}_{3,T}, m^{3}_{2,3}, m^{2}_{1,2}, m^{1}_{S,1}$ },

$MAC(K_{S,T}, Q_{SEQ}, Q_{ID}, T, V_3, \ldots, V_1, S, m^{T}_{3,T}, \ldots, m^{1}_{0,1})$

5. $T \rightarrow V_3$: RREP;
6. $V_3 \rightarrow V_2$: RREP;
7. $V_2 \rightarrow V_1$: RREP;
8. $V_1 \rightarrow S$: RREP;

QoS-SRP (cont'd)

- Metric types
  - Δ_{good}^{add} , $g_{add}\left(m_{0,1}^{1},\ldots,m_{n-1,n}^{n}\right)=\sum_{i=0}^{n-1} m_{i,i+1}^{i+1}$
  - If $m_{i,i+1}^{i+1}>0$, $g\left(m_{0,1}^{1},\ldots,m_{n-1,n}^{n}\right)=\prod_{i=0}^{n-1} m_{i,i+1}^{i+1}$

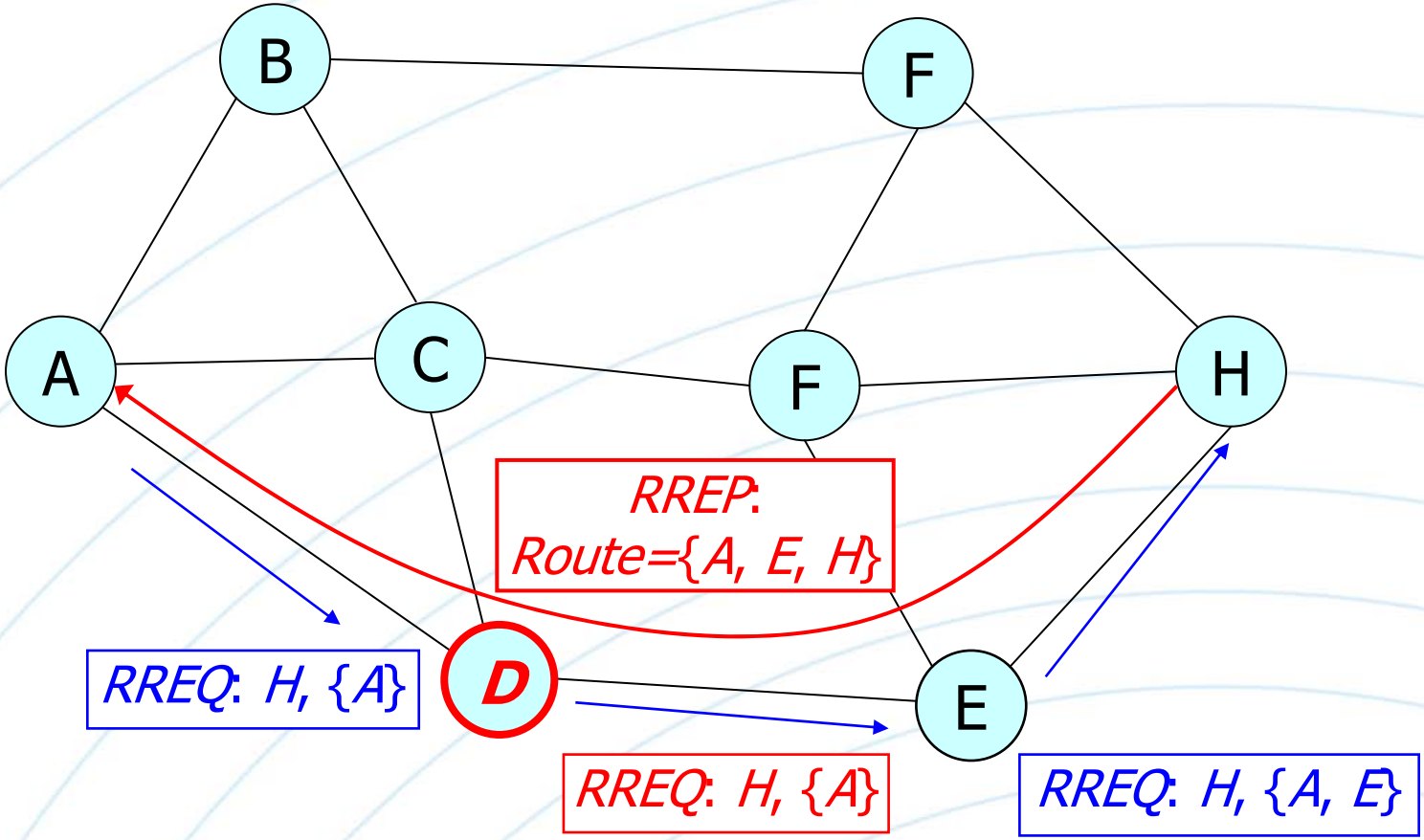
    can be written as $g_{add}\left(\overline{m}_{0,1}^{1},\ldots,\overline{m}_{n-1,n}^{n}\right)$

    where $\overline{m}_{i,i+1}^{i+1}=\log(m_{i,i+1}^{i+1}),\ \text{for } 0\le i\le n-1$
  - $\Delta_{good}^{\max}$, $g_{\max}\left(m_{0,1}^{1},\ldots,m_{n-1,n}^{n}\right)=\max_{0\le i\le n-1}\left\{m_{i,i+1}^{i+1}\right\}$
  - $\Delta_{good}^{\min}$, $g_{\min}\left(m_{0,1}^{1},\ldots,m_{n-1,n}^{n}\right)=\min_{0\le i\le n-1}\left\{m_{i,i+1}^{i+1}\right\}$

QoS-SRP (cont'd)

- Routes discovered by SRP in the presence of independent adversaries are accurate, with respect to (*i*) g_{add} and $\Delta_{good}^{add} = n^2\varepsilon + n\tilde{\delta}$ (*ii*) g_{max} and $\Delta_{good}^{\max} = n\varepsilon + \tilde{\delta}$, and (*iii*) g_{min} and $\Delta_{good}^{\min} = n\varepsilon + \tilde{\delta}$, with n the number of route links, $\varepsilon > 0$ the maximum allowable difference between $m_{i,i+1}^{i}$ and $m_{i,i+1}^{i+1}$, and $\tilde{\delta} \geq 0$ the maximum error for a metric calculation by a correct node.

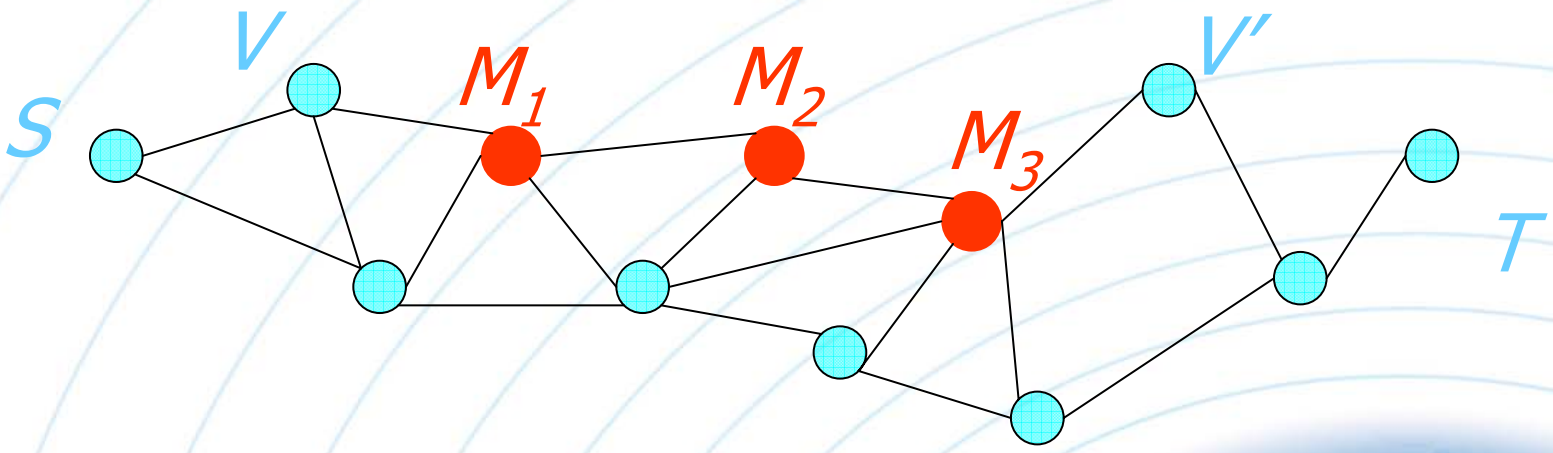
Attacking route discovery (cont'd)



- Adversary acting as a relay, 'creating' Byzantine links
- Secure neighbor discovery and hop-by-hop authentication can defeat this attack

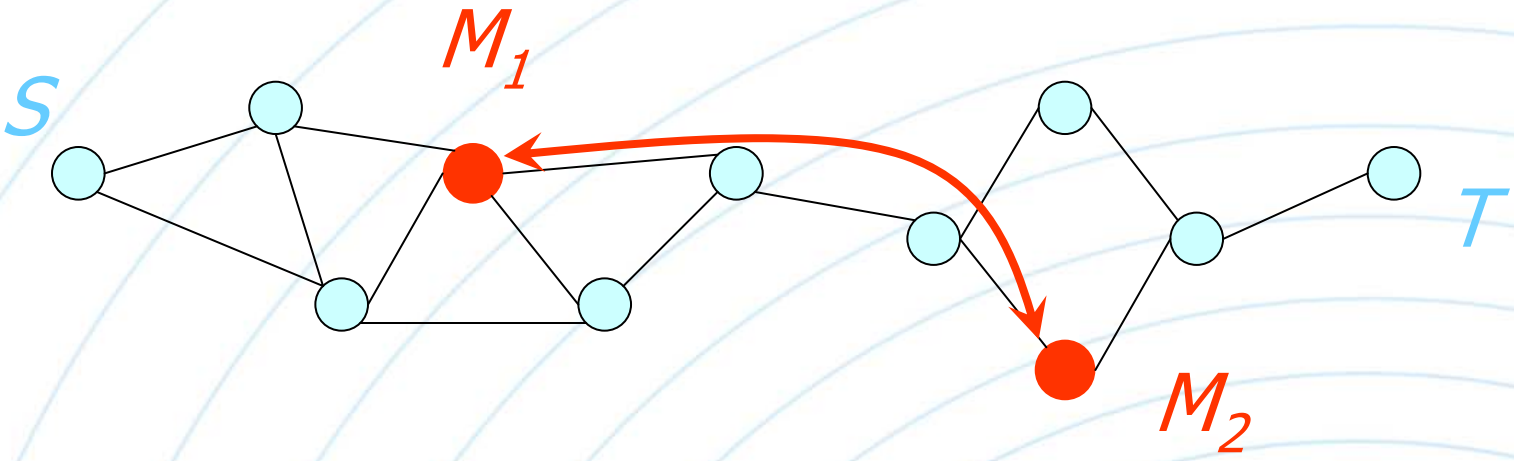
Attacking Routing – Revisited (cont'd)

- Multiple Colluding Attackers
  - M_1 and M_3 are seemingly correct to their neighbors, but they 'omit' protocol functionality when handling packets from M_2
  - Example: M_2 relays RREQ and RREP packets without appearing in the route discovery



Attacking Routing - Revisited

- Tunneling Attack
  - Two colluding attackers: M_1, M_2
  - M_1 encapsulates control traffic and forwards to M_2 and vice versa
  - Attackers seemingly follow the protocol with respect to their neighbors



P. Papadimitratos and Z.J. Haas, "Secure Routing for Mobile Ad Hoc Networks," CNDS 2002

Summary

- Route discovery is vulnerable
- Secure route discovery specification
  - Loop freedom
  - Freshness
  - Accuracy
- Protocols relying on different trust assumptions
- Securing basic and augmented route discovery in open, dynamic networks
- Colluding adversarial nodes can subvert any route discovery protocol; 'tunneling attack'
- Additional reading
  - More secure routing protocols, including sensor network protocols

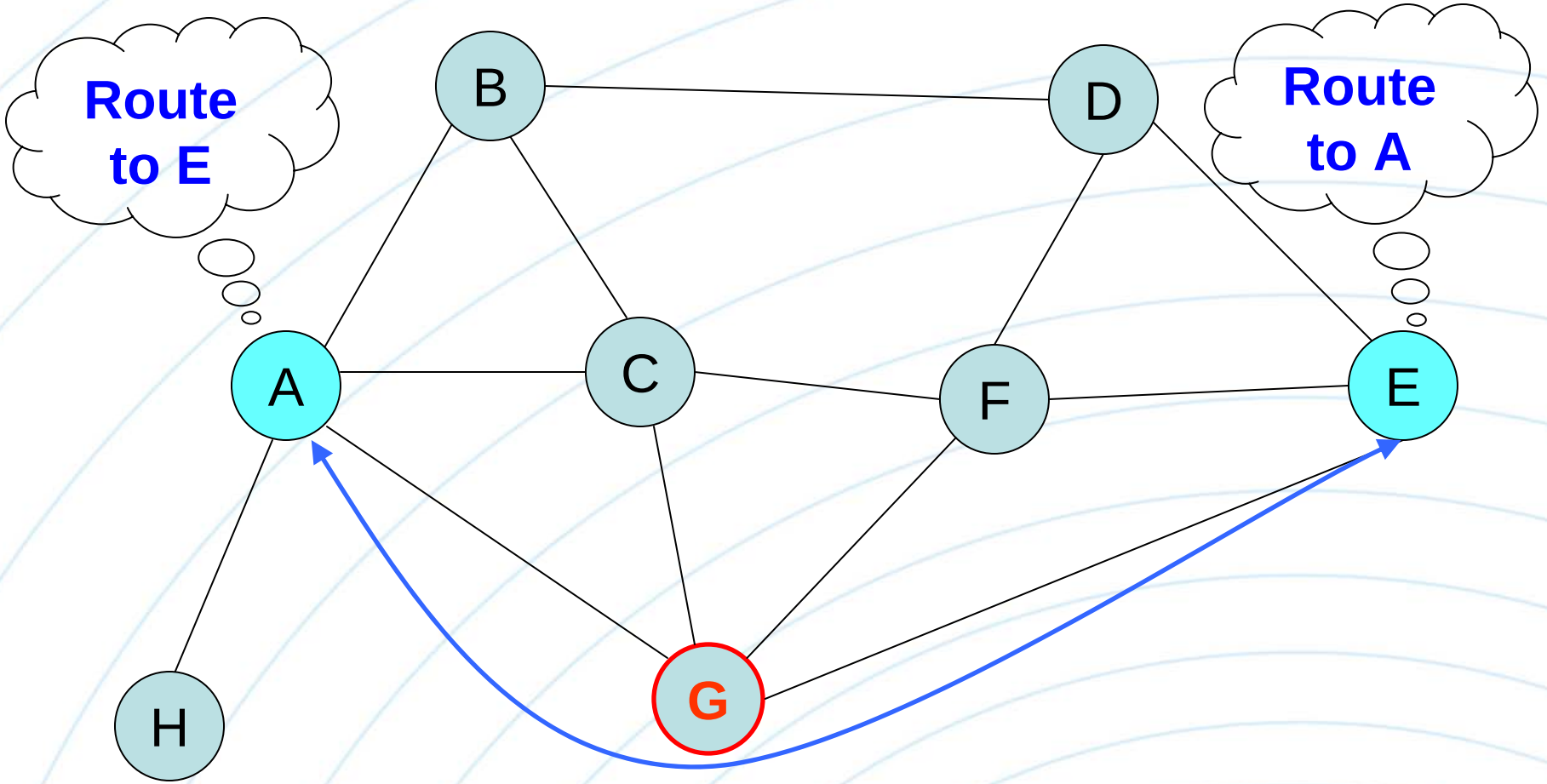
Secure Data Communication

Problem statement

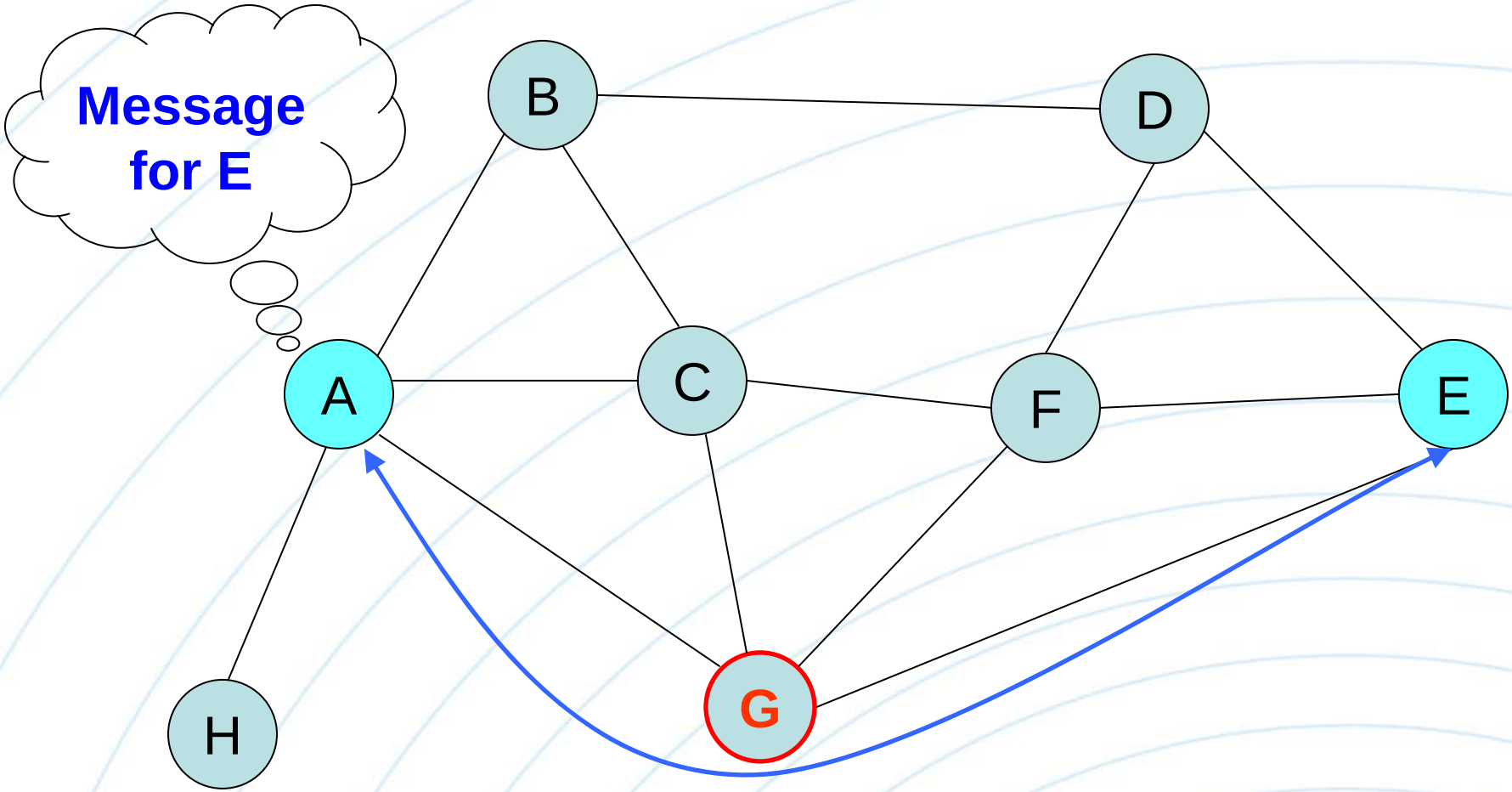
- **Goal:**
  - Reliable and low-delay data delivery in the presence of attackers that disrupt the data communication

- **Solution:**
  - Detect and avoid compromised and failing routes
  - Tolerate malicious and benign faults
    - In general, hard to distinguish in highly dynamic networking environments

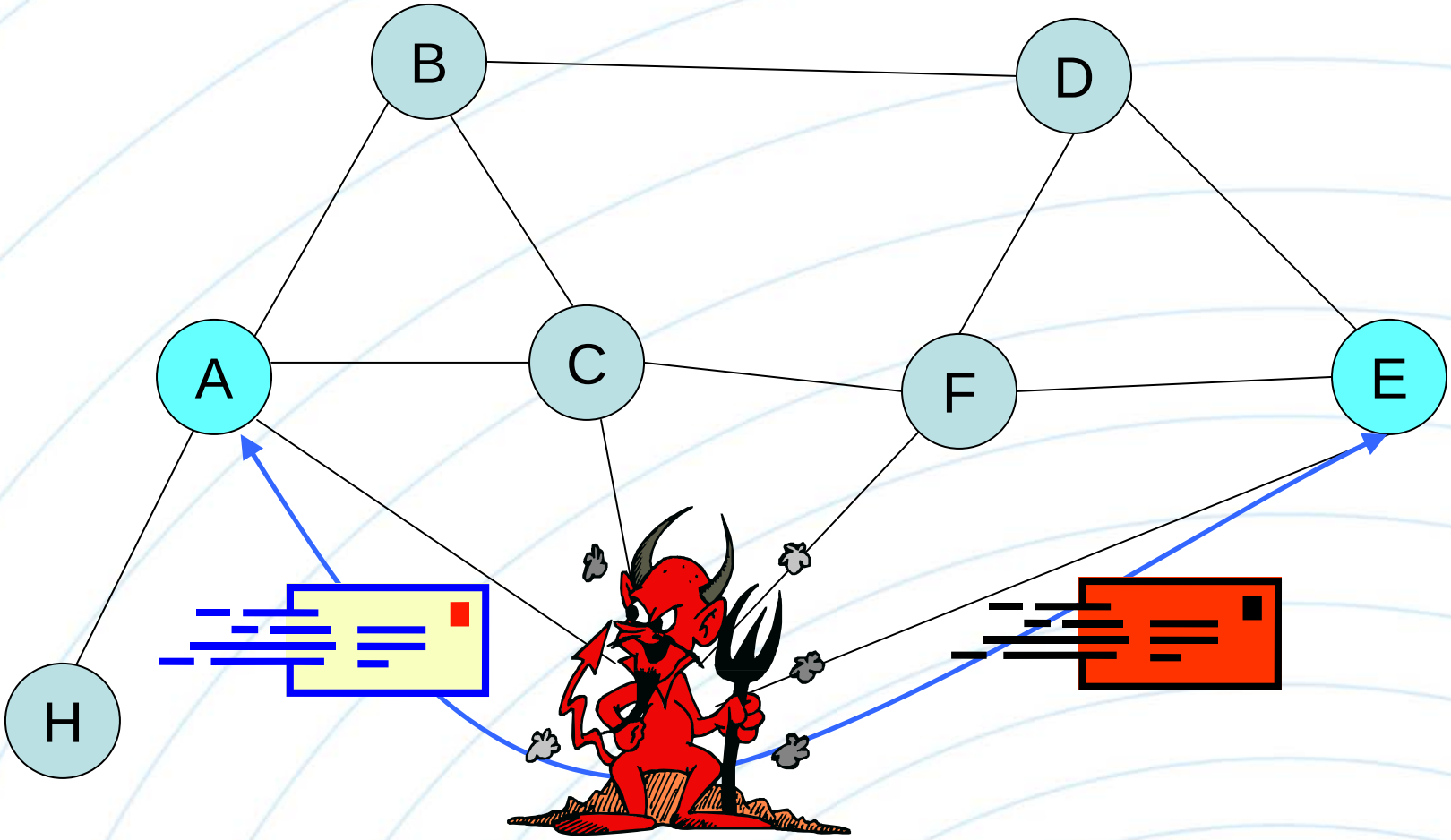
Data Communication



Data Communication (cont'd)



Data Communication (cont'd)

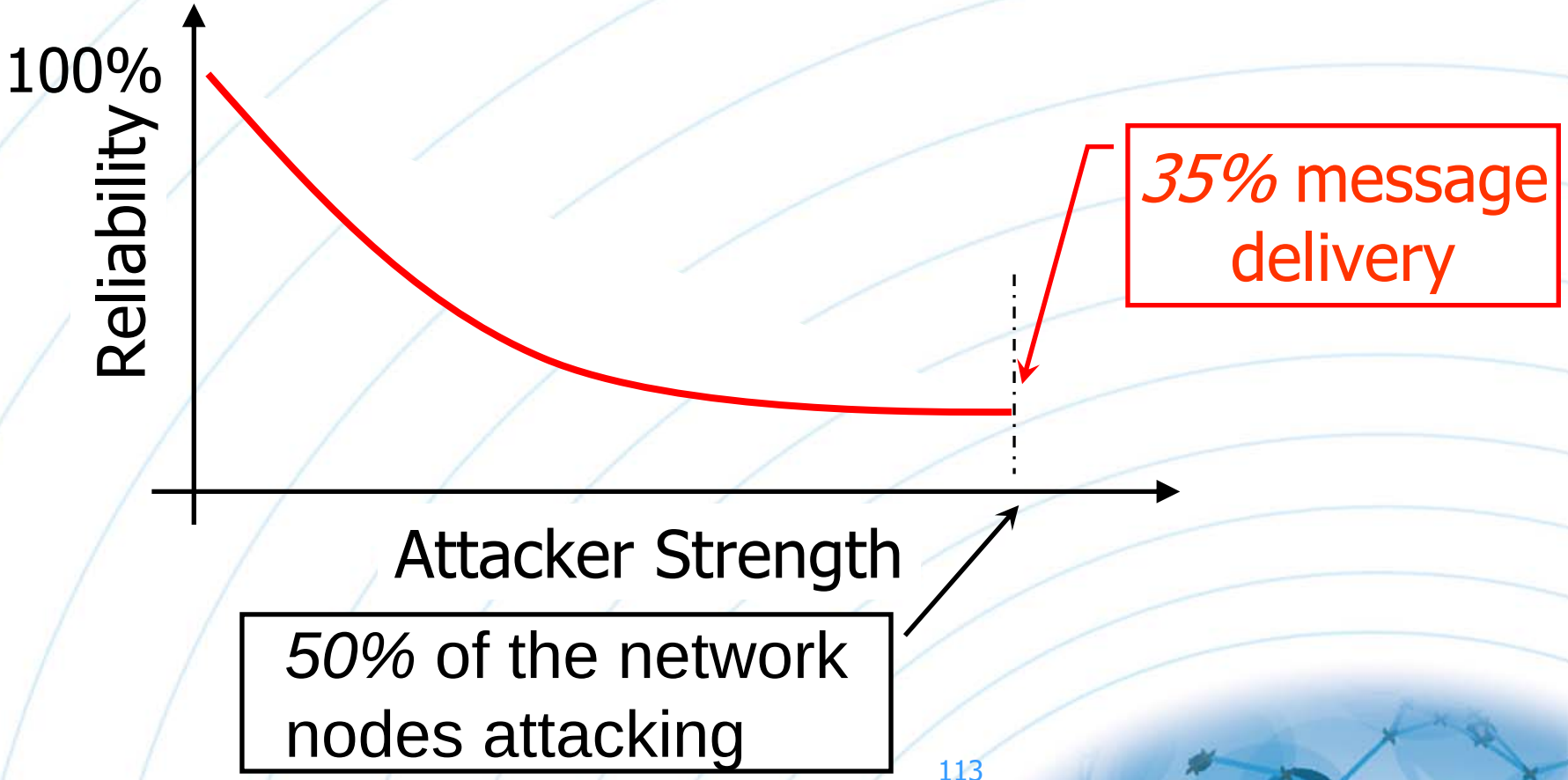


Data Communication (cont'd)

- How can an attacker be part of a route?
  - Make the route appear 'preferable' (shorter in hops, delay, or any other metric)
  - Other routing protocol-specific attacks (e.g., 'rushing')
  - Do nothing that disrupts the secure route discovery
- Consider
  - An ideal secure routing protocol, ensuring loop-free, fresh, and accurate routes against any possible attack
  - All nodes on the discovered route authenticated
- Still, the attacker can deny communication, dropping packets
- Worse even, the attacker can choose to hit when it hurts the most

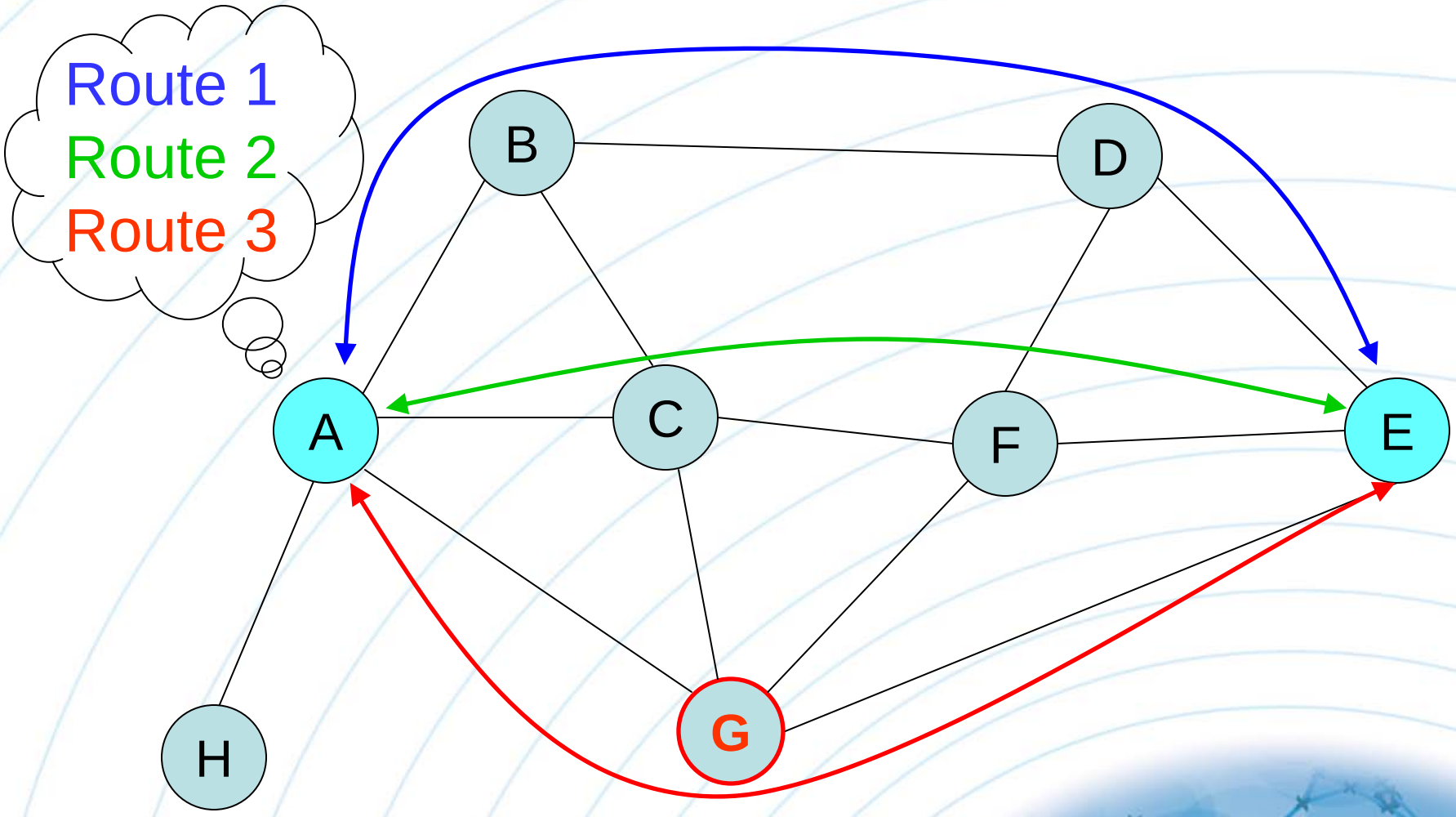
Data Communication (cont'd)

- What is the impact of the adversary that 'lies low' and disrupts only the data communication?



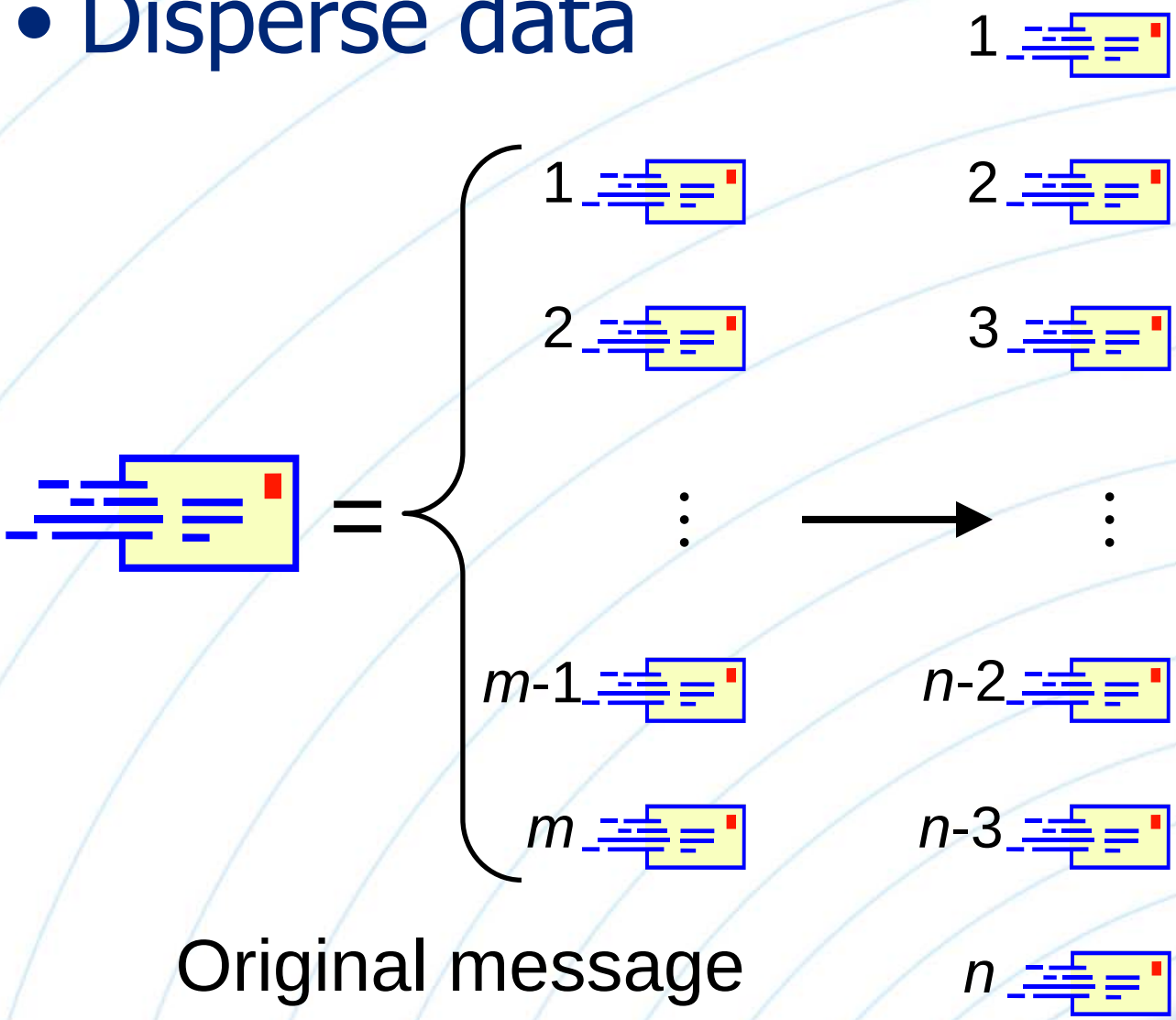
Securing Data Communication

- Use multiple routes



Securing Data Communication (cont'd)

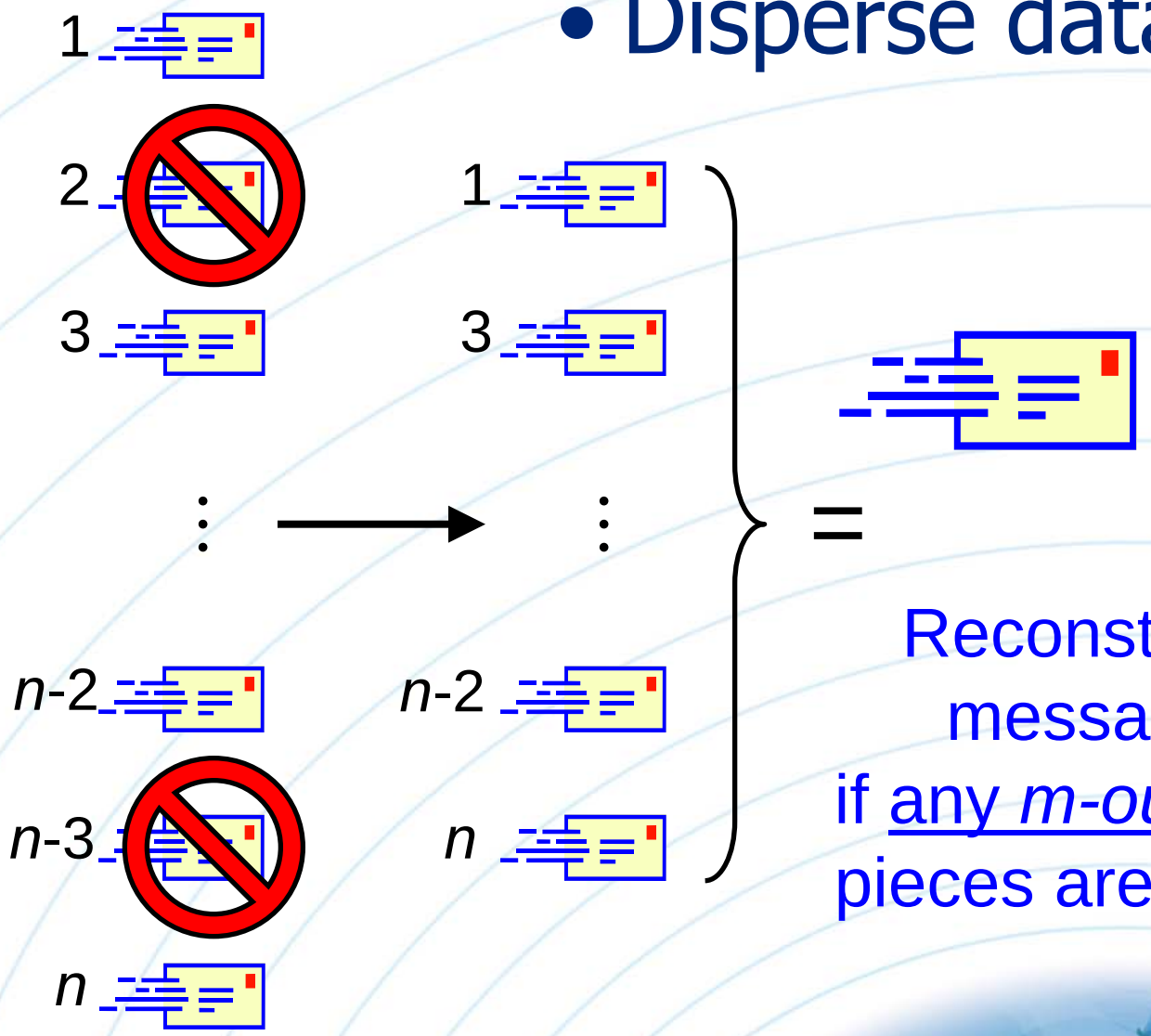
- Disperse data



Introduce *redundancy* to the original message

Securing Data Communication (cont'd)

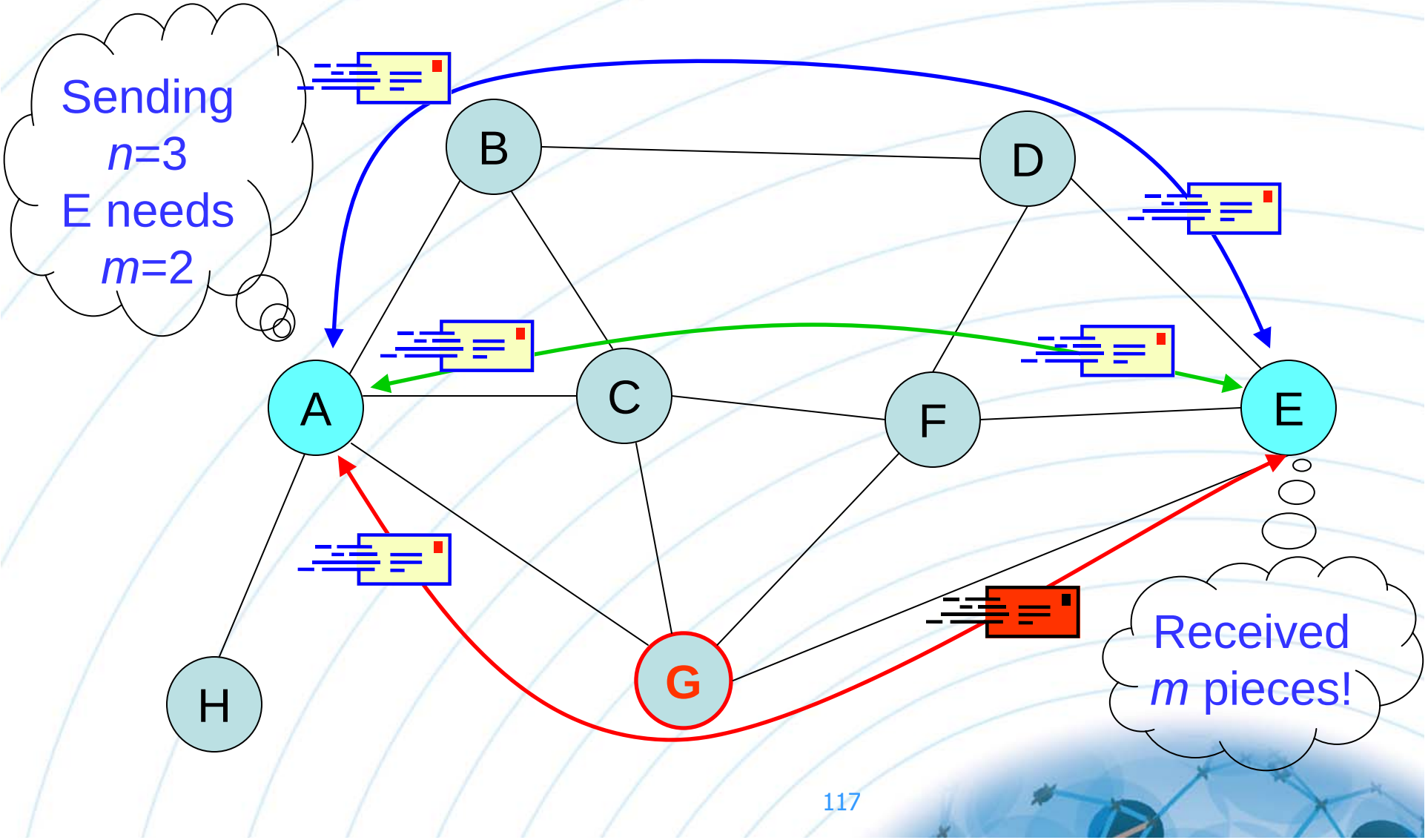
- Disperse data



Reconstruct message if any *m-out-of-n* pieces are intact

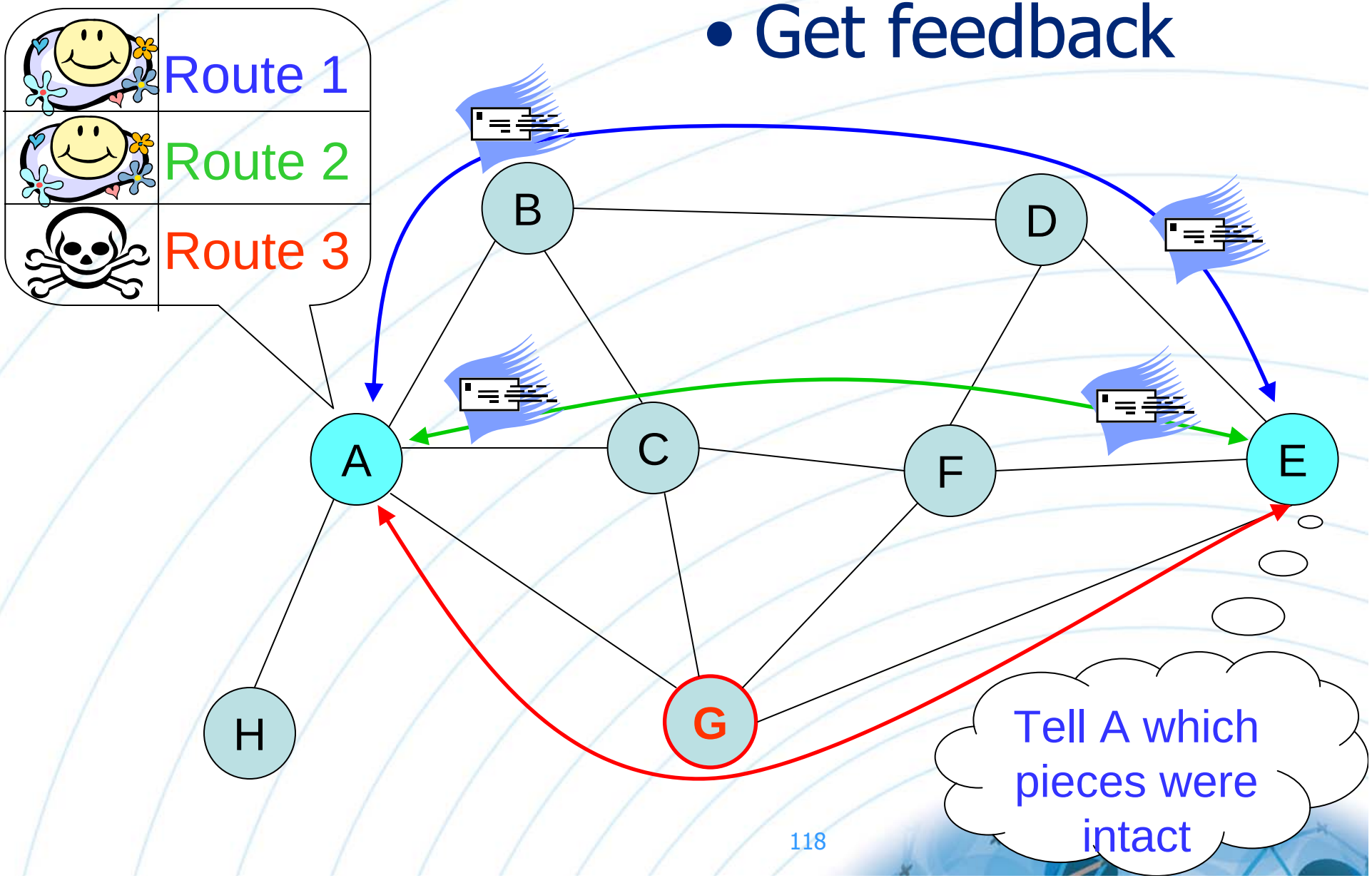
Securing Data Communication (cont'd)

- Transmit simultaneously across the routes



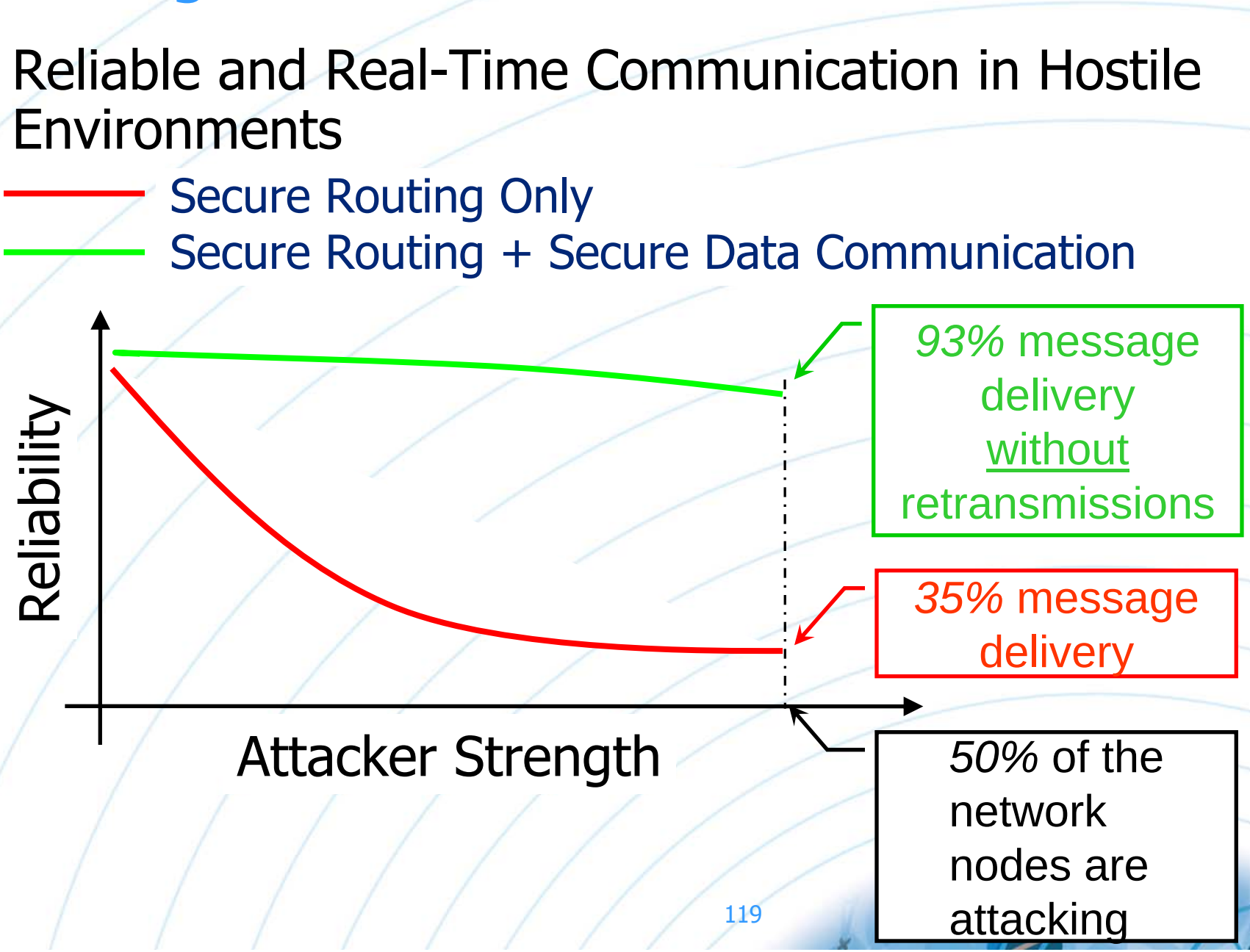
Securing Data Communication (cont'd)

- Get feedback



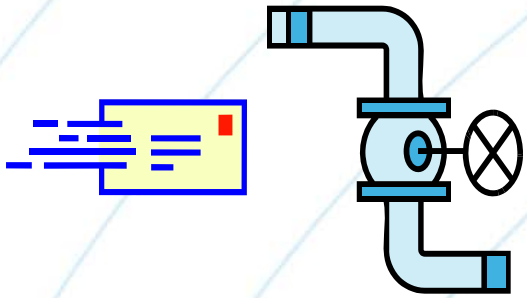
Securing Data Communication (cont'd)

- Reliable and Real-Time Communication in Hostile Environments

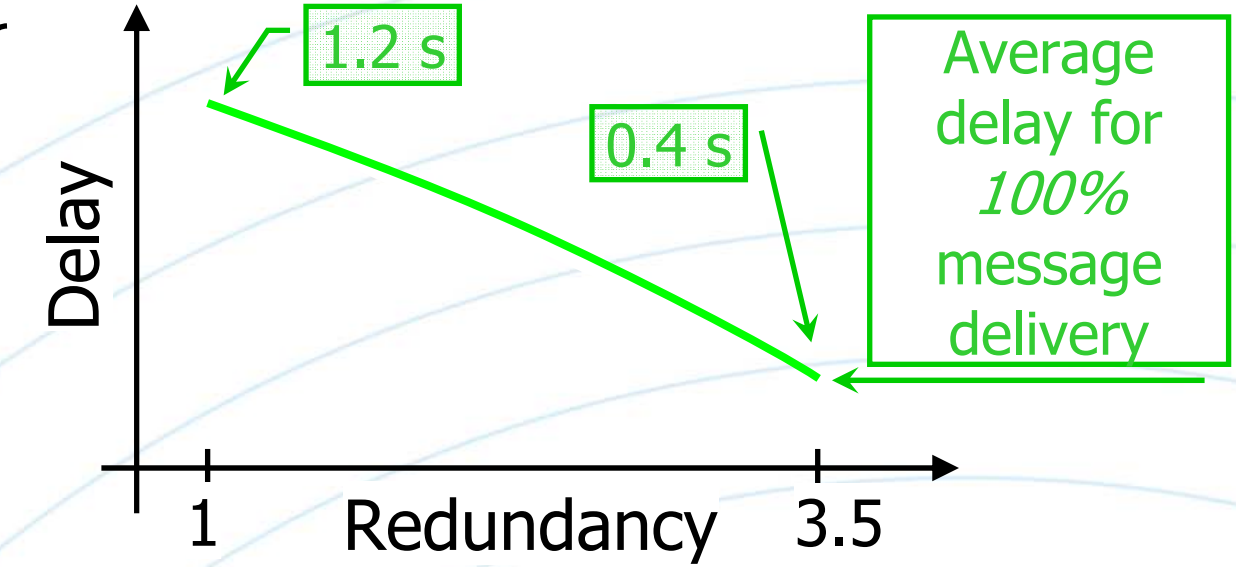


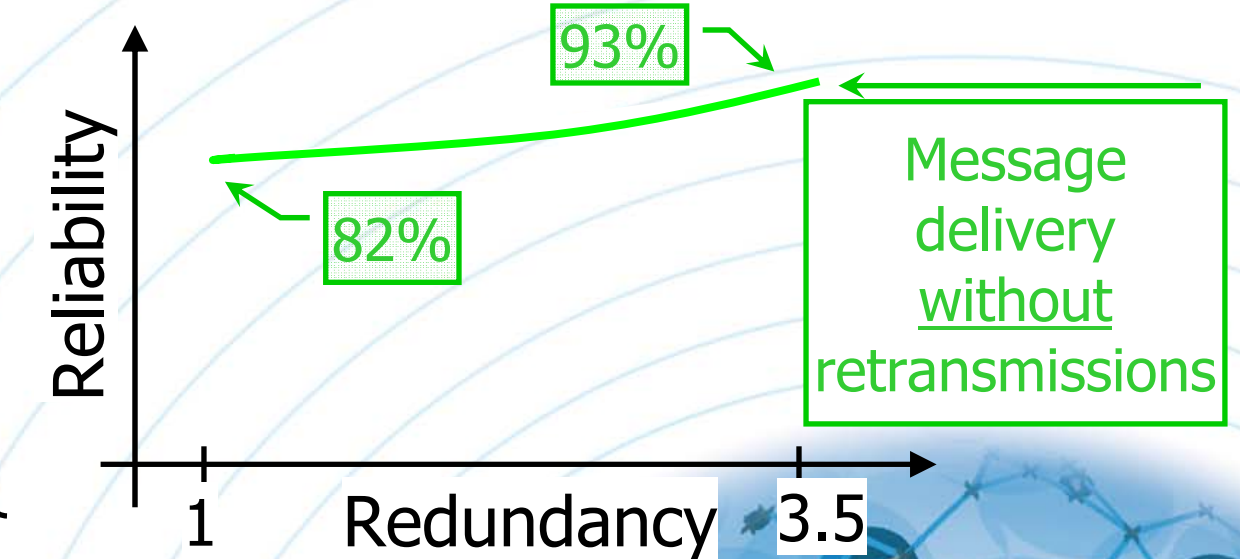
Securing Data Communication (cont'd)

Bandwidth For Security



Redundancy





Securing Data Communication (cont'd)

- Secure Message Transmission (SMT) protocol
  - Dispersion of the transmitted data
  - Simultaneous usage of multiple node-disjoint routes
  - Data integrity and origin authentication
  - End-to-end secure and robust feedback
  - Adaptation to the network conditions

- Secure Single Path (SSP) protocol
  - Discovery and utilization of a single route
  - End–to–end security and feedback

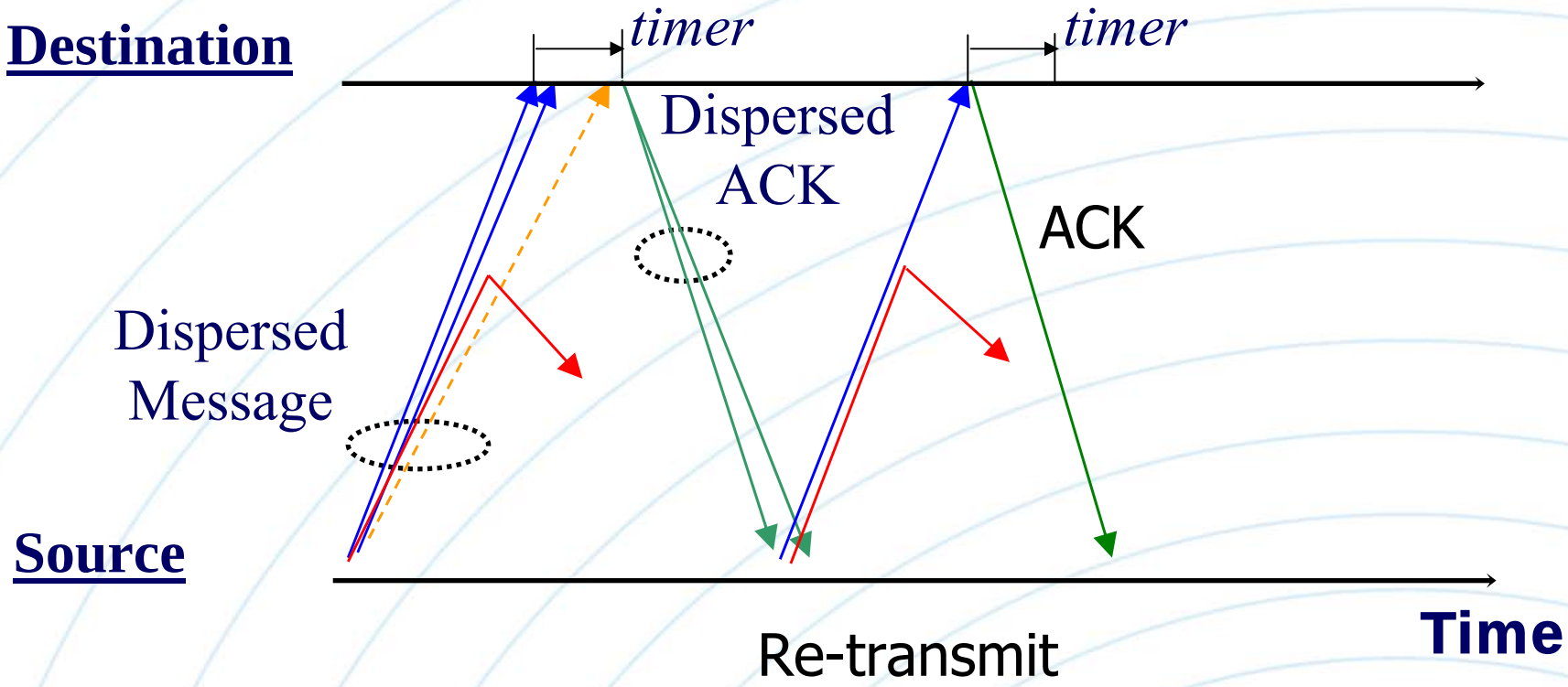
SMT Operation

- The Active Path Set (*APS*)
  - Maintain a (partial) view of the network topology
  - Construct a set of *node disjoint* routes (per destination)
  - Routes remain in the *APS* until deemed non-operational

- Multi-path operation
  - Select the *APS* routes to transmit a *dispersed* message
  - Route selection attributes
    - Path rating
    - Probability of path survival
    - Overall probability of successful message delivery
  - Assign each message piece to one of the selected routes

SMT Operation (cont'd)

- Example: Transmission of a single message



SMT Operation (cont'd)

- Secure and robust end-to-end feedback
  - Dispersed and returned over multiple routes
  - Informs on the successfully received pieces
  - Allows the correlation of successfully received pieces with data routes
  - Provides "safe" information for the adaptation of the protocol operation

SMT Operation (cont'd)

- Adapt to the network conditions
  - Detect non-operational routes
  - Switch to alternate (new) routes
  - Adapt the protocol configuration
    - Number of routes
    - Transmission redundancy
    - Route selection
    - Additional route discovery

SMT Operation (cont'd)

- Path rating mechanism
  - Each route is associated to a rating $r_s \in \left[r_S^{thr}, r_S^{\max} \right]$
    - Update r_s for each transmission across the route
    - For each *delivered* piece, r_s is *increased* by a constant β
    - For each *lost* piece, r_s is *decreased* by a constant α
    - The route is discarded when its rating reaches r_s^{thr}

$$r_s(i) = \begin{cases} \max\{r_s(i-1) - \alpha, r_s^{thr}\}, & \text{if a piece is lost} \\ \min\{r_s(i-1) + \beta, r_s^{\max}\}, & \text{if a piece is received} \end{cases}$$

SMT Operation (cont'd)

- Robustness to arbitrary attack patterns
  - Bounded fraction of data the adversary can drop (Bandwidth Loss (BWL)) before the compromised route is detected

$$BWL \leq \frac{\beta}{\alpha + \beta}$$

  - Non-operational routes are promptly discarded
  - Route re-instatement after transient data loss

P. Papadimitratos and Z.J. Haas, "Secure Message Transmission in Mobile Ad Hoc Networks," ACM WiSe'03
P. Papadimitratos and Z.J. Haas. "Secure Message Transmission in Mobile Ad Hoc Networks," Ad Hoc Networks, 2003

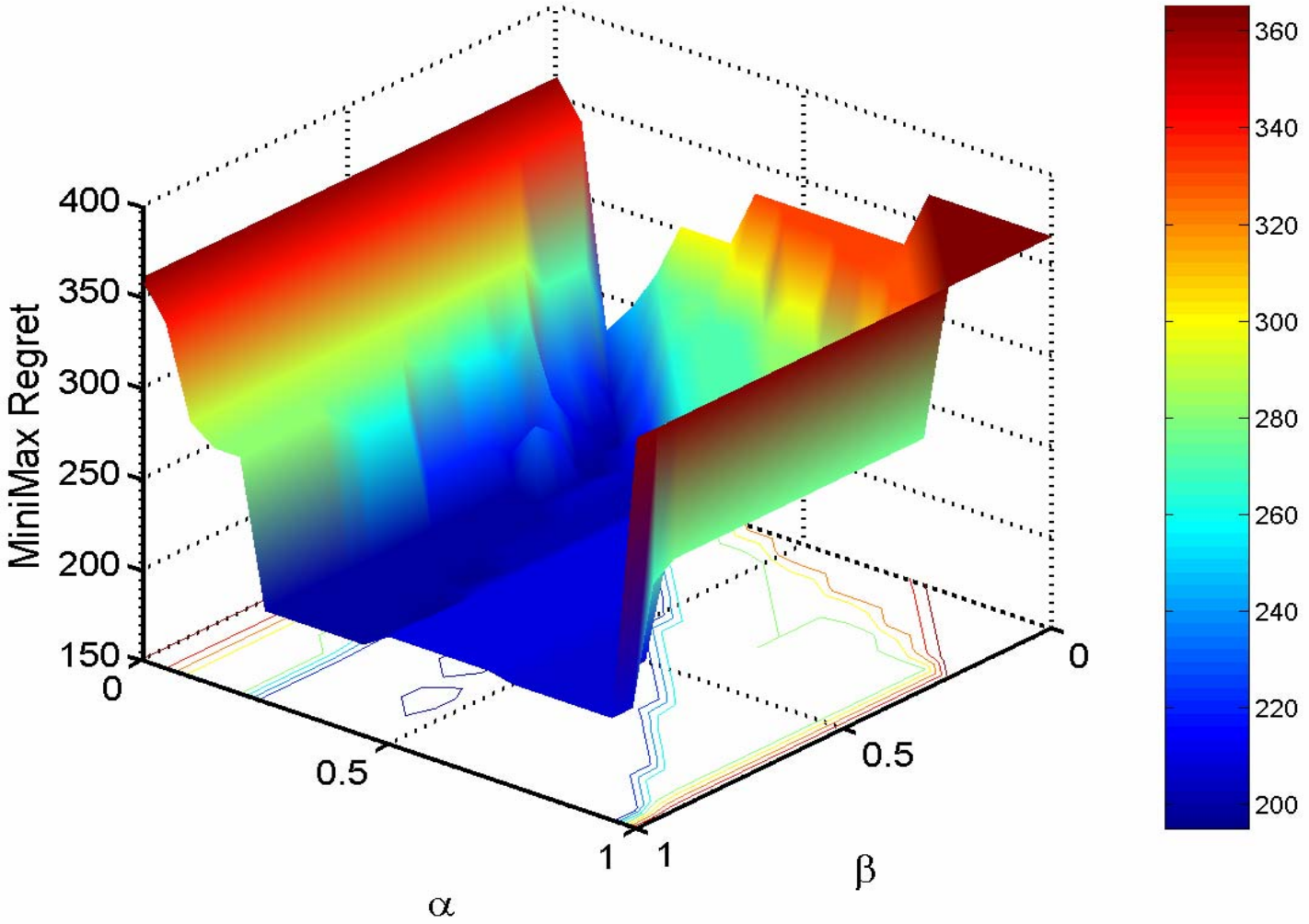
SMT Operation (cont'd)

- What is the appropriate choice for α,β ?
  - The attack pattern is not known in advance
  - The faster a non-operational route is discarded the better
  - Not discarding a route after a transient packet loss is preferable
- One criterion
  - Min-Max Regret

P. Papadimitratos and Z.J. Haas, " Secure Data Communication in Mobile Ad Hoc Networks," JSAC, 2006

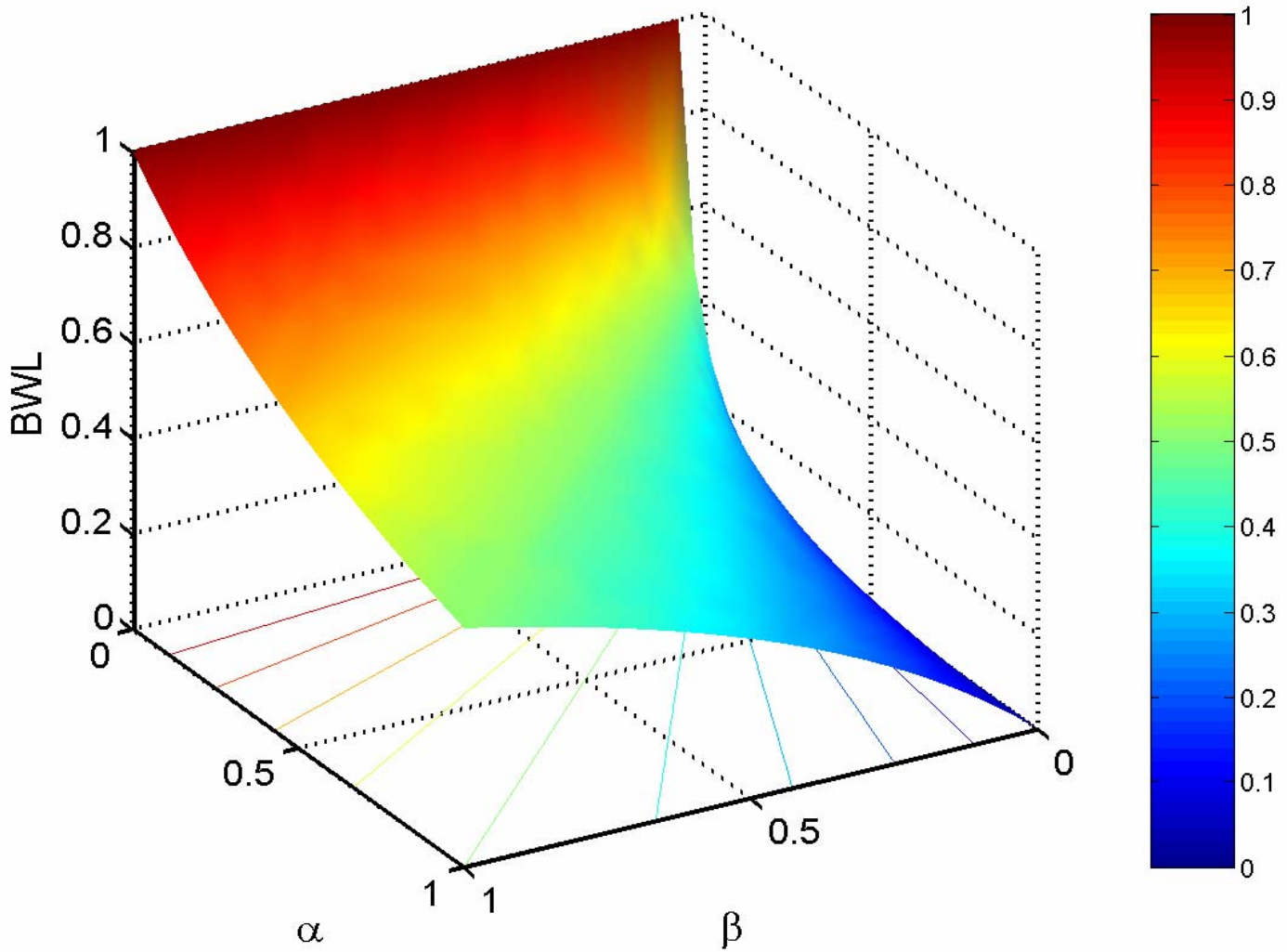
SMT Operation (cont'd)

- Selection of α, β



SMT Operation (cont'd)

- Selection of *α*, *β*

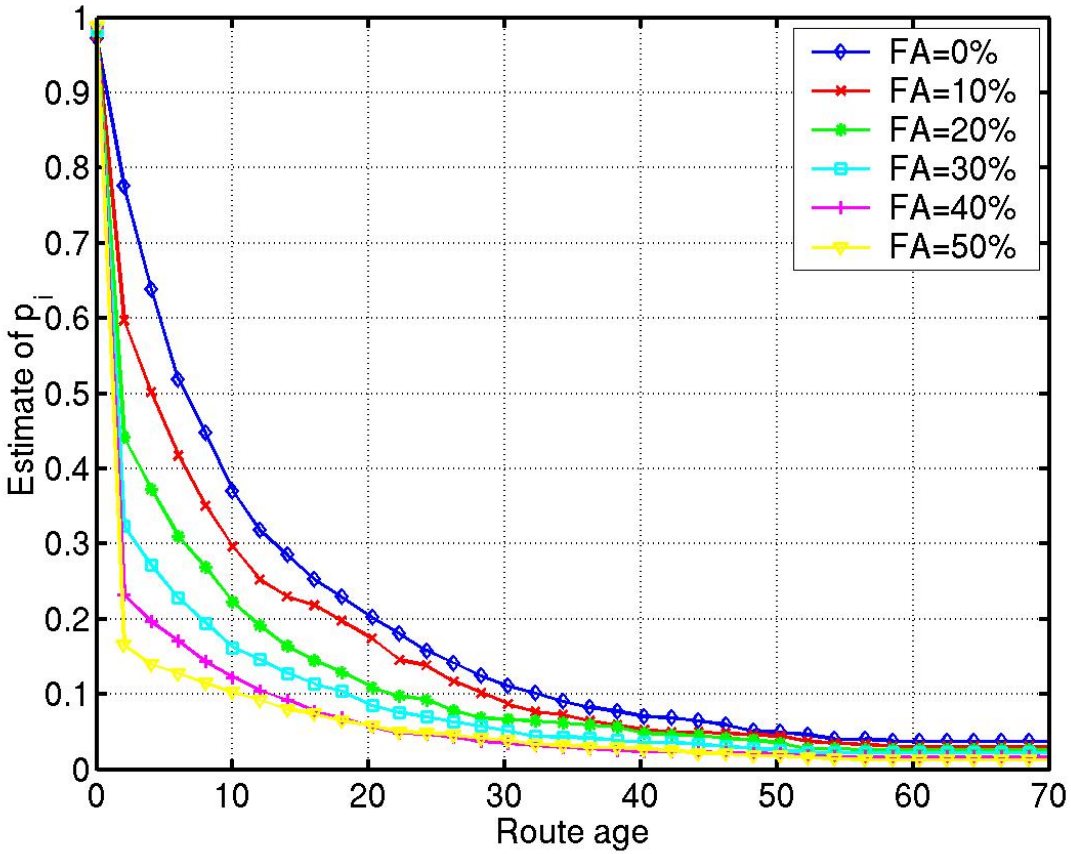


SMT Operation (cont'd)

- Is the route rating sufficient to maintain reliable communication? What about mobility?
- The higher the route age is, the more likely it is to break
- t : current route age of the i-th route in APS
- $p_i(t)$: probability of survival of the route during a piece transmission (delay d)
- Estimate this from *route lifetime* samples (periods of time from the discovery till the route removal from the APS

$$\hat{p}_i(t) = \begin{cases} \dfrac{S-1}{S}, & \text{if } t+d < \tau_1 \\ \dfrac{S-j}{S}, & \text{for } j \text{ such that: } \tau_j \le t+d < \tau_{j+1} \\ \dfrac{1}{S}, & \text{if } t+d > \tau_D \end{cases}$$

SMT Operation (cont'd)



- Example of the estimated probability of path survival , based on collected data
- FA: Fraction of adversaries present in the network

SMT Operation (cont'd)

- Determine the appropriate message dispersion
  - To achieve the sought end-to-end reliability, P_{GOAL}, while minimizing
    - The transmission redundancy: P_{GOAL} - r_{min}
    - The number of utilized paths: P_{GOAL}-N_{min}
  - To achieve a redundancy goal while maximizing the end-to-end reliability: r_{GOAL}

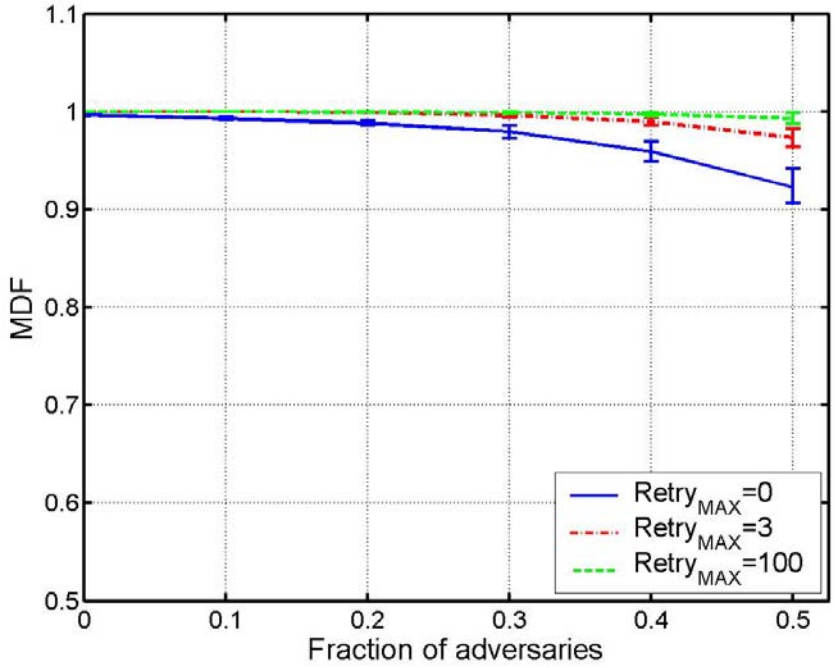
Performance Evaluation

Nodes	50
Fraction of Adversaries	10%, 20%, 30%, 40%, or 50% of the network
Measurements	50 randomly seeded runs for each point
Security Bindings	Single destination per source
Simulated time	300 sec
Mobility	Random waypoint; Pause times: 0, 20, 40, 60, 100, 150, 200, 250 seconds
Load	3, 7, 15, 20 CBR flows, Data payload: 512 Bytes Rates: 4, 10, 15, 20, 25, and 30 packets/sec
Coverage Area	1000m-by-1000m
PHY/MAC	802.11, DCF, 2 and 5.5 Mbps, 300m
Tool	OPNET

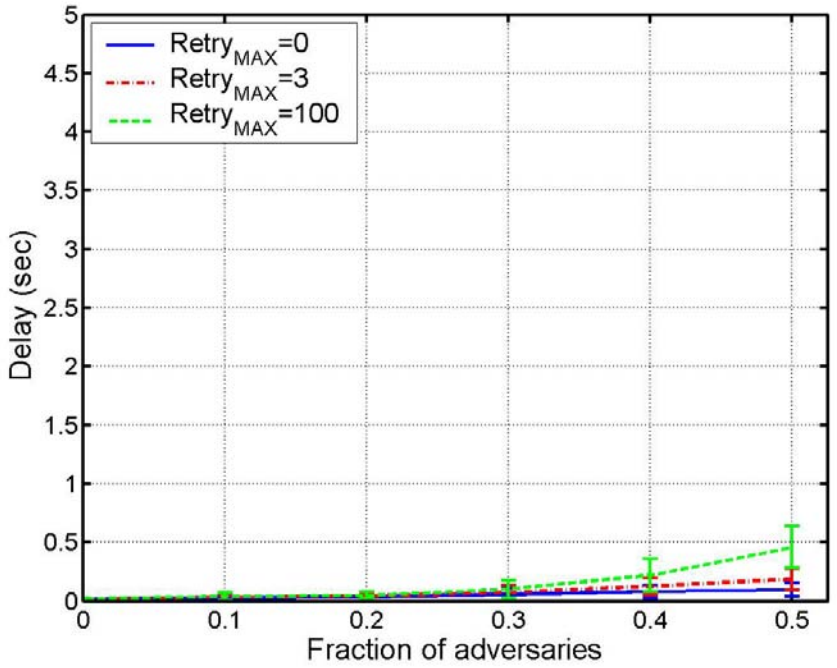
Performance Evaluation (cont'd)

- Secure Message Transmission (SMT) protocol
- Secure Single Path (SSP) protocol
- Secure route discovery for both protocols
  - Explicit, basic
    - Reactive, Proactive
    - SRP, SLSP
- Attack pattern
  - Full compliance with the route discovery
  - Discarding in–transit data packets

Performance Evaluation (cont'd)



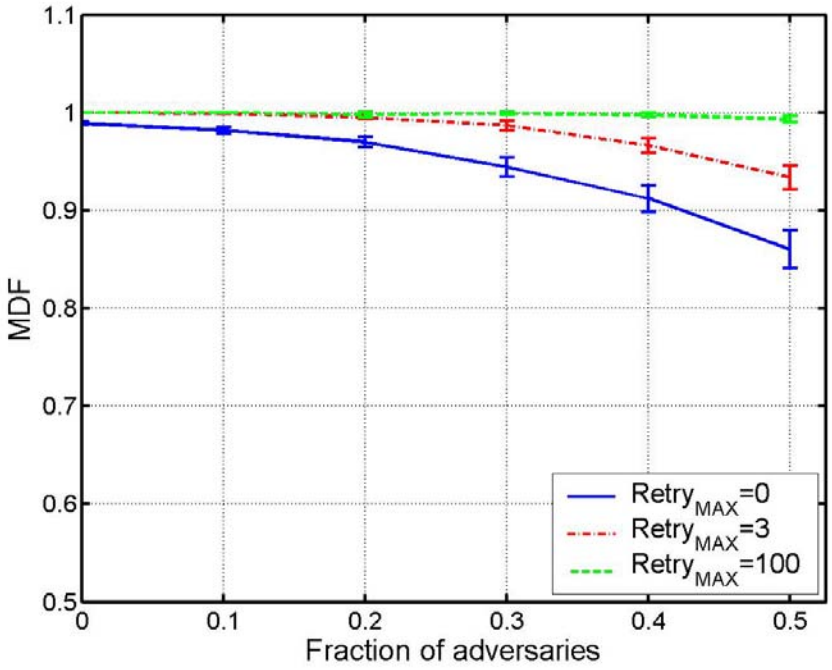
Message Delivery Fraction



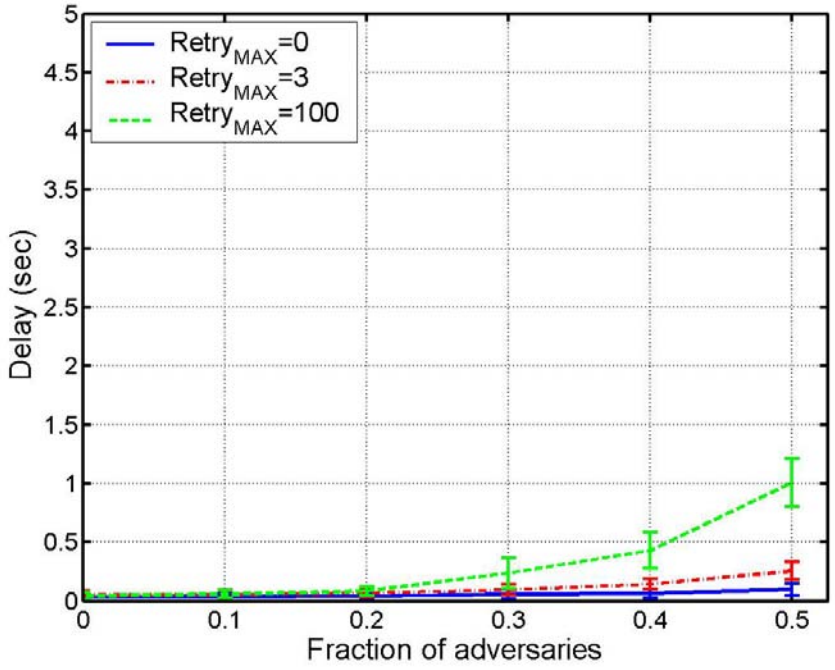
Message Delay

SMT-LS: SMT with a Link State Protocol

Performance Evaluation (cont'd)



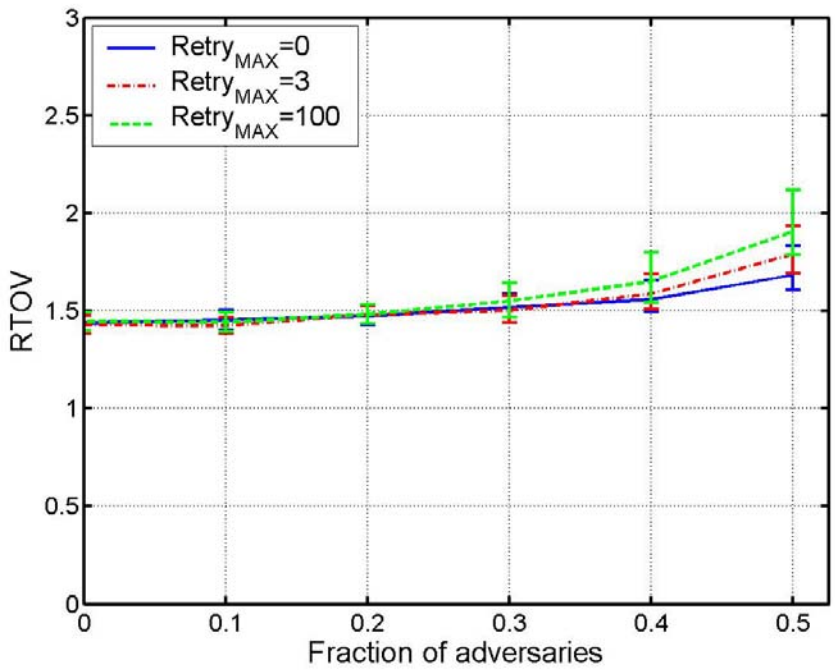
Message Delivery Fraction



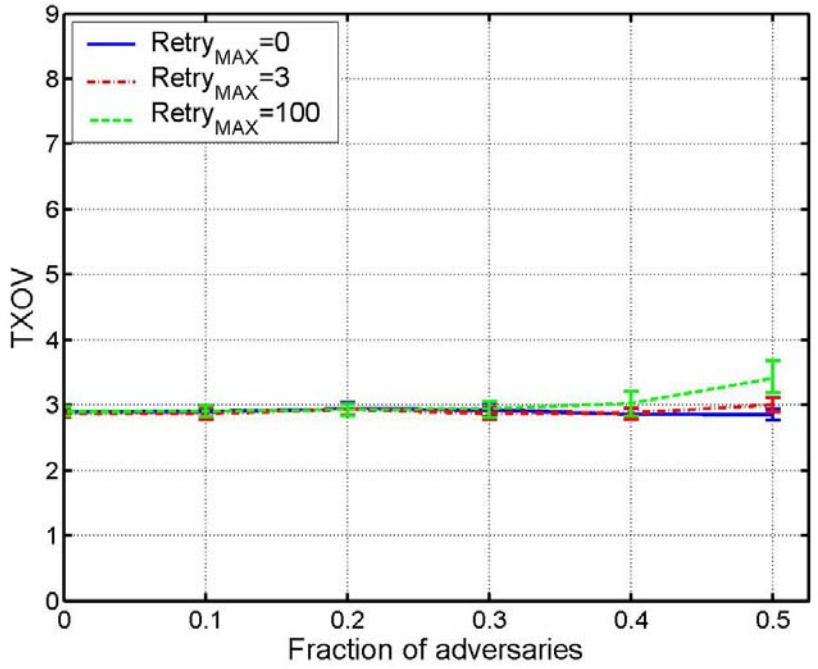
Message Delay

SMT-RRD: SMT with SRP

Performance Evaluation (cont'd)



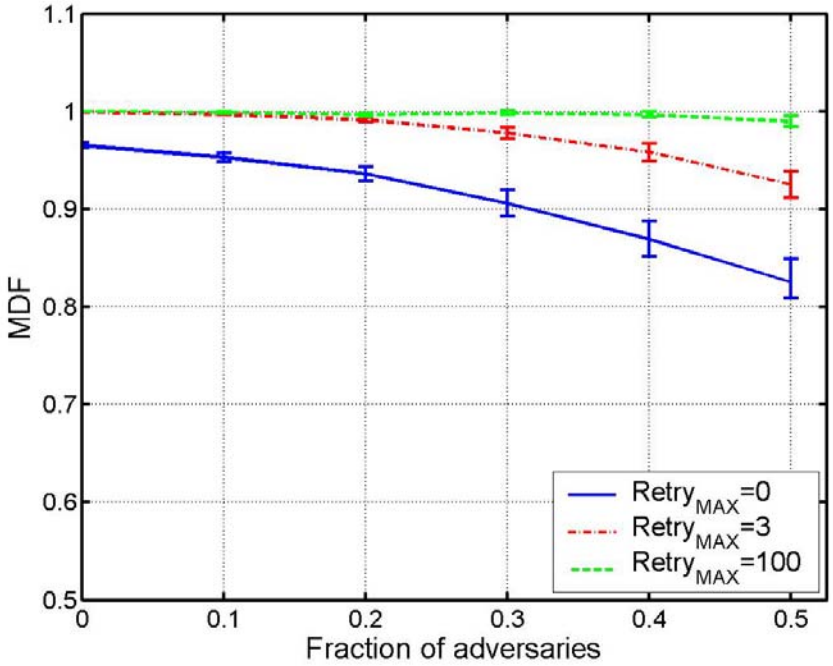
Routing Overhead

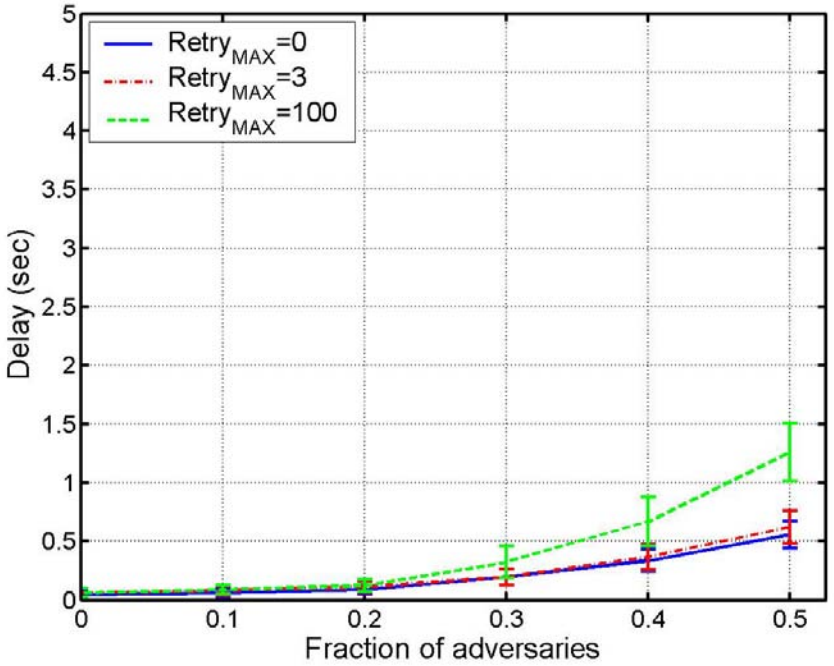


Transmission Overhead

SMT-RRD: SMT with SRP

Performance Evaluation (cont'd)



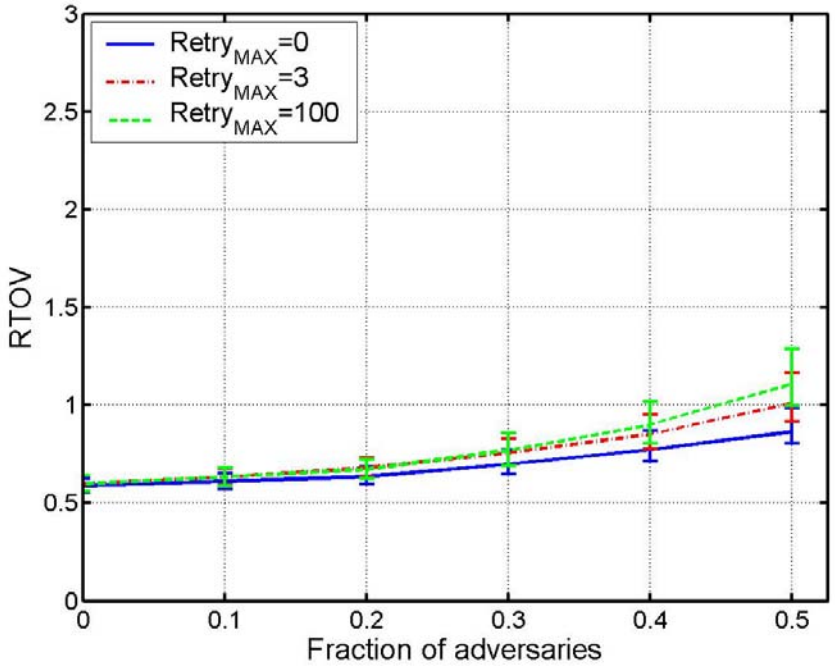


Message Delivery Fraction

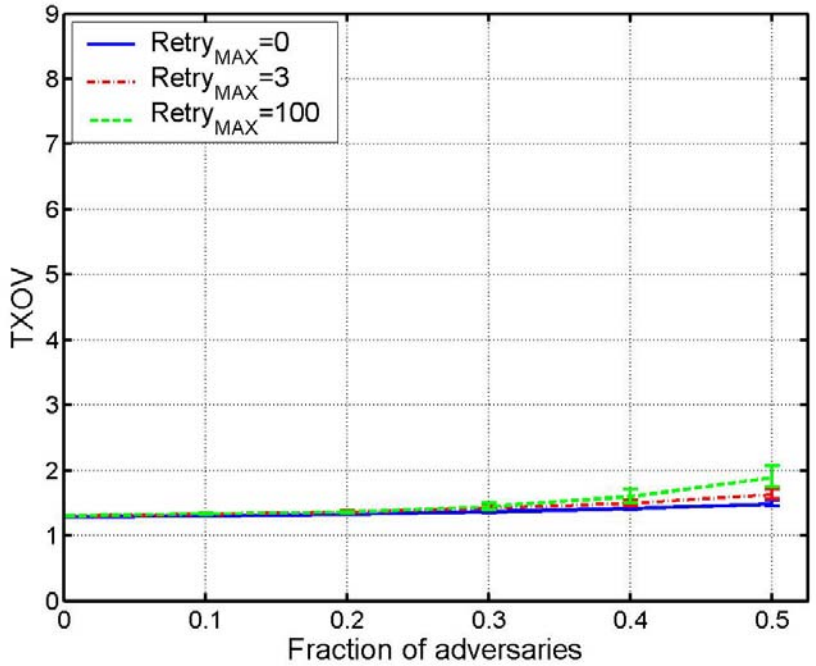
Message Delay

SSP-RRD: SSP with SRP

Performance Evaluation (cont'd)



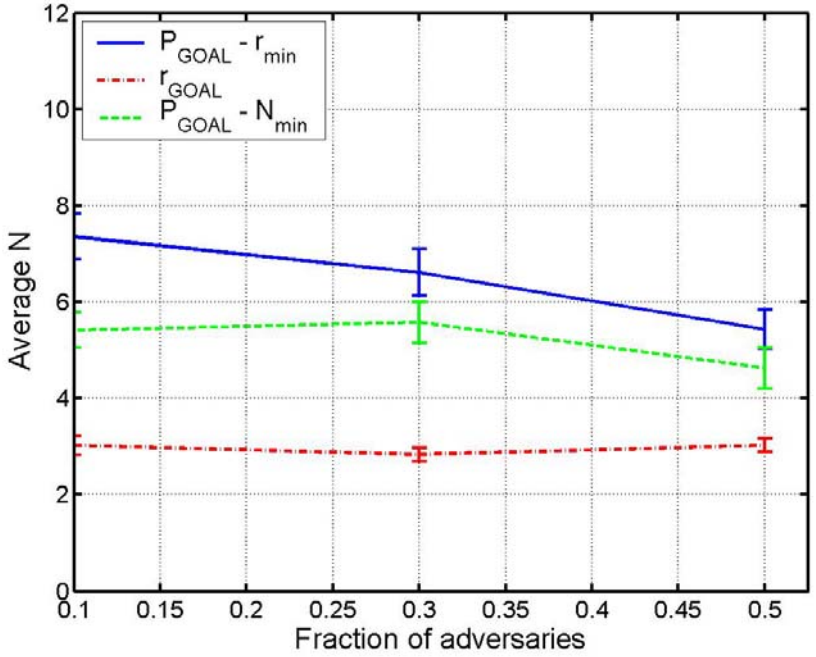
Routing Overhead

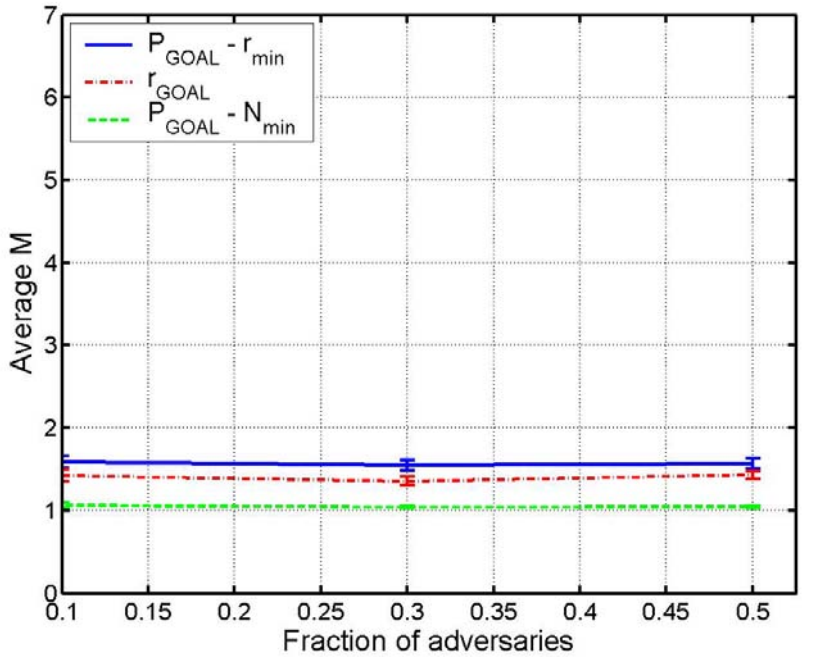


Transmission Overhead

SSP-RRD: SSP with SRP

Performance Evaluation (cont'd)

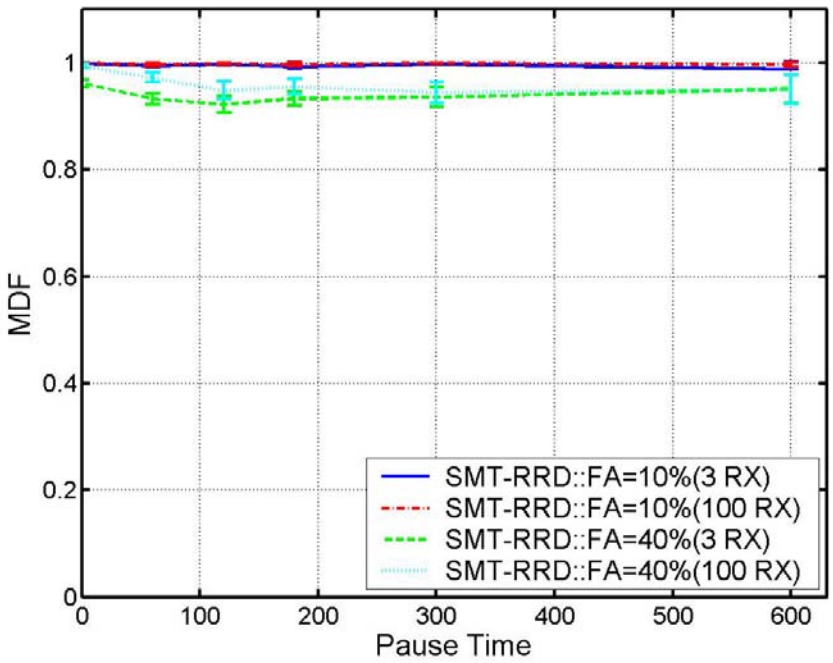




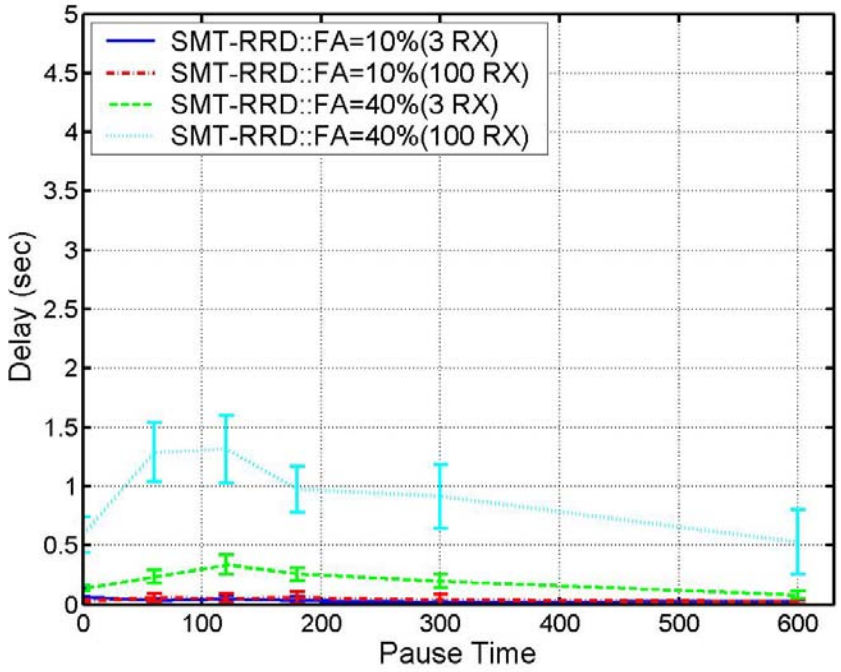
Average number of sent pieces (N) **Average number of required pieces (M)**

Transmission Redundancy

Performance Evaluation (cont'd)



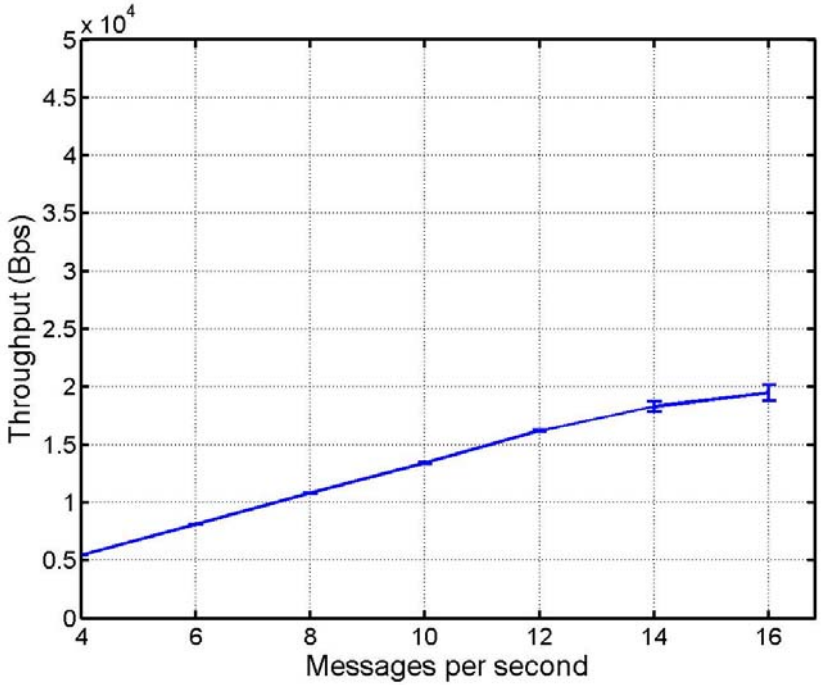
Message Delivery Fraction



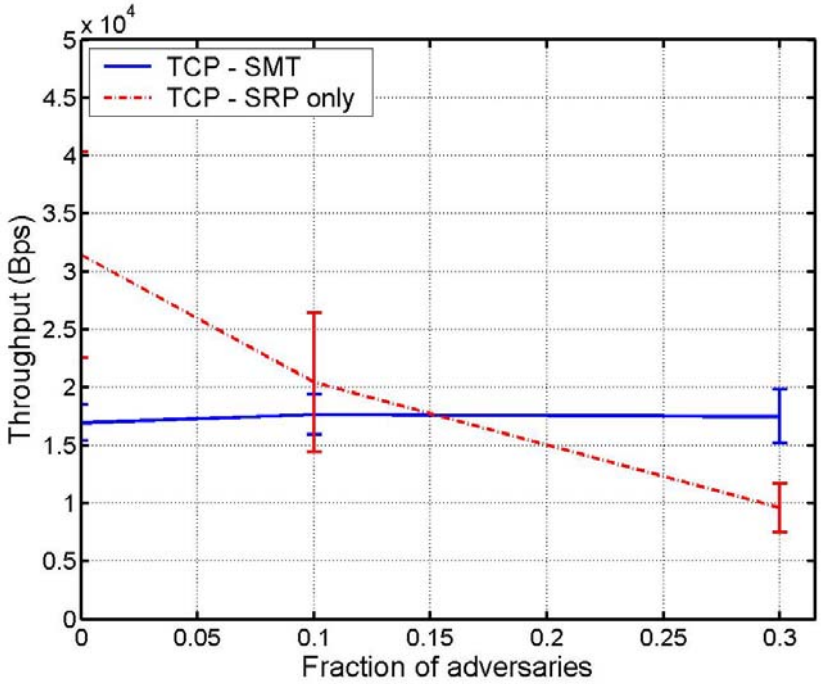
Message Delay

Impact of mobility; SMT-RRD

Performance Evaluation (cont'd)



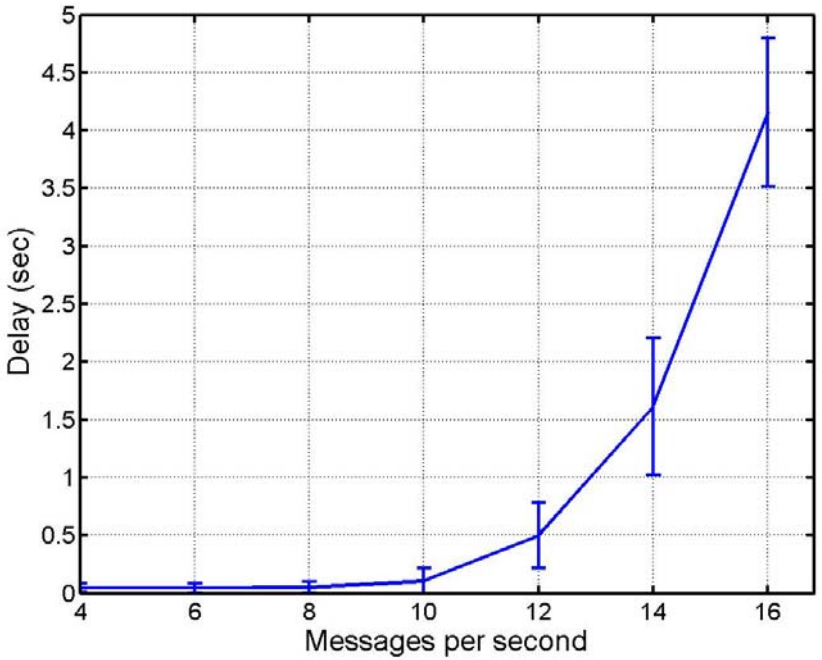
Throughput – no flow control

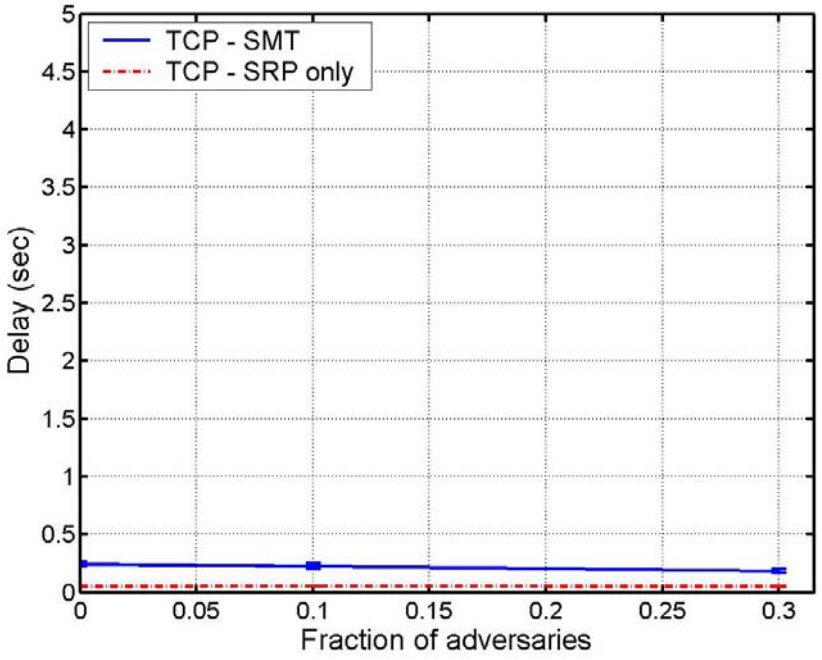


Throughput - SMT-RRD with TCP

Impact of Load and interaction with TCP

Performance Evaluation (cont'd)





Message delay – no flow control

Message delay - SMT-RRD with TCP

Impact of Load and SMT interaction with TCP

Summary

- Secure data communication is critical
  - Secure routing protocols are vulnerable
  - As long as attackers can place themselves on utilized routes, they can degrade or deny communication
  - The only answer is to assess whether data are delivered, and avoid non-operational routes
- Secure data communication is practical
  - Low-delay, low-jitter, and highly reliable; essentially, real-time
  - Flexible
  - Low overhead
  - End-to-end
  - Effective against any data-dropping pattern

Securing Ad Hoc Networks and Vehicular Communications

Part 2: Securing Vehicular Communications

Outline

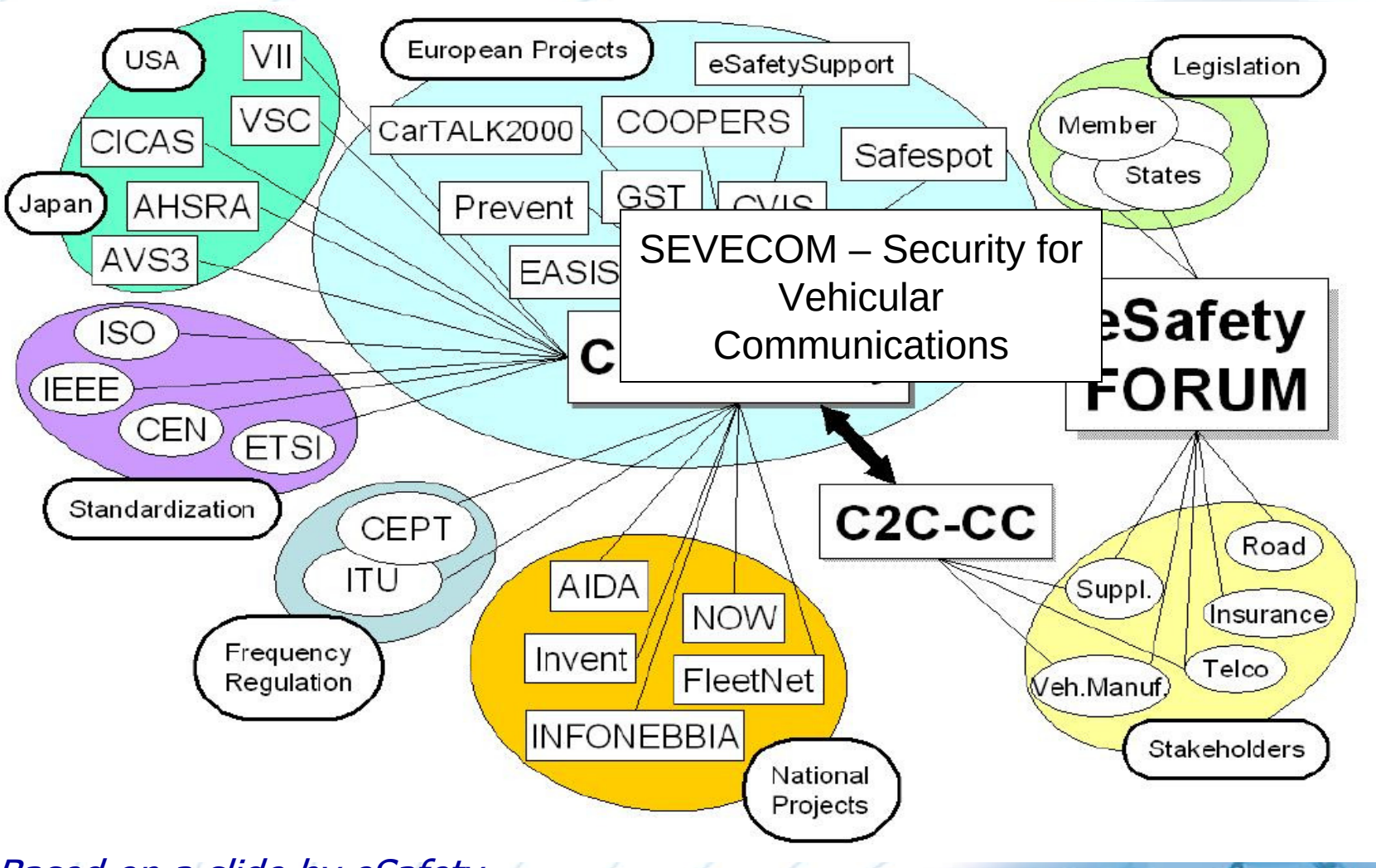
- Part 2 topics
  - Overview of vehicular communications
  - Vulnerabilities
  - Security requirements
  - Elements of security architecture for vehicular communication systems

Vehicular Communications (VC)

- Technology in the making
  - Mobile Ad Hoc Networking
    - Vehicular Ad Hoc Networks (VANET)
  - Infrastructure-based wireless communications
- Eventually wide, gradual deployment
- Interoperability
- Standardization

Vehicular Communications (VC) (cont'd)

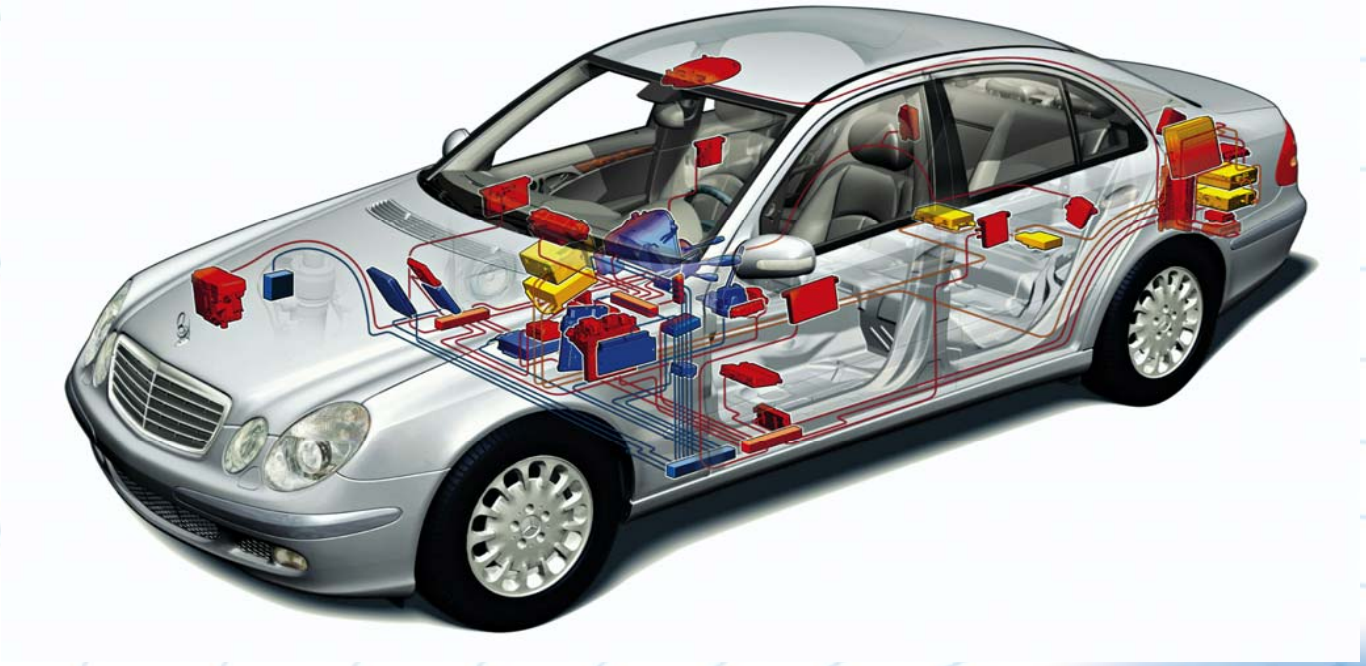
- VC research, development, and standardization



Based on a slide by eSafety

Vehicular Communications (VC) (cont'd)

- Vehicles equipped with
  - Computers
  - Sensors, including global positioning and navigation systems
  - Wireless transceivers



Vehicle graphic courtesy of DC

Vehicular Communications (VC) (cont'd)

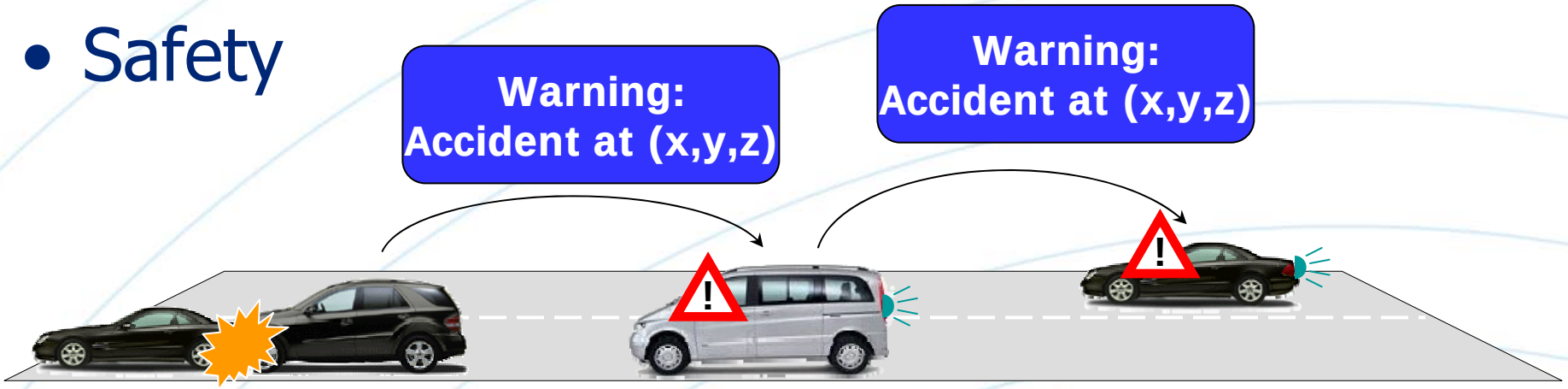
Illustration by the Car-to-Car Communication Consortium

Vehicular Communications (VC) (cont'd)

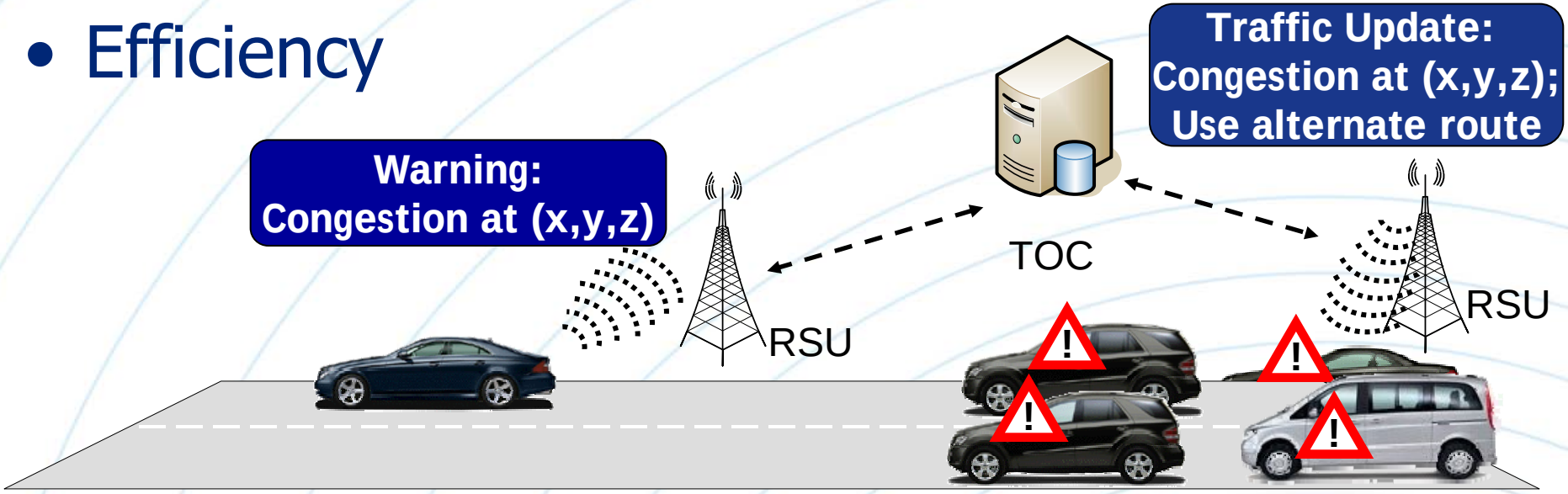
- Frequent, high-rate vehicle-to vehicle and vehicle-to-infrastructure communication
  - Periodic, triggered, dependent on network characteristics (e.g., density)
  - Example: a vehicle transmits safety messages every 100 to 300 milliseconds
  - Safety messages include vehicle-specific information; e.g., its coordinates

Vehicular Communications (VC) (cont'd)

- Safety

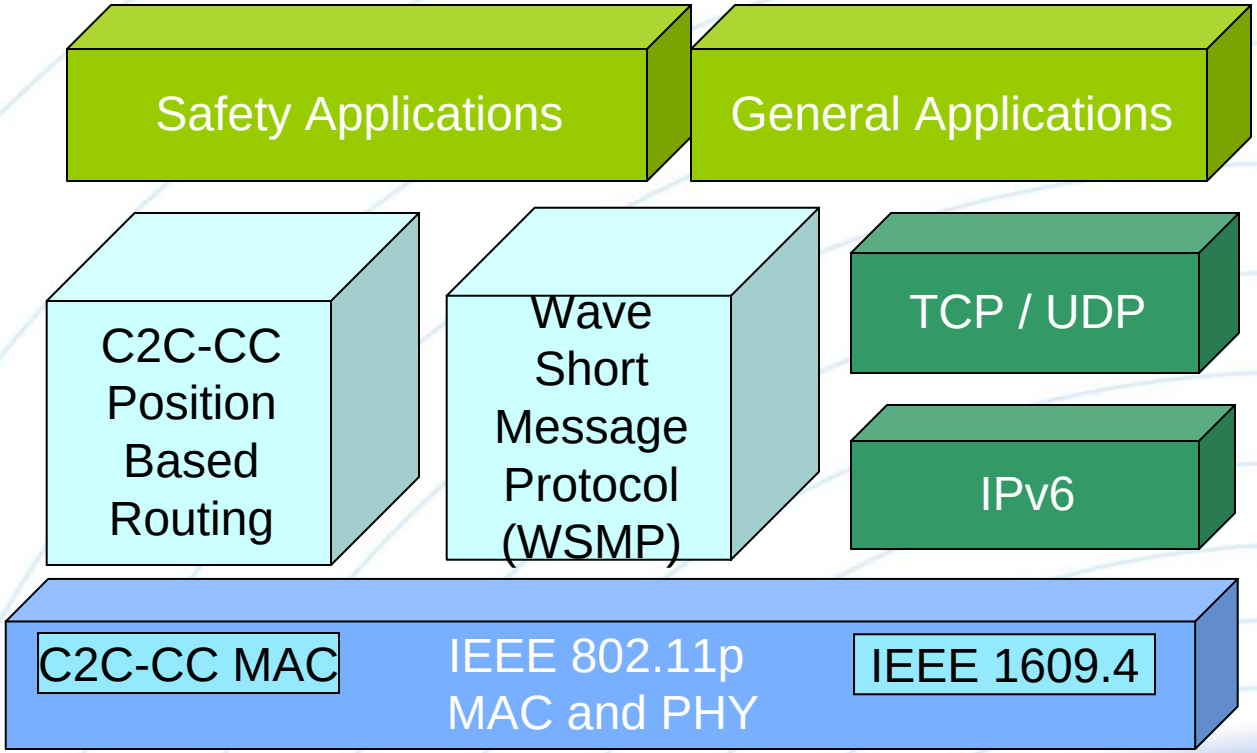


- Efficiency



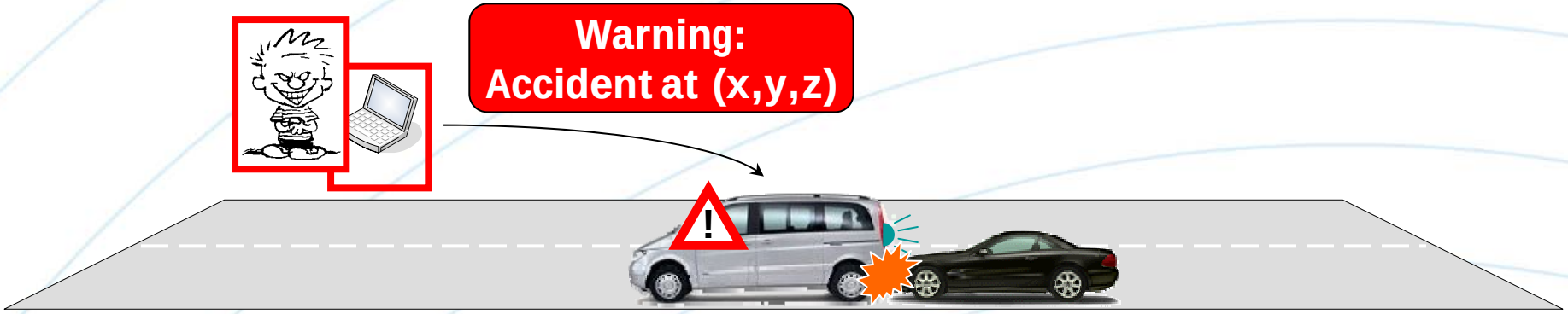
Vehicular Communications (VC) (cont'd)

- High rate broadcast communication
- VANET-only (e.g., safety) and TCP/IP communication

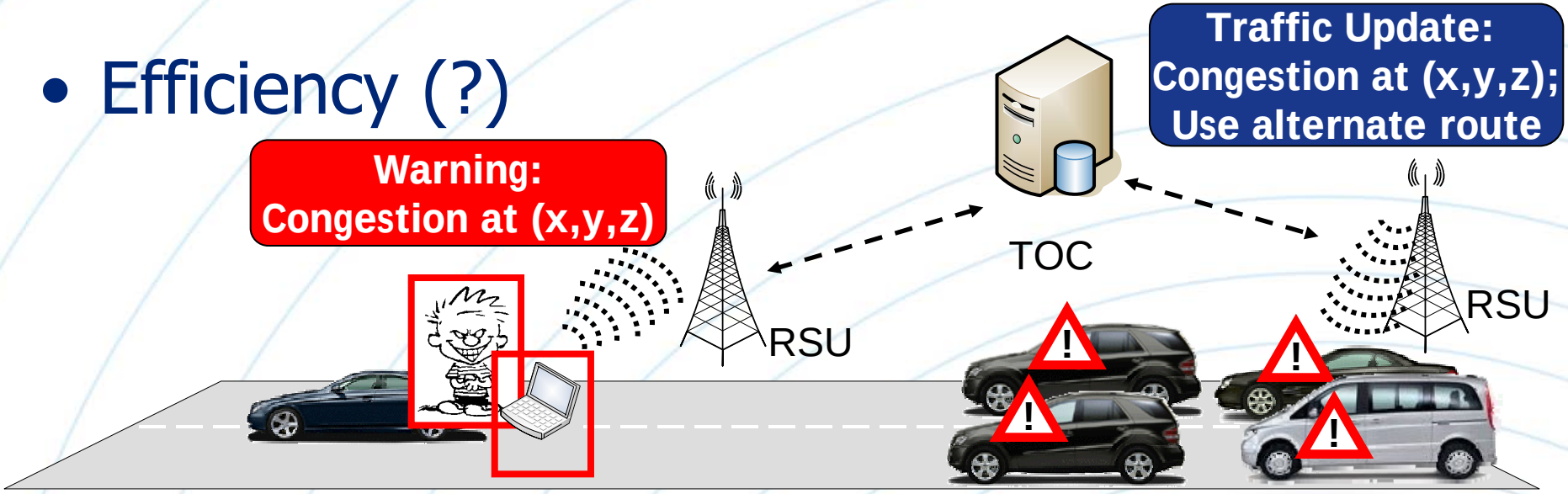


Security and Privacy – Why?

- Safety (?)

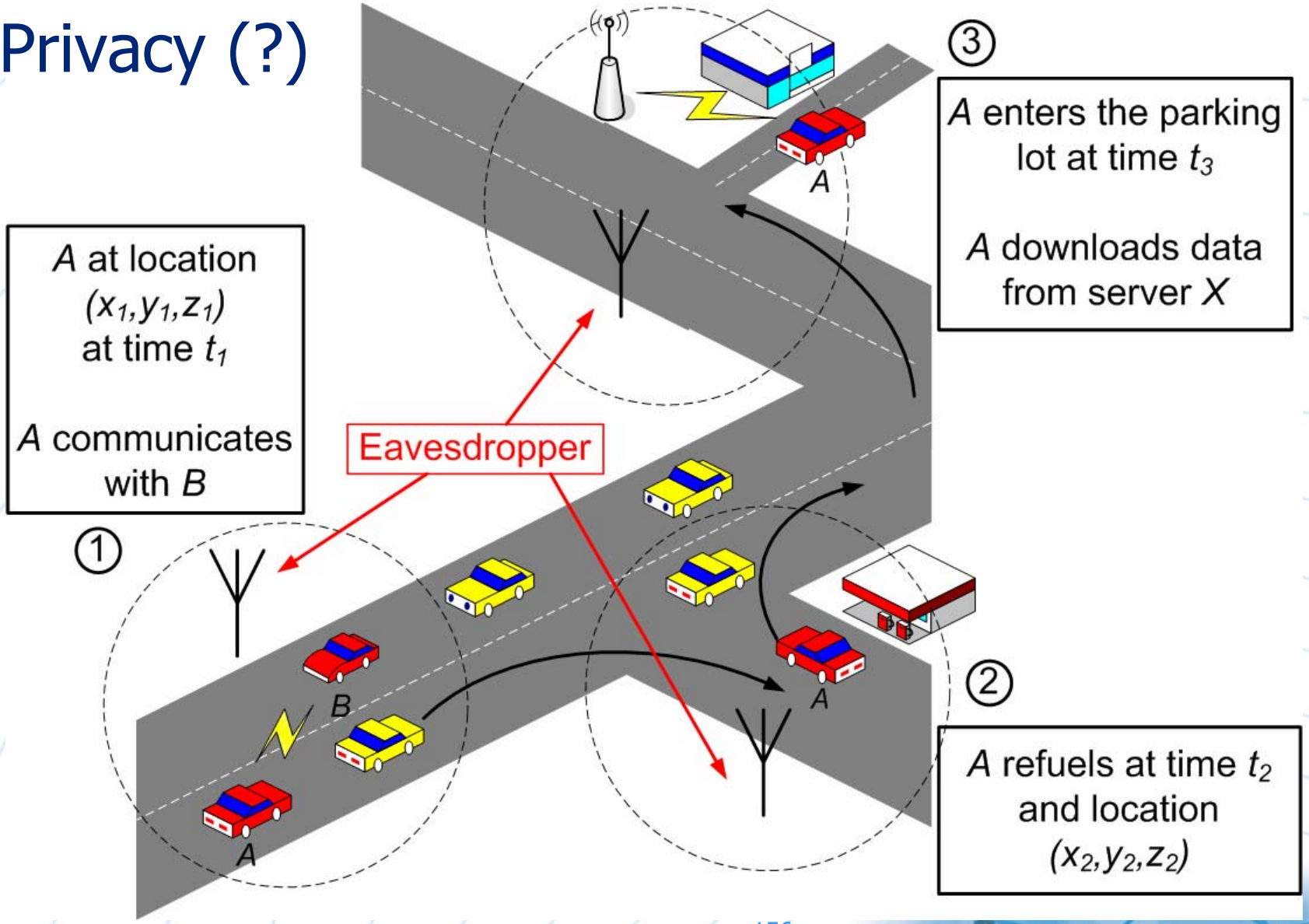


- Efficiency (?)



Security and Privacy for VC – Why?

- Privacy (?)



Security and Privacy for VC – Why?

- Without robust designs, VC systems may facilitate antisocial behavior
- The deployment of vulnerable VC systems may cancel out their envisioned benefits
- Abused, poorly defended VC systems can cause damages and high cost
- Attackers and adversaries will always be present

Adversary Model

- Any wireless device that implements a rogue version of the VC protocol stack can be an adversarial node
- Internal adversaries equipped with the system credentials
- Adversaries can forge and inject any message, modify in-transit messages, replay any received message

Adversary Model (cont'd)

- Input controlling adversary
  - Tamper with sensory inputs
  - Much easier that hacking with the VC system software
  - Control the node's behavior
- Adversarial parsimony
  - A small number/fraction of adversaries are more likely than a large number to be present in a network area
  - Adversaries are more likely to be independent than colluding

What makes VC security different?

- Complexity of the system
  - Hybrid (ad hoc, infrastructure) networking
  - Sensory inputs
- Tight coupling between users, applications, and network
- Pre-VC transportation systems and 'legacy' constraints and requirements
  - Liability identification
- Large scale and high mobility
- Stronger privacy concerns

Secure VC

- Requirements
  - Authentication, integrity, non-repudiation, access control, confidentiality, availability
  - Privacy
  - Liability identification
- System and adversary model
- Design principles

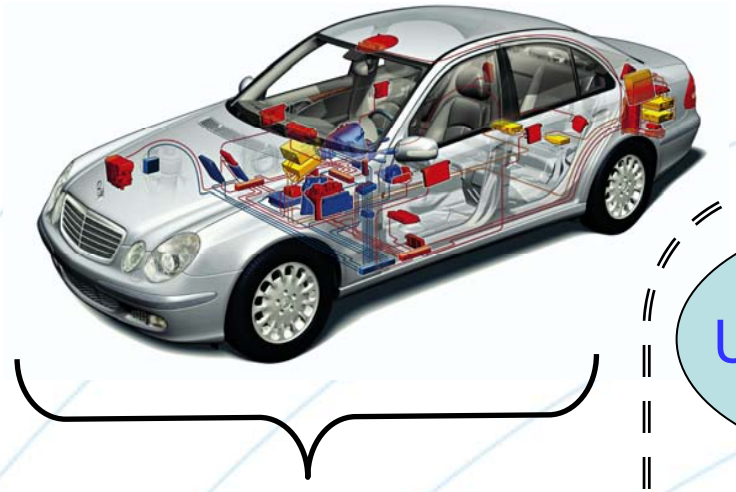
P. Papadimitratos, V. Gligor, J.-P. Hubaux, "Securing Vehicular Communications – Assumptions, Requirements and Principles," ESCAR 2006

Secure VC

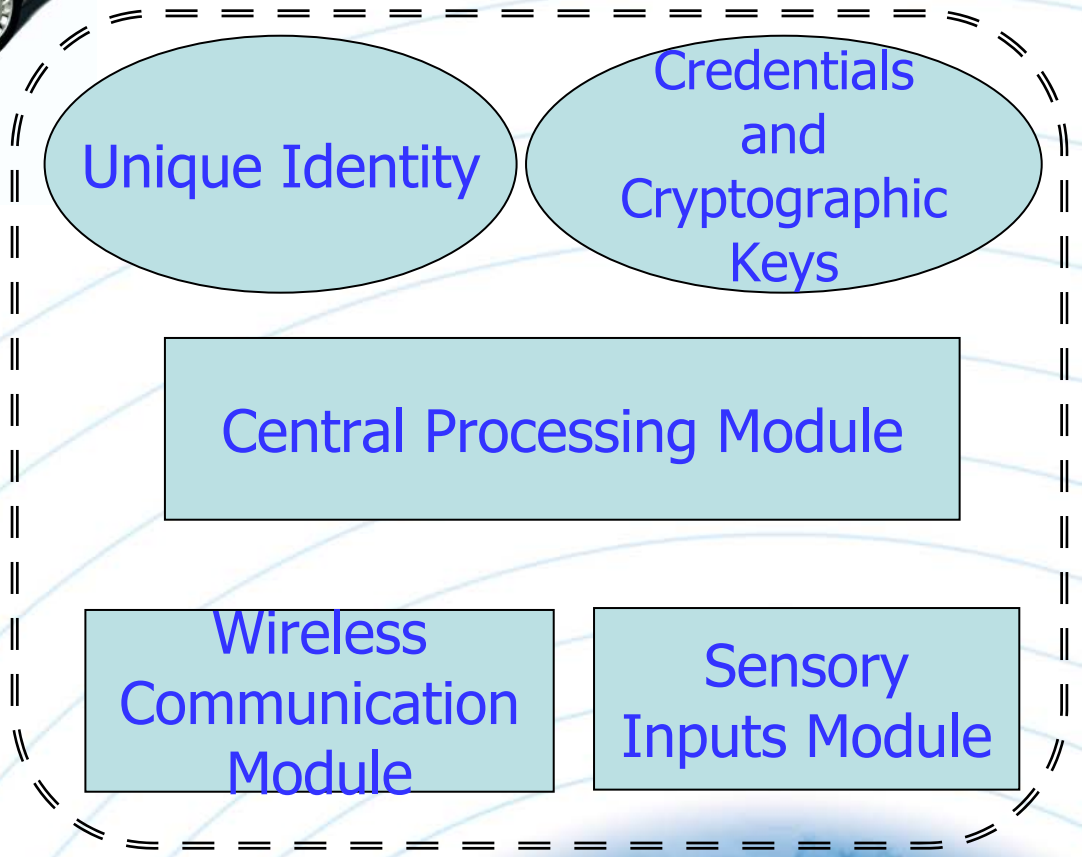
- Authorities
  - Trusted entities
  - Issuing and managing identities and credentials
- Network nodes
  - Vehicles
    - Public
    - Private
  - Road-side units
- Users

Secure VC (cont'd)

Graphic courtesy of DC



Abstract view of a vehicle in a (secure) vehicular communications system



Secure VC (cont'd)

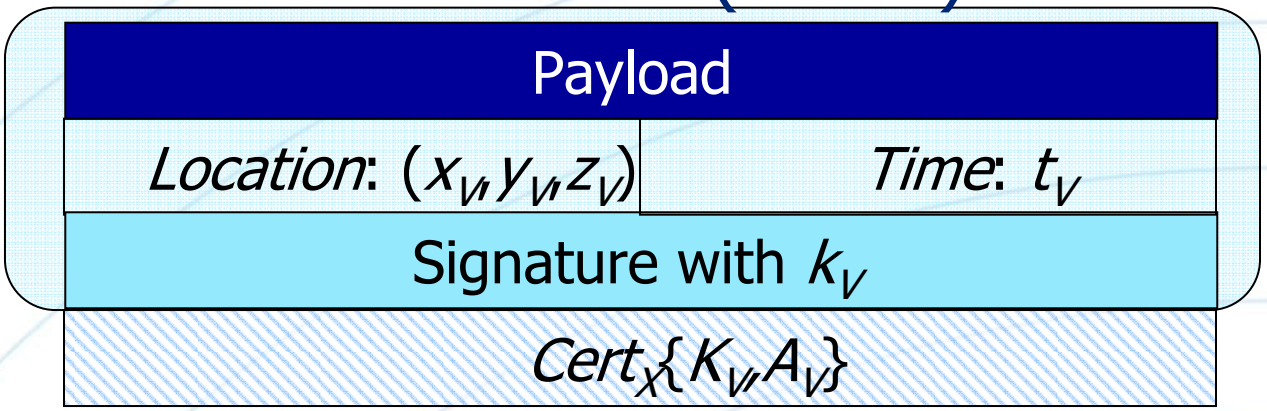
- Node Identity
  - Unique identity V
  - Integration of pre-VC and VC-specific identifiers
- Node Keys
  - Public / private key pair K_V, k_V
- Node Credentials
  - Certificate $Cert_X\{K_V, A_V\}$
  - A_V: attributes of node V
- Long-term identification

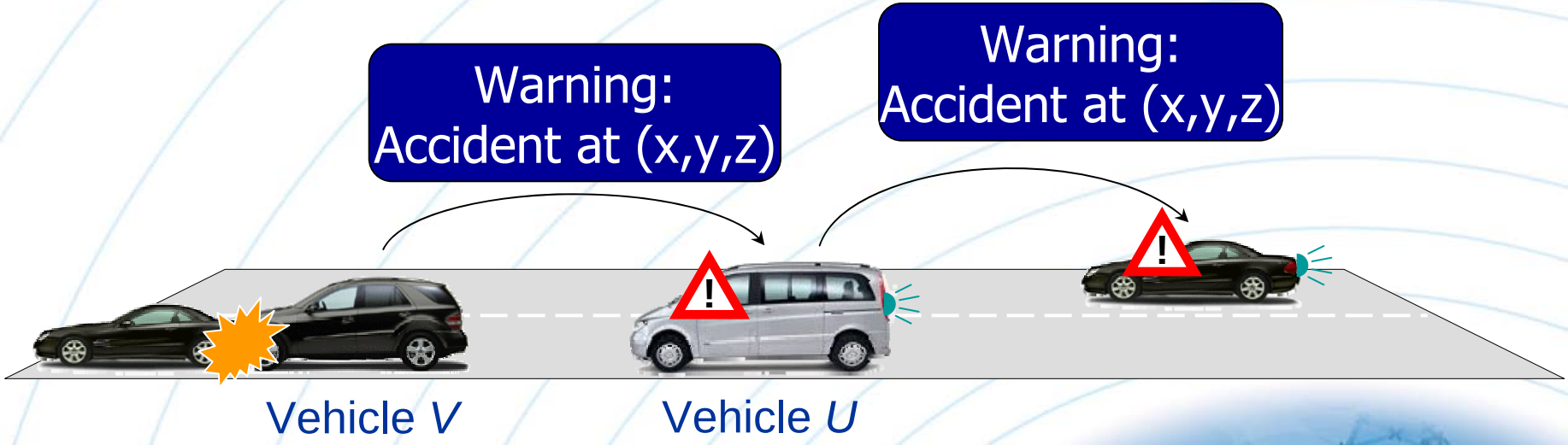
Secure VC (cont'd)

- Secure Communication
  - Single- and Multi-hop
  - Vehicle to vehicle
  - Vehicle to infrastructure
- Digital signatures more appropriate tool
  - Any-to-any communication; e.g., broadcast, geo-cast
  - High mobility
- Relatively simple networking protocols 'shift' the security focus to the application

Secure VC (cont'd)

- Secure Communication (cont'd)





A closer look to privacy concerns

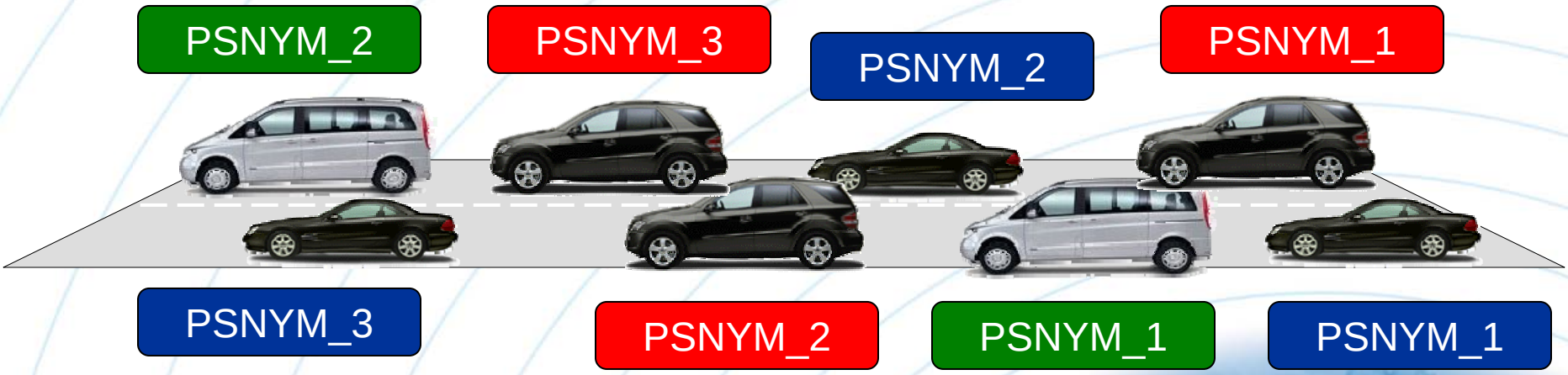
- Communication cannot be regulated or controlled by the node/user
  - Safety messaging will be essentially an 'always-on' application
- Vehicle-originating wireless transmissions are particularly easy to eavesdrop
  - Data link very similar to a widely adopted technology: IEEE 802.11p
  - Very large and increasing numbers of 802.11 access points already deployed
  - Road-side infrastructure deployed for other services could be subverted into acting as an eavesdropper
- Linking messages to the transmitting vehicle and inferring private information about its passengers

A closer look to privacy concerns (cont'd)

- What are we after?
  - At least the same degree of privacy achieved nowadays, before the advent of vehicular communications
  - Combination of strong security and privacy-enhancing technologies
  - Ideally, anonymous and authentic communications, but:
    - High processing and communication overhead
    - Often, messages from the same vehicle should be linkable
  - Requirement: messages generated by a given vehicle can be linked at most over a protocol-selectable period of time
    - The shorter this period, the harder to track a vehicle becomes
  - Privacy Enhancing Technologies (PET) for VC

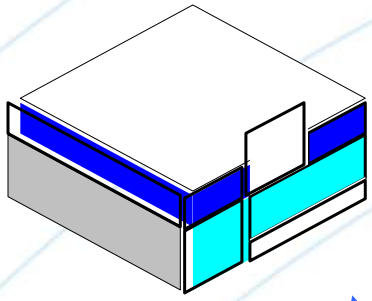
PET for VC (cont'd)

- **Pseudonym**: Remove all identifying information from certificate
- Equip vehicles with multiple pseudonyms
  - Alternate among pseudonyms over time (and space)
  - Sign message with the private key corresponding to pseudonym
  - Append current pseudonym to signed message



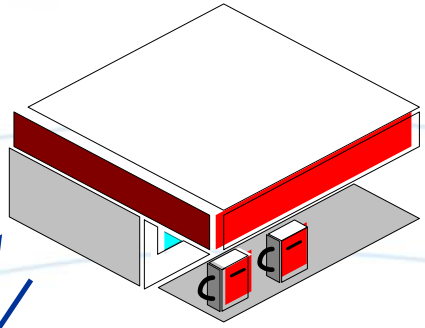
PET for VC (cont'd)

- PET system setup



Authority X

Long-term Identification



Authority A

Pseudonym Provider

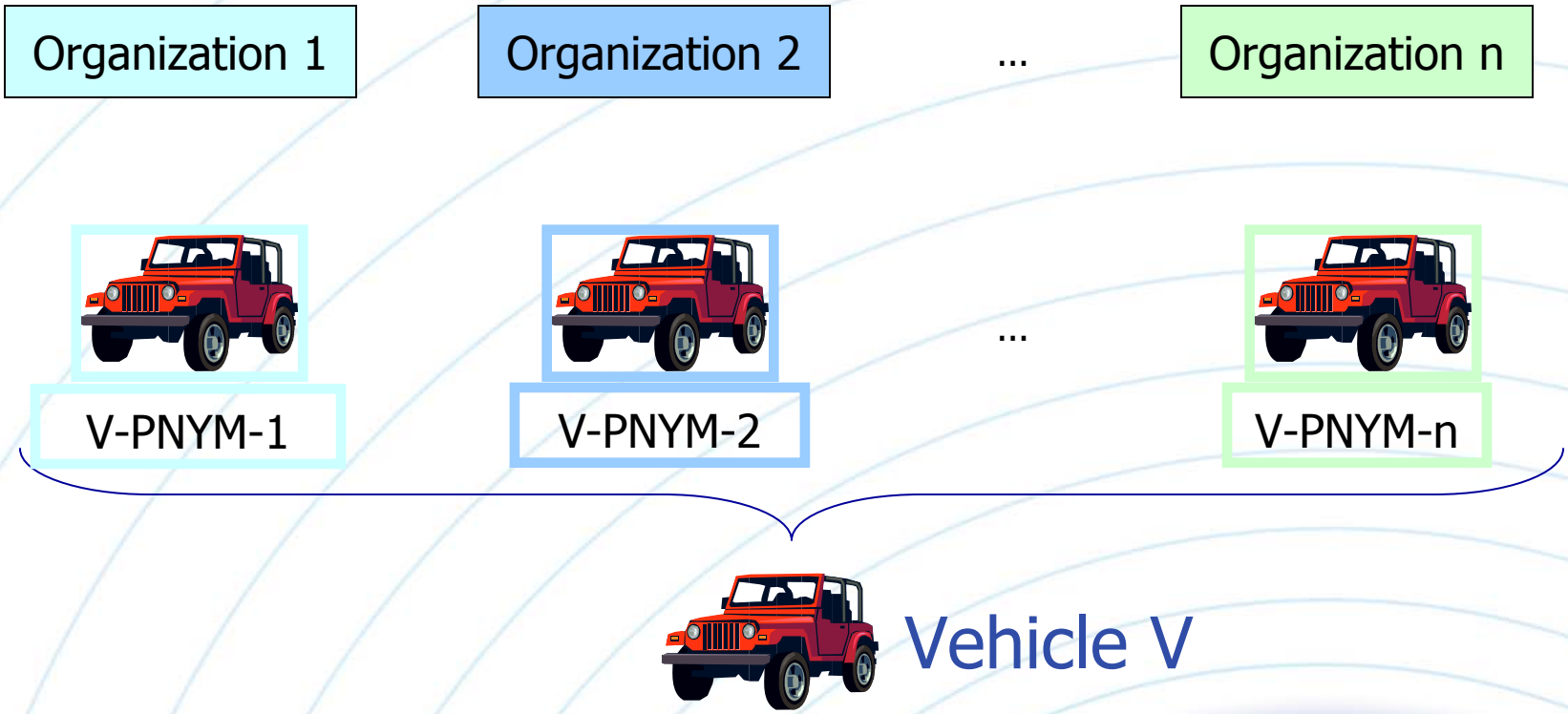
Vehicle *V*

PSNYM_1, ..., PSNYM_k

Set of pseudonyms for *V*

PET for VC (cont'd)

- PET system setup (cont'd)
  - Multiple pseudonym providers

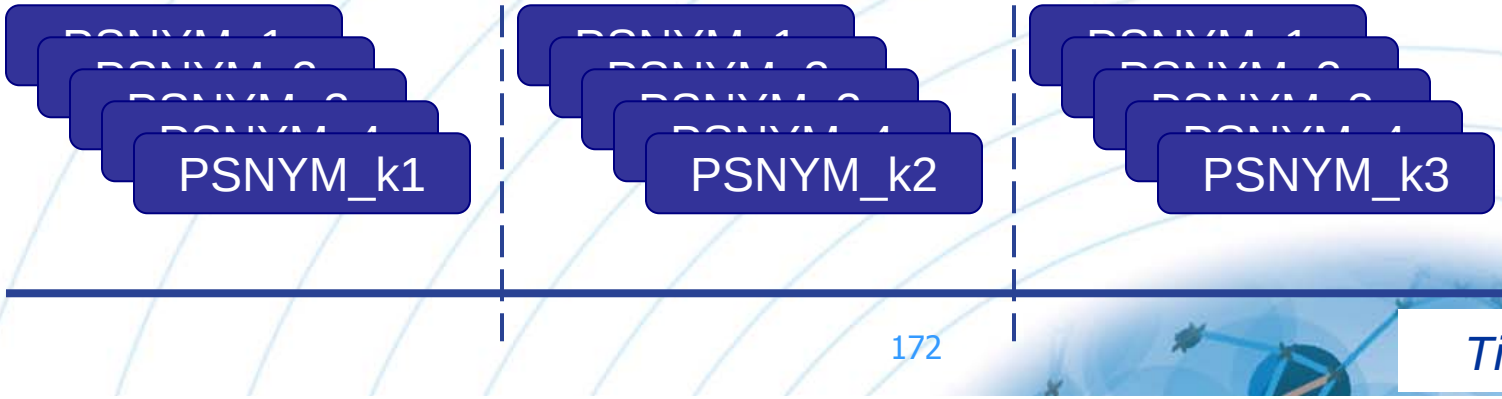


PET for VC (cont'd)

- Pseudonym format

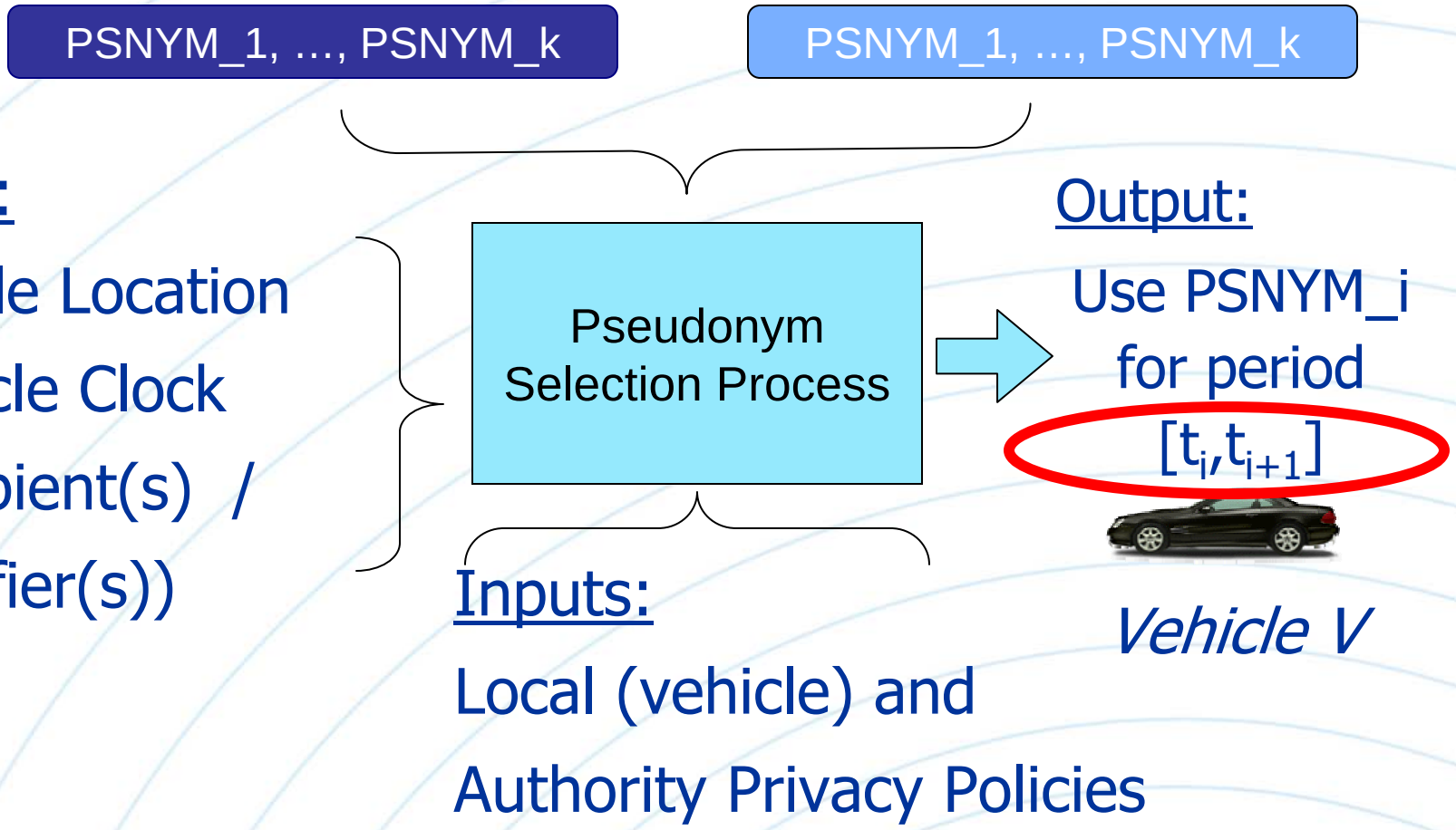
PSNYM-Provider ID	PSNYM Lifetime
Public Key K_i	
PSNYM-Provider Signature	

- Supplying vehicles with pseudonyms
  - Sufficient in number
  - Periodic 'refills'



PET for VC (cont'd)

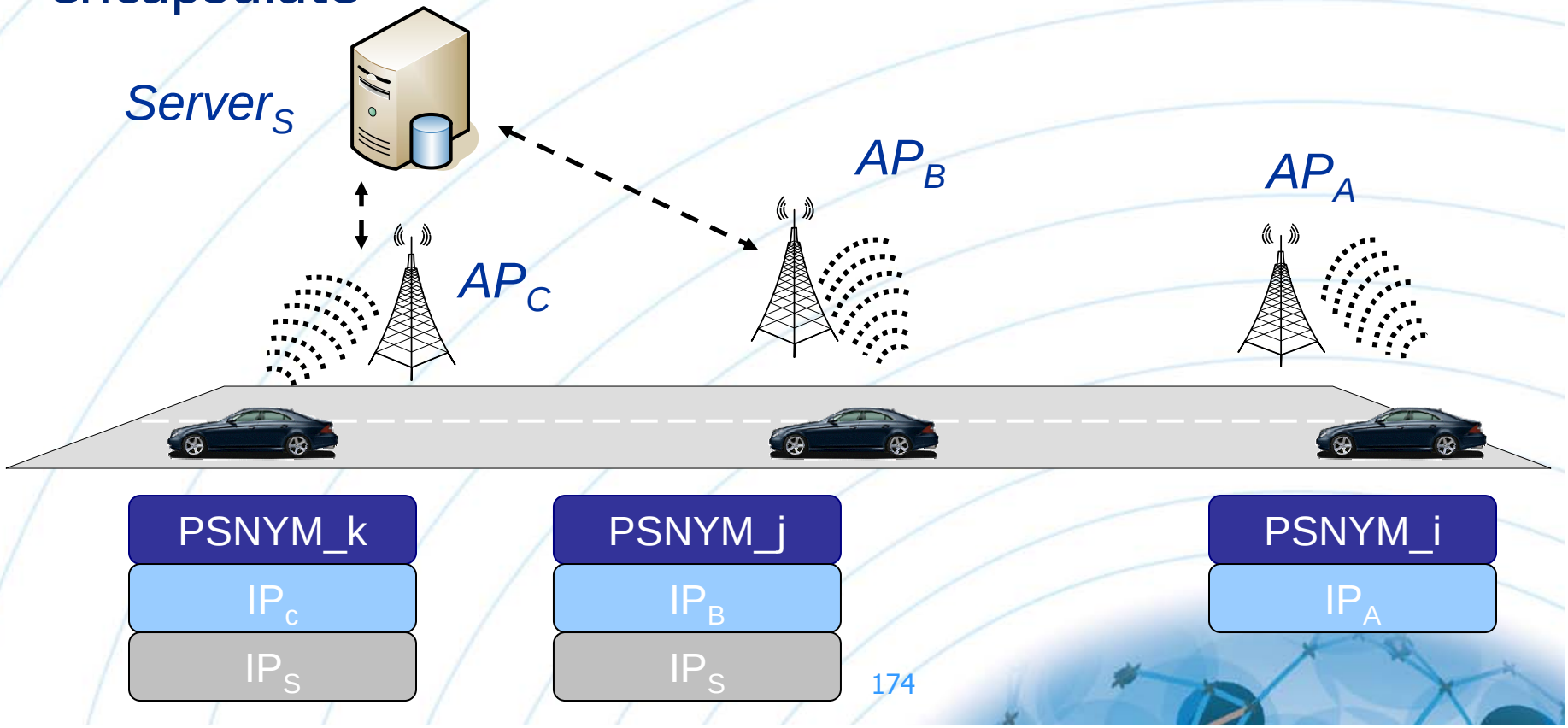
- Pseudonym Change Mechanism



- *One pseudonym per day (?)*
- *One per transaction (?)*

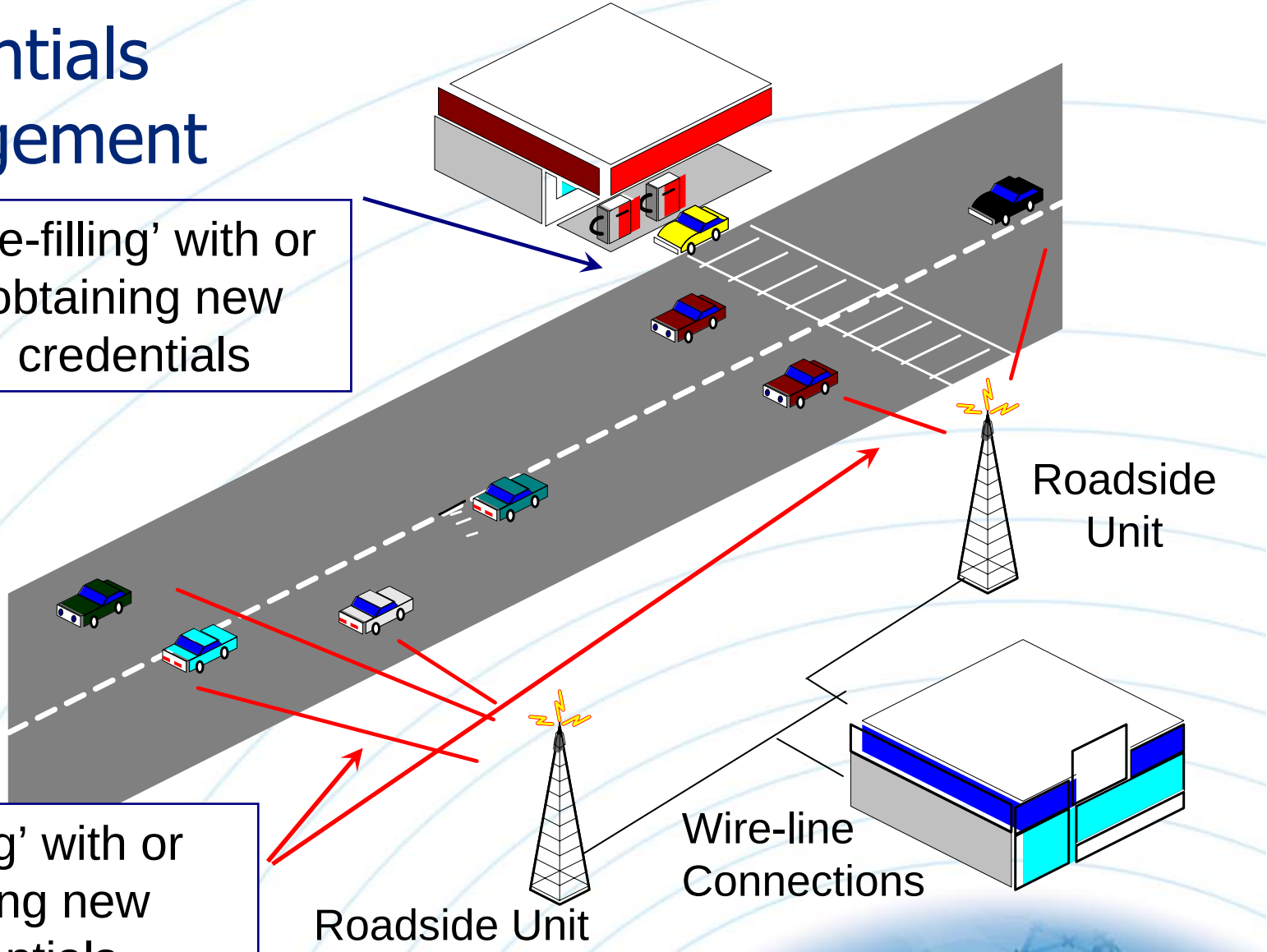
PET for VC (cont'd)

- Other vehicle network identifiers: e.g., IP and MAC addresses
- Change addresses along with pseudonyms
- Maintain addresses only when necessary, but encapsulate



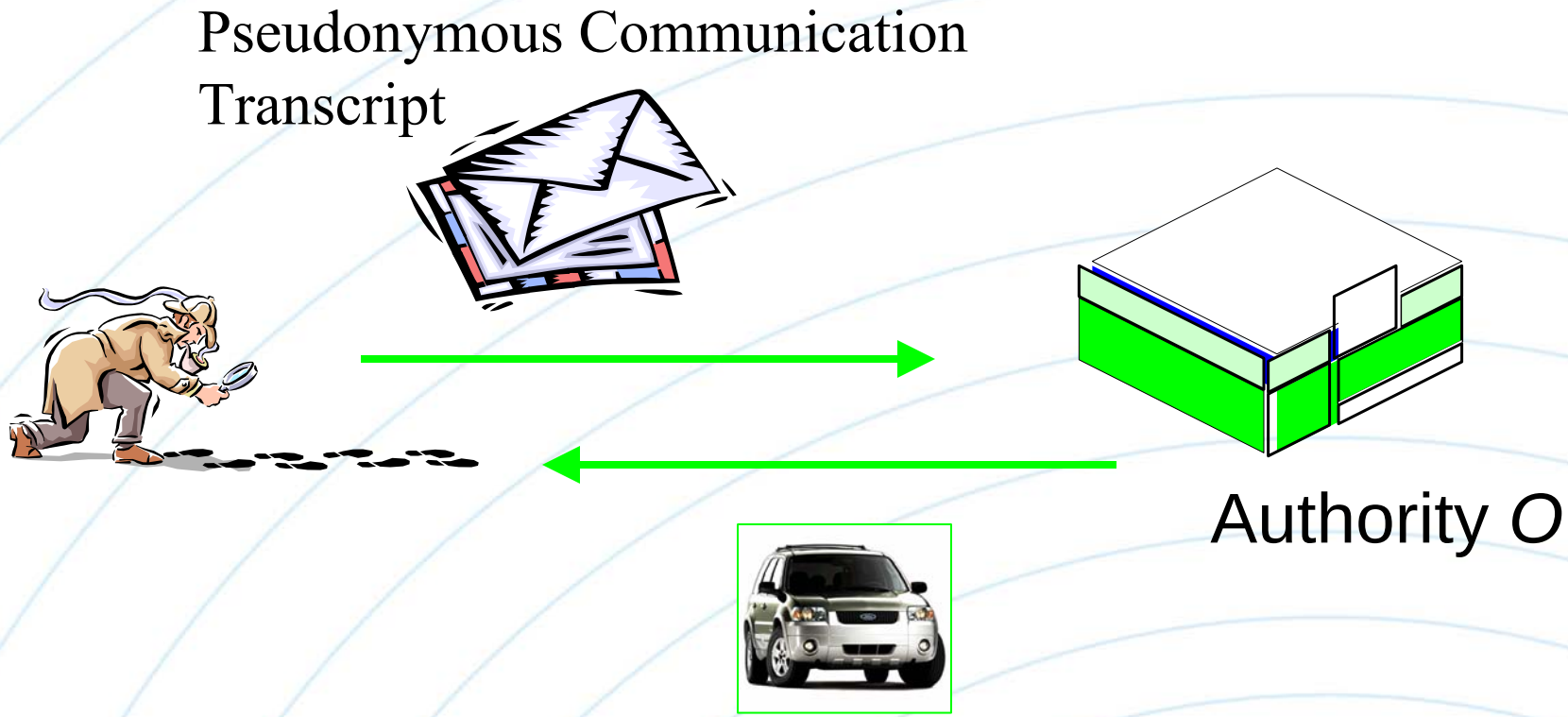
PET for VC (cont'd)

- Credentials Management



PET for VC (cont'd)

- Pseudonym Resolution



PET for VC (cont'd)

- Challenge
  - Managing a pseudonymous authentication system is cumbersome
    - Preload large numbers of pseudonyms or obtain them on-the-fly
    - Costly computations at the side of the pseudonym provider
    - Costly wireless communication to obtain pseudonyms
    - Need reliable access to the pseudonym provider
- Solution
  - On-board generation of pseudonyms
  - G. Calandriello, P. Papadimitratos, A. Lloy, and J.-P. Hubaux, "Efficient and Robust Pseudonymous Authentication in VANET," VANET 2007

Summary

- VC emerging as a convincing large-scale instantiation of mobile ad hoc networking
- Security and privacy-enhancing mechanisms are a prerequisite for the VC systems deployment
- Securing VC systems is a complex yet 'real' problem that attracts the attention of the community
- Opportunity: Awareness and joint efforts in industry and academia

Acknowledgements

NCCR MICS
National Competence Center In Research
Mobile Information and Communication Systems

Thank you!

Questions?

panos.papadimitratos@epfl.ch

http://people.epfl.ch/panos.papadimitratos