



# **GIAO THỨC AN TOÀN MẠNG**

## **Bài 4.2. Bộ giao thức IPSec (...)**

**TS. Trần Thị Lượng**

1

Giao thức ESP

2

Kết hợp AH & ESP

3

Giao thức IKE

# Mục tiêu bài học

## ❑ Kiến thức

- Hiểu được cơ chế hoạt động của giao thức ESP, sự kết hợp của giao thức AH & ESP,
- Hiểu các cơ chế hoạt động của giao thức IKE

## ❑ Kỹ năng

- Phân tích hoạt động của các giao thức ESP, kết hợp AH & ESP ở các chế độ Transport hoặc Tunnel qua việc chặn bắt lưu lượng mạng.

# Tài liệu tham khảo

1. Giáo trình "Giao thức an toàn mạng máy tính">// Chương 3 "**Các giao thức bảo mật mạng riêng ảo**", năm 2013.
2. Giáo trình "An toàn mạng riêng ảo", năm 2007.
3. William Stalling, **Cryptography and Network Security Principles and Practice (5e)**//Part 3, chapter 16 – pp. 483- 527, Prentice Hall, 2011

1

Giao thức ESP

2

Kết hợp AH & ESP

3

Giao thức IKE

# Giao thức ESP

- ESP (Encapsulating Security Payload):
  - Là giao thức đóng gói tải an toàn của IPSec
  - Đảm bảo tính:
    - Toàn vẹn
    - Xác thực
    - Bí mật (mã hóa)

# Giao thức ESP

- Trong IPSec version 1: ESP chỉ cung cấp mã hóa cho phần Payload.
- Trong IPSec version 2: ESP cung cấp cả xác thực, toàn vẹn, mã hóa.
- Gói IP sau khi tiêu đề ESP được thêm vào như trong hình vẽ



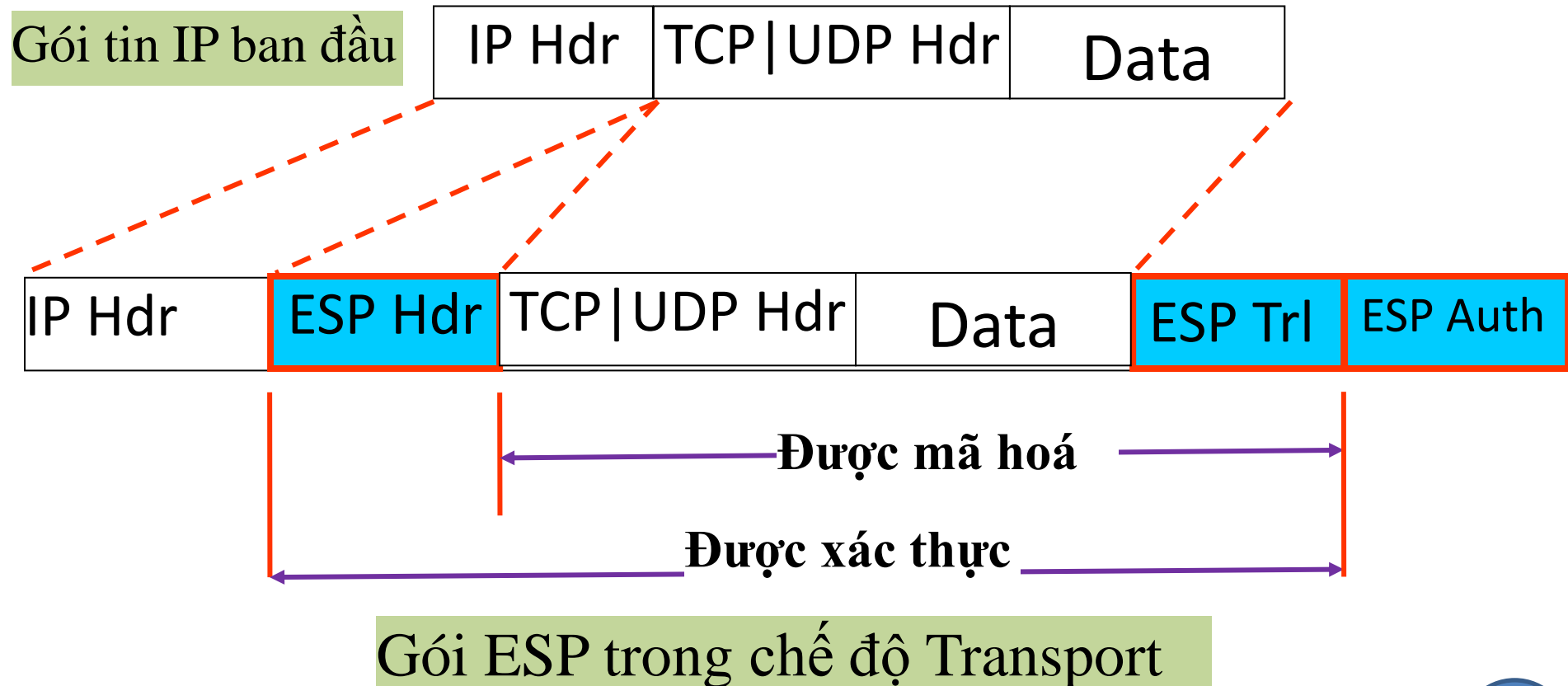
# Giao thức ESP

- Chế độ hoạt động:
  - ESP cũng được sử dụng ở 2 chế độ:
    - Transport :
      - » Dùng IP Header gốc
      - » Chỉ mã hóa và/hoặc đảm bảo toàn vẹn cho nội dung gói tin và một số thành phần ESP, nhưng không có IP Header.



# Giao thức ESP

- Chế độ Transport:

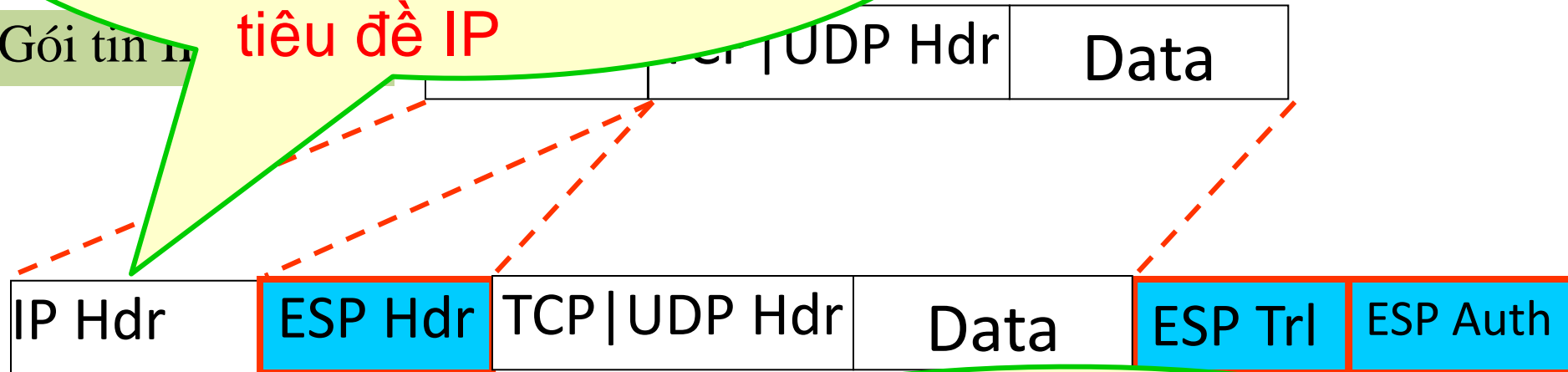


# Giao thức ESP

## Chế độ Transport:

Cho phép bảo vệ  
giao thức lớp trên  
nhưng không bảo vệ  
tiêu đề IP

Gói tin n



Gói

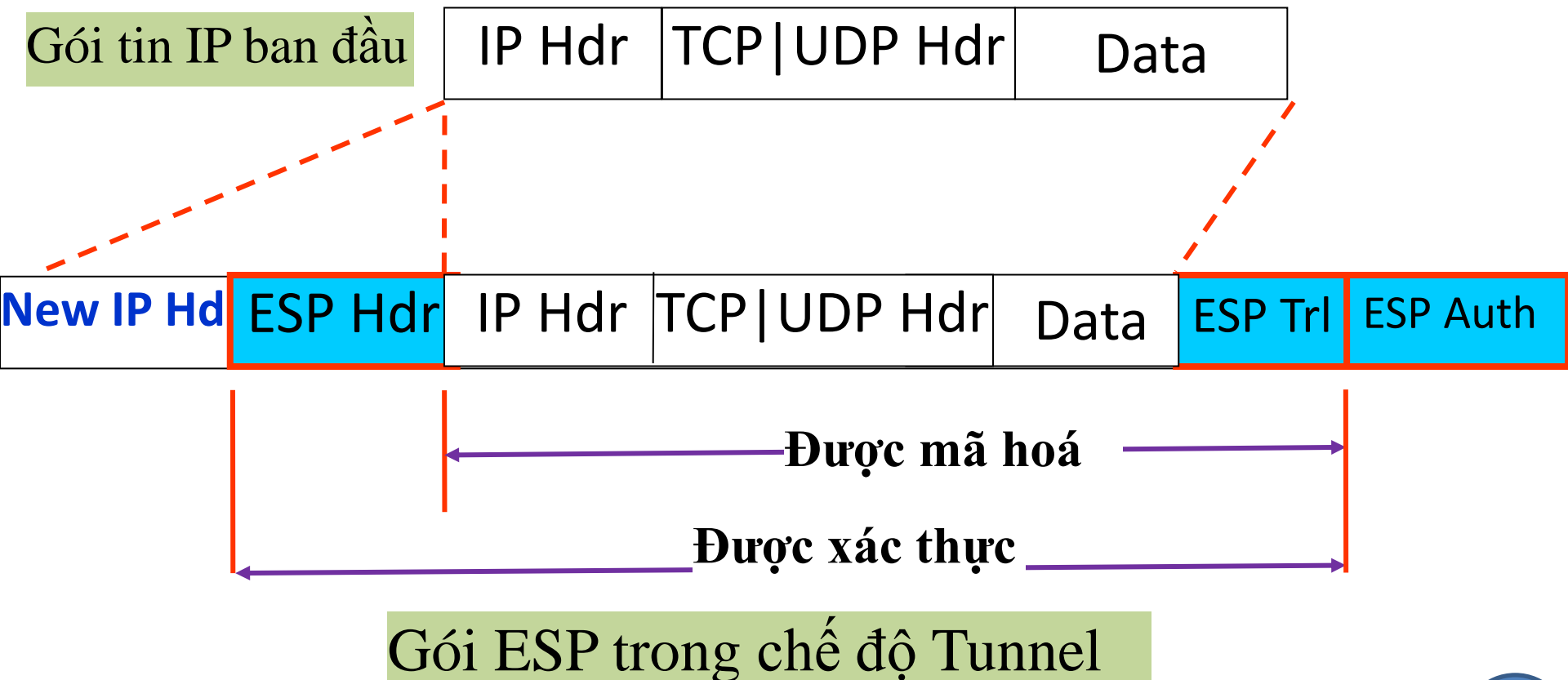
**Chế độ Transport:** Gói IP ban đầu được chèn thêm tiêu đề ESP vào giữa phần tiêu đề IP và dữ liệu được tải (Payload = TCP|UDP Header + Data)

# Giao thức ESP

- Chế độ hoạt động:
  - Tunnel:
    - » Tạo một **IP Header mới**: liệt kê các đầu cuối của ESP Tunnel (như 2 IPSec Gateway)
    - » Mã hóa và/hoặc đảm bảo toàn vẹn cho nội dung gói tin, **có cả IP Header** và một số thành phần ESP.

# Giao thức ESP

- Chế độ Tunnel:

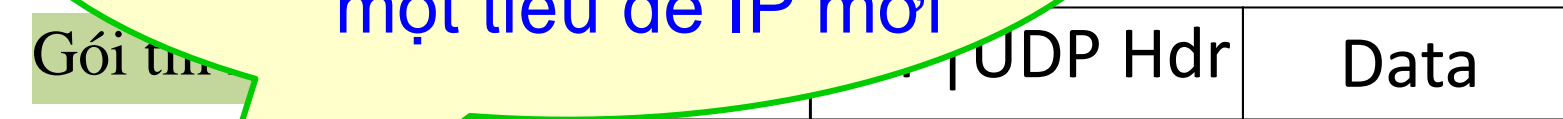


# Thực ESP

## Chế độ Tunnel:

Gói IP mới được xây dựng cùng với một tiêu đề IP mới

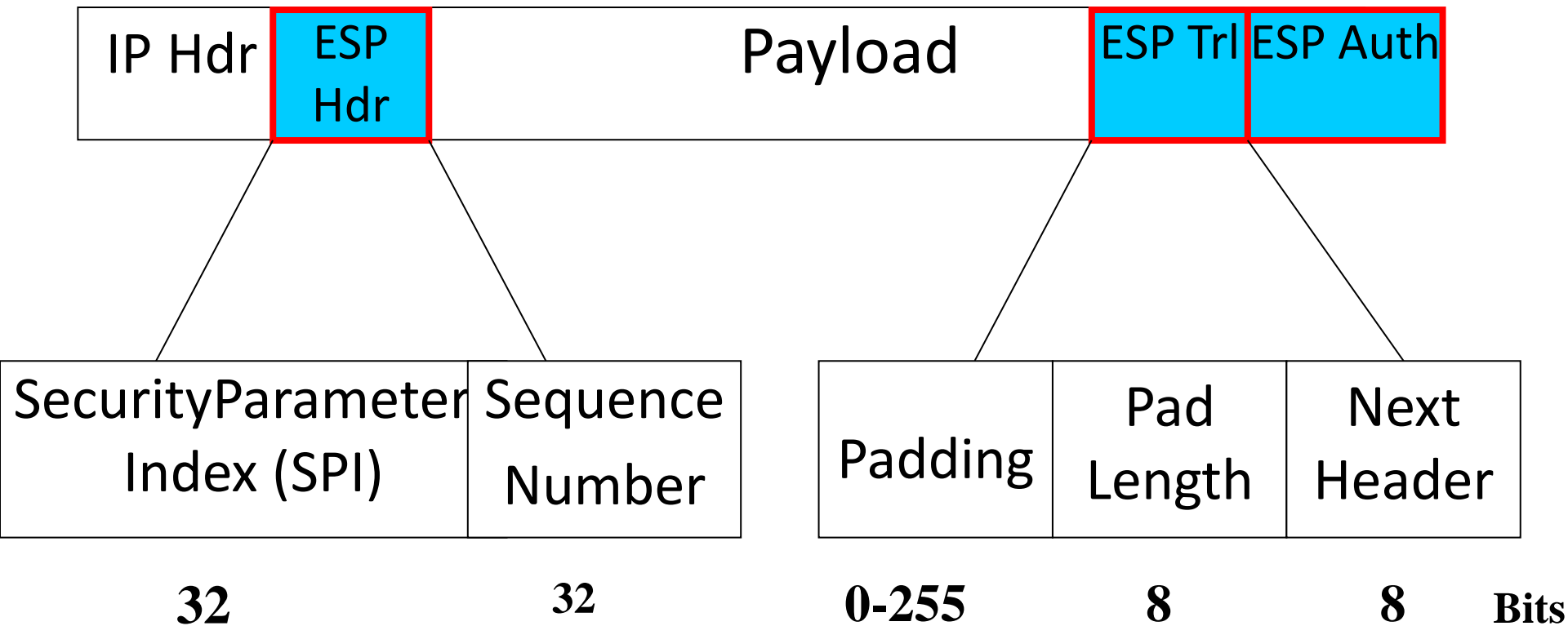
Gói tin



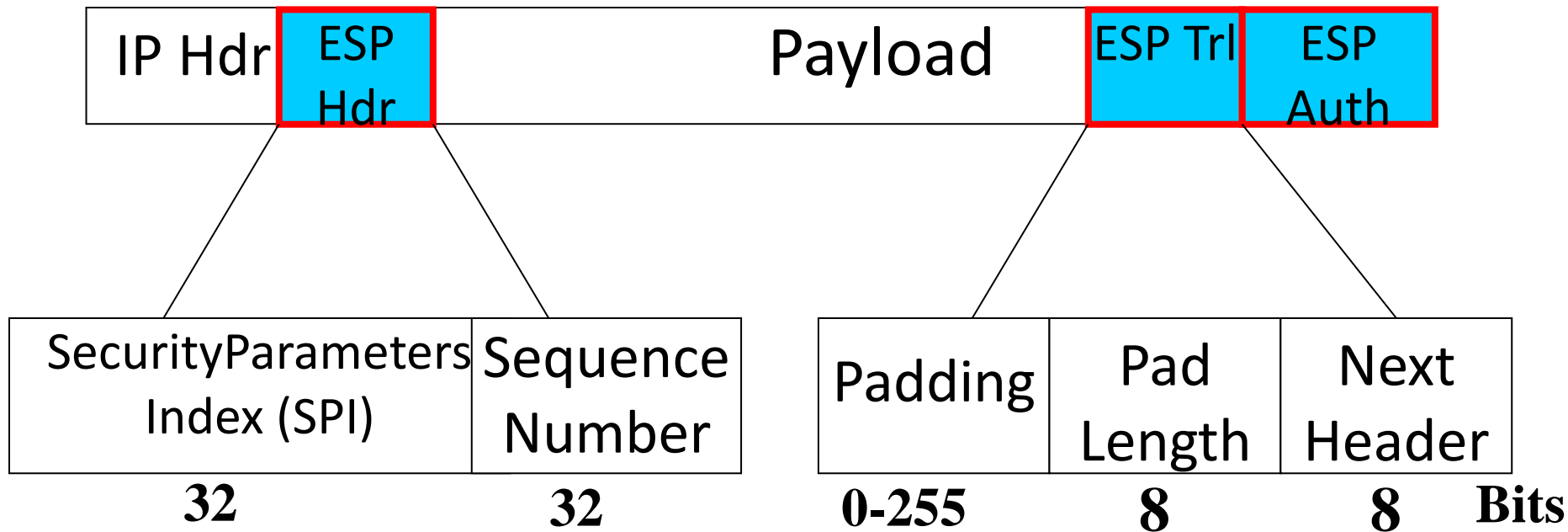
**Chế độ Tunnel:** ESP bảo vệ cả gói tin ban đầu, bao gồm cả tiêu đề IP và Payload = TCP|UDP Header + Data

# Giao thức ESP

- Khuôn dạng gói dữ liệu ESP:



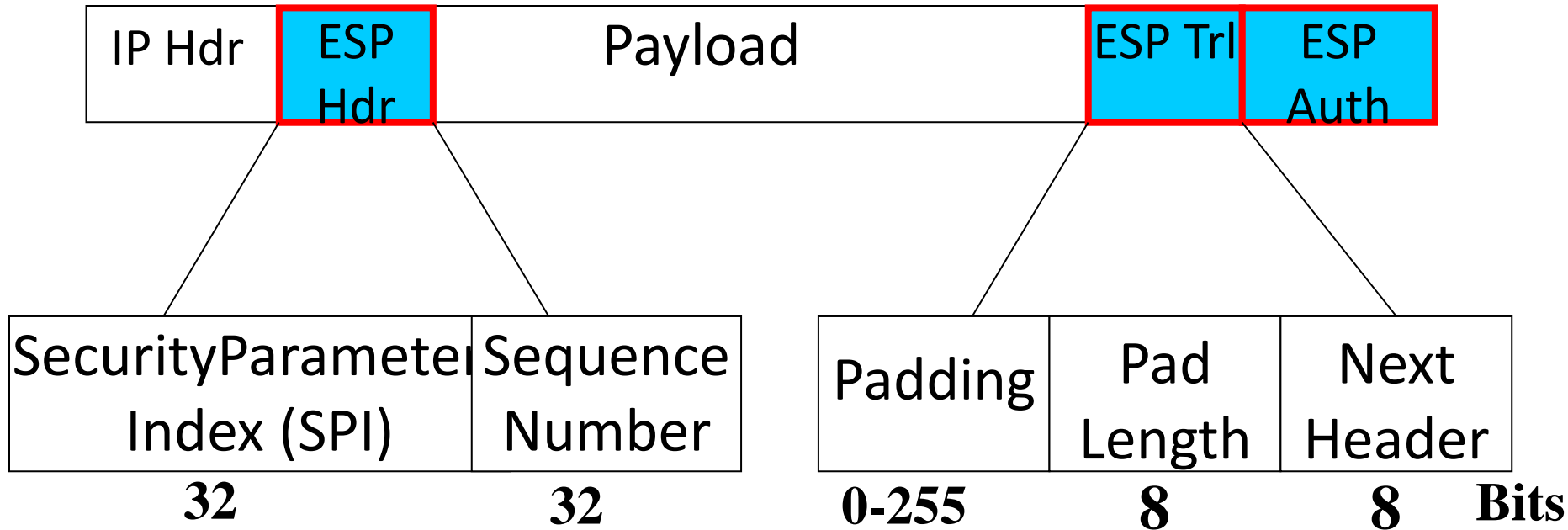
# Giao thức ESP



## SPI:

- Mỗi bên liên lạc tùy chọn gtri SPI
- Bên nhận dựa vào **SPI**, đ/c IP đích, gthức IPSec (**ESP**) => xđ một SA duy nhất để áp cho gói tin nhận được.

# Giao thức ESP

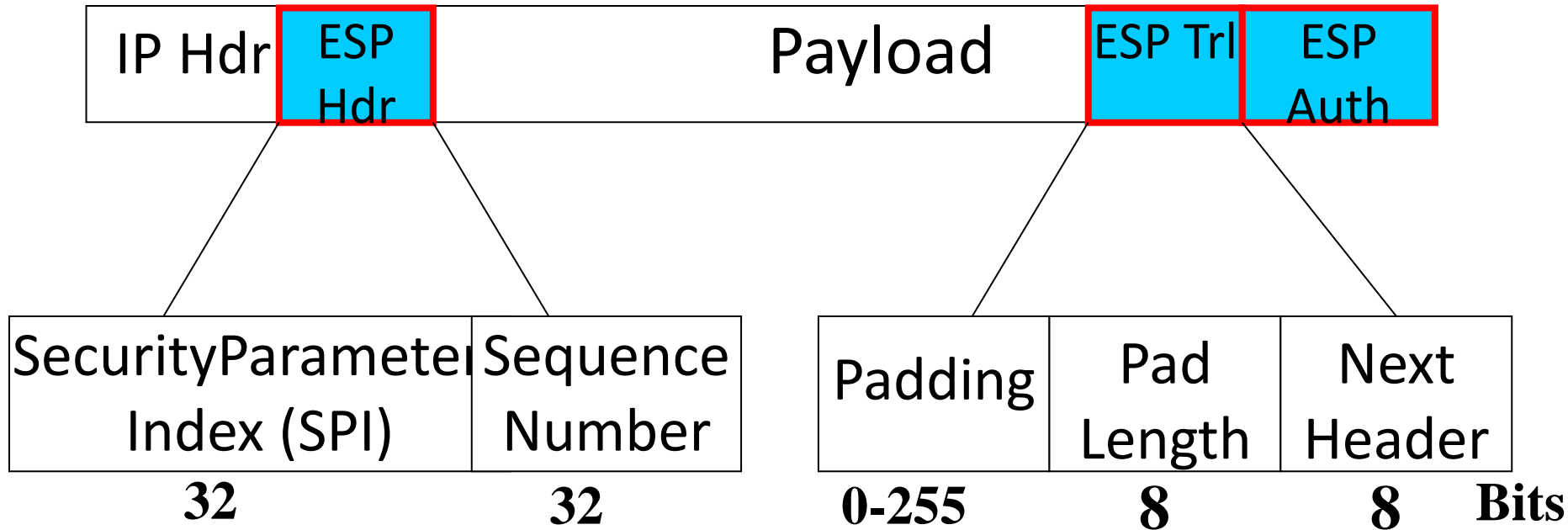


## Sequence Number:

- Khởi tạo bằng 0
- Tăng lên 1 nếu mỗi gói tin được gửi
- Để chống trùng lặp gói tin



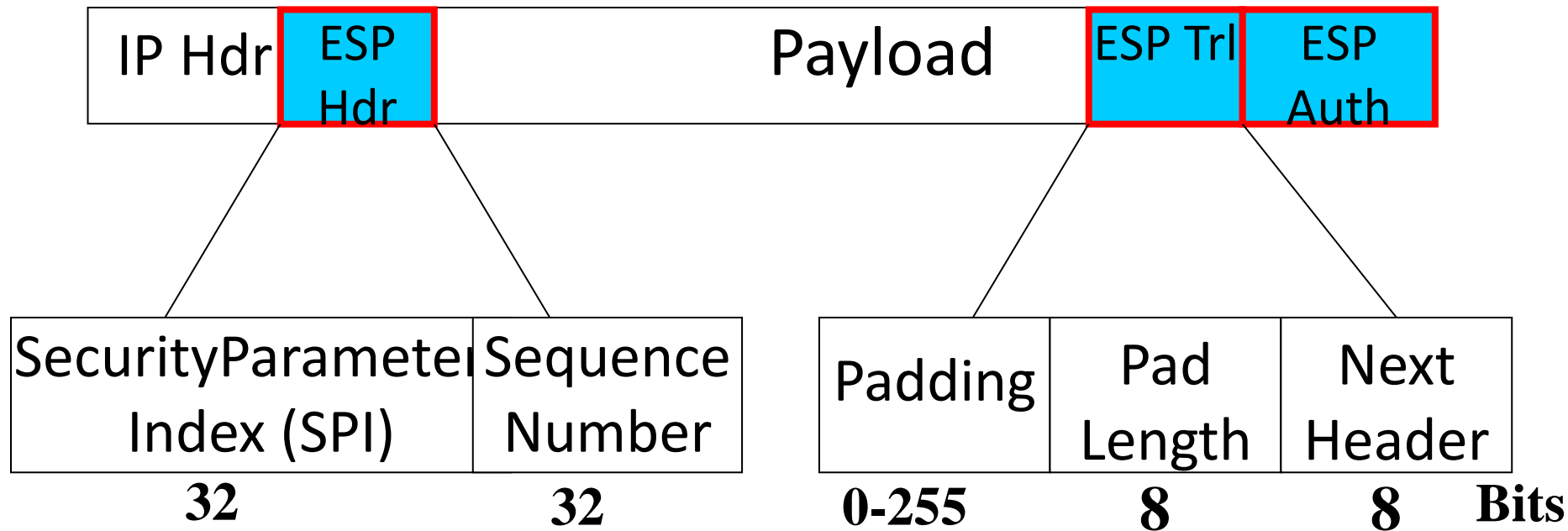
# Giao thức ESP



## Payload:

- Là phần payload data được mã hóa

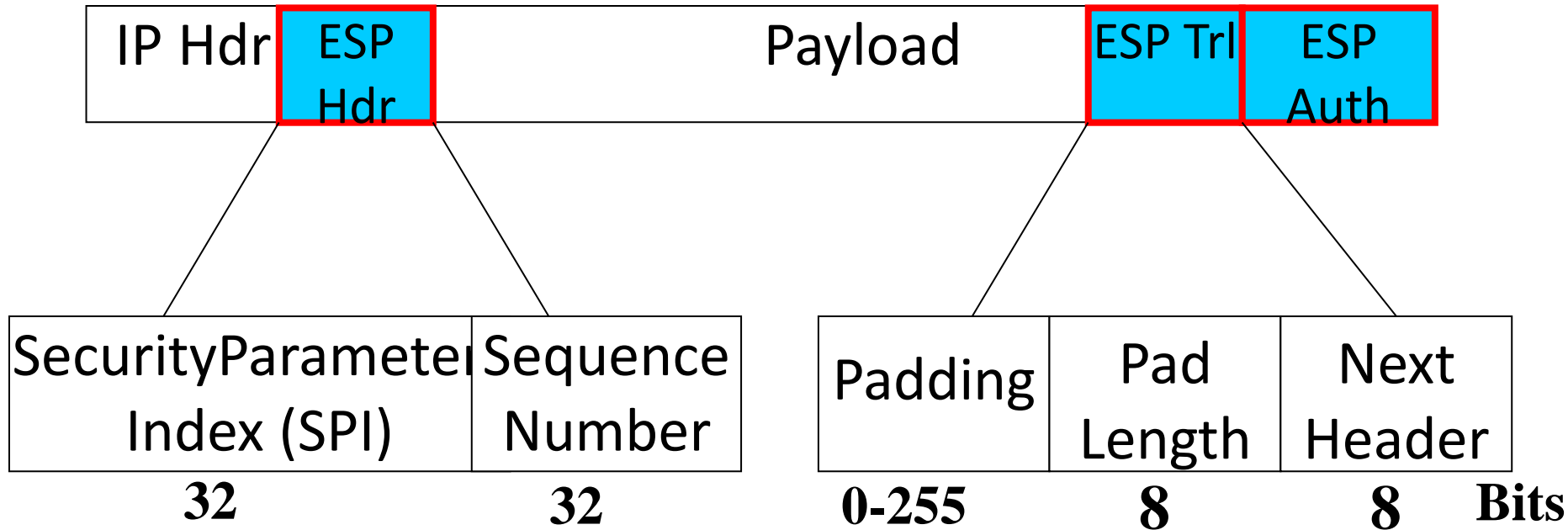
# Giao thức ESP



## Padding (0-255 bytes):

- Là phần dữ liệu được thêm vào gói tin (trước khi mã hóa) để đoạn dữ liệu được mã hóa là một số nguyên lần của một khối các byte
- Nó cũng được dùng để che dấu độ dài thực của Payload

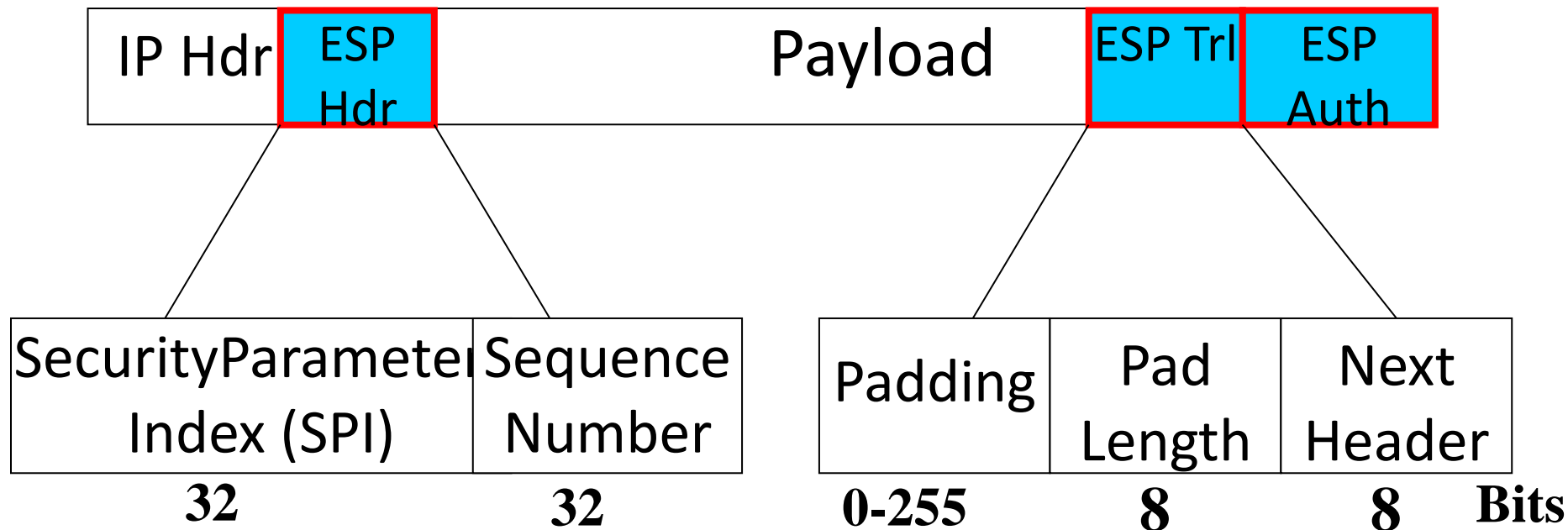
# Giao thức ESP



## Pad Length:

-Trường này xác định số byte padding đã thêm vào

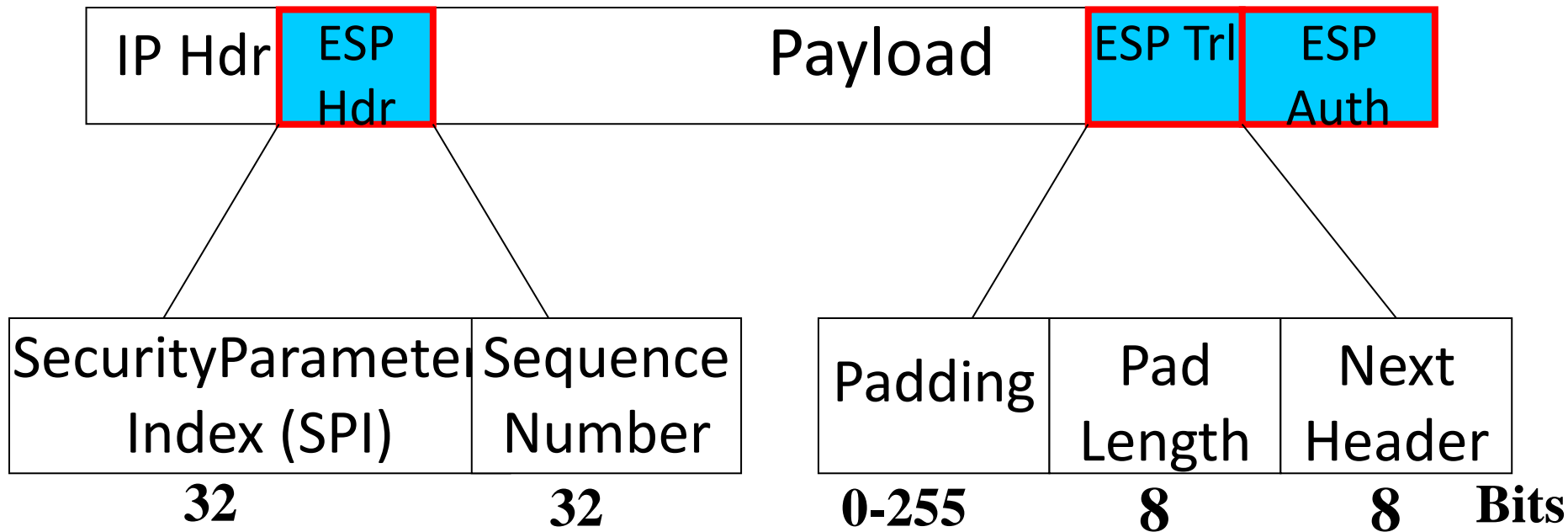
# Giao thức ESP



## Next Header:

- Trong **Tunnel Mode**, Payload là gói tin IP, thì **Next Header = 4** (IP –in-IP)

# Giao thức ESP



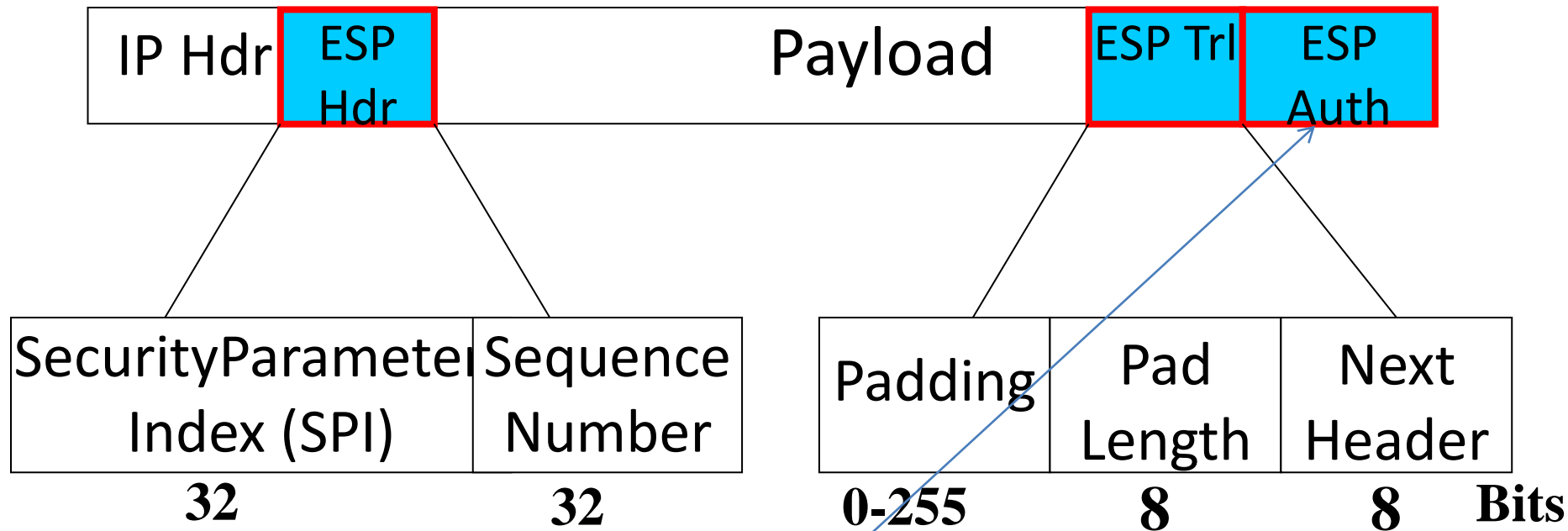
## Next Header:

-Trong **Transport Mode**, Payload là giao thức tầng 4 Transport.

+ Nếu là **TCP** thì **Next Header = 6**

+ Nếu là **UDP** thì **Next Header = 17**

# Giao thức ESP



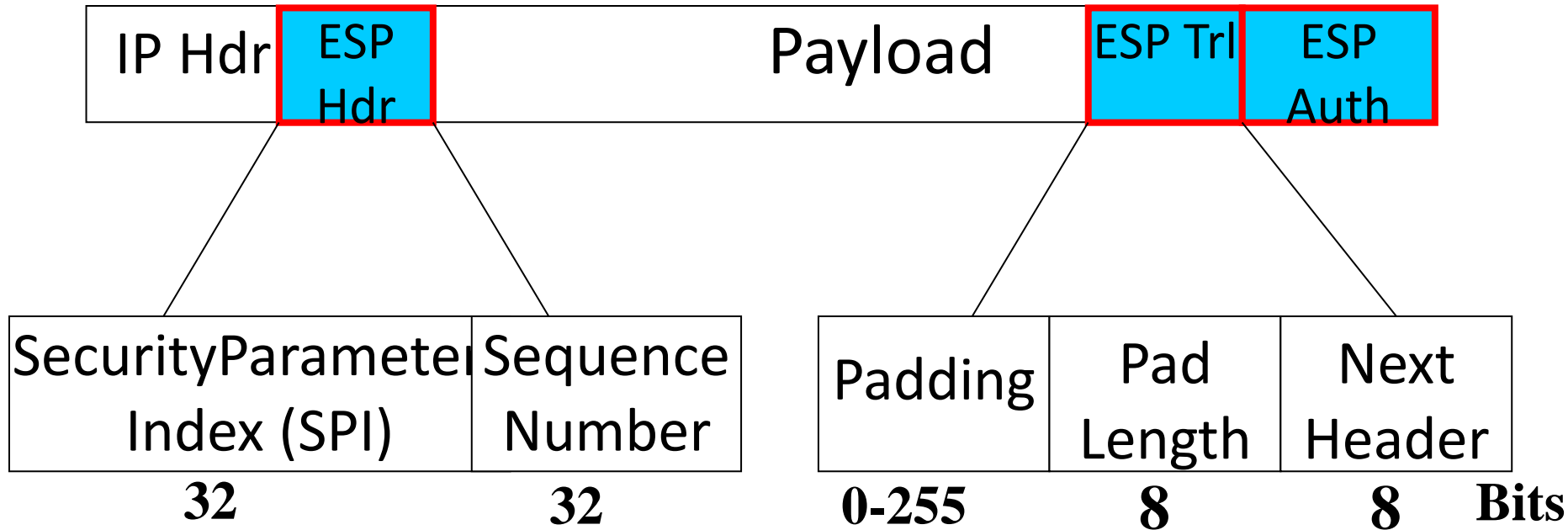
## Authentication Data:

- Chứa giá trị **ICV** (Integrity Check Value)

$$\text{ICV} = \text{HMAC}(\text{ESP Hdr} + \text{Payload} + \text{ESPTrl} + \text{Key})$$

- ICV phải là bội của 32 bit

# Giao thức ESP



**Lưu ý:** trong AH, xác thực được cả phần IP Header, trong ESP thì không

# Giao thức ESP

- **Các thuật toán sử dụng:**
  - Thuật toán mã hóa:
    - AES-CBC, AES-CTR, 3DES
  - Thuật toán xác thực:
    - MD5, SHA1



# Giao thức ESP

---

## Xử lý gói tin ESP đầu vào & đầu ra

*(SV tìm hiểu thêm trong Giáo trình “Các giao thức bảo mật mạng riêng ảo”, HVKTMM, năm 2013)*

# Giao thức ESP

- **Xử lý gói tin đầu ra:**
  - Tìm kiếm SA
  - Mã hoá gói tin:
  - Tạo Sequence Number
  - Tính ICV
  - Phân mảnh
- **Xử lý gói tin đầu vào:**
  - Ghép mảnh
  - Tìm kiếm SA
  - Kiểm tra SN
  - Kiểm tra ICV
  - Giải mã gói tin

# Giao thức ESP

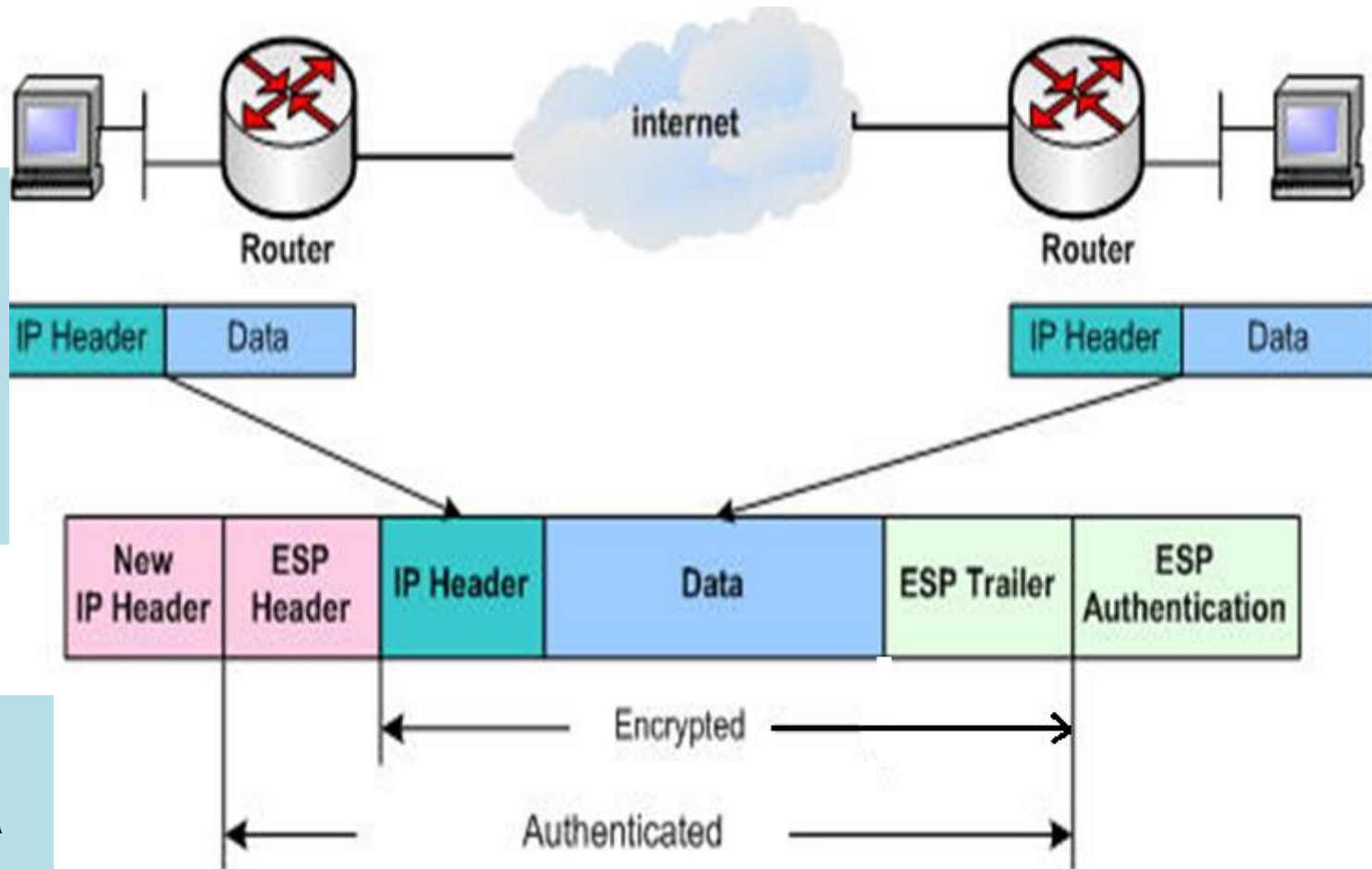
---

ESP: mã hóa dữ liệu

# Giao thức ESP

ESP sử dụng mã hóa đối xứng để mã hóa các gói tin

ESP sử dụng thuật toán mã hóa: AES-CBC, AES-CTR, 3DES...



Mã hóa ở chế độ Tunnel

# Giao thức ESP

---

## Phân tích gói tin ESP

# Giao thức ESP

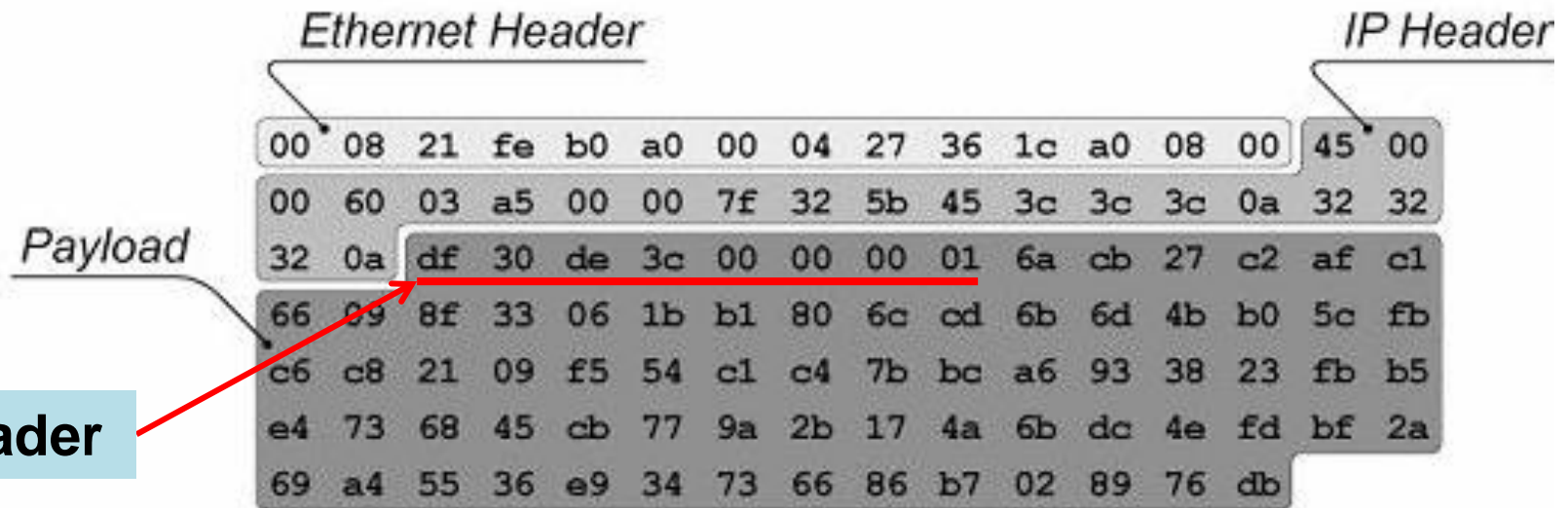


Figure 3-11: ESP Packet Capture

	SPI	Sequence Number
A→B	df 30 de 3c	00 00 00 01
B→A	d9 64 ce 53	00 00 00 01
A→B	df 30 de 3c	00 00 00 02
B→A	d9 64 ce 53	00 00 00 02

Figure 3-12: ESP Header Fields from Sample Packets

# Giao thức ESP

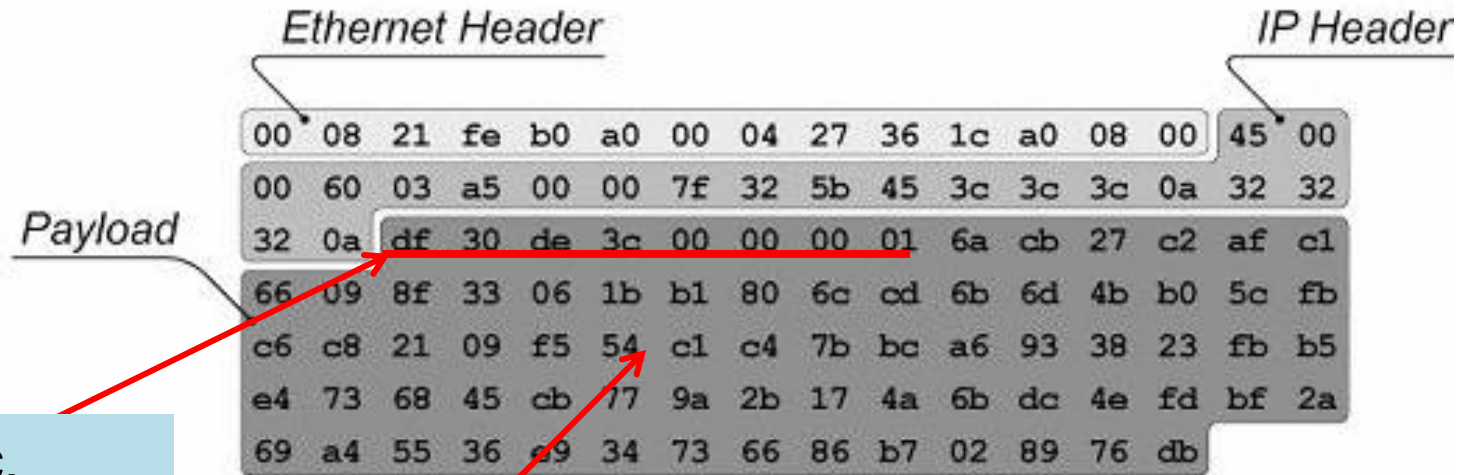


Figure 3-11: ESP Packet Capture

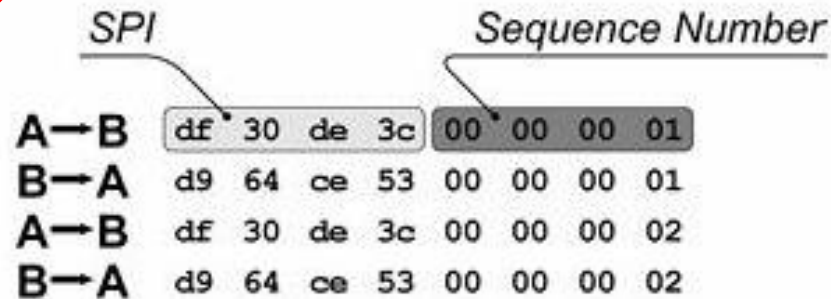


Figure 3-12: ESP Header Fields from Sample Packets

**SPI=df30de3c,  
Sequence Number  
=00000001**

Phần dữ liệu không  
thể xác định được,  
vì được mã hóa

# Giao thức ESP

- ESP là giao thức đảm bảo tính toàn vẹn, xác thực và bí mật, chống replay gói tin cũ.
- Hoạt động trong hai chế độ: Transport và Tunnel
- ESP version 1 chỉ mã hóa cho phần Payload data.
- ESP version 2: đảm bảo cả toàn vẹn và xác thực
- ESP version 3: hỗ trợ thêm thuật toán AES Counter mode (AES-CTR).
- ESP Tunnel thường sử dụng phổ biến trong IPSec vì nó mã hóa IP Header gốc, có thể che giấu đ/c source, dest thật của gói tin



# Bảng so sánh giữa giao thức AH và ESP

Security	AH	ESP
Layer-3 IP protocol number	51	50
Provides for data integrity	yes	Yes
Provides for data authentication	Yes	yes
Provides for data encryption	No	Yes
Protects against data replay attacks	yes	yes
Works with NAT	No	yes
Works with PAT	No	No
Protects the IP packet	yes	No
Protects only the data	No	yes

1

Giao thức ESP

2

Kết hợp AH & ESP

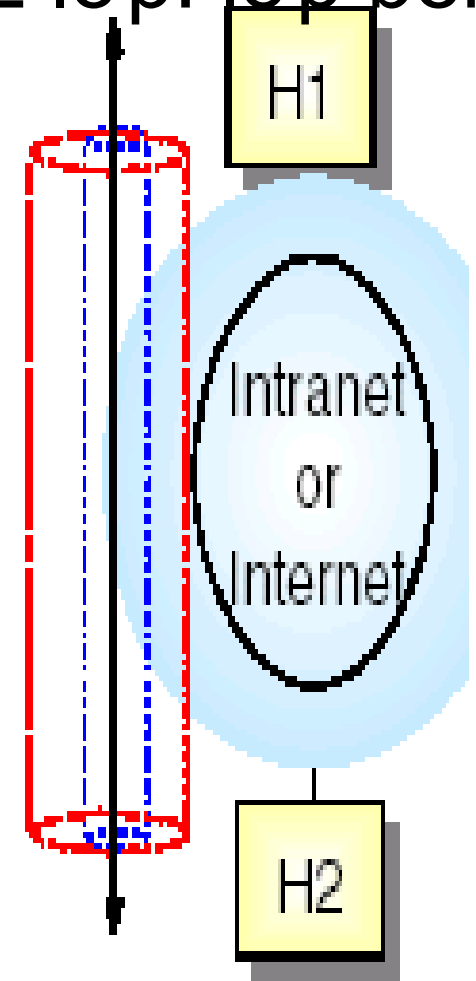
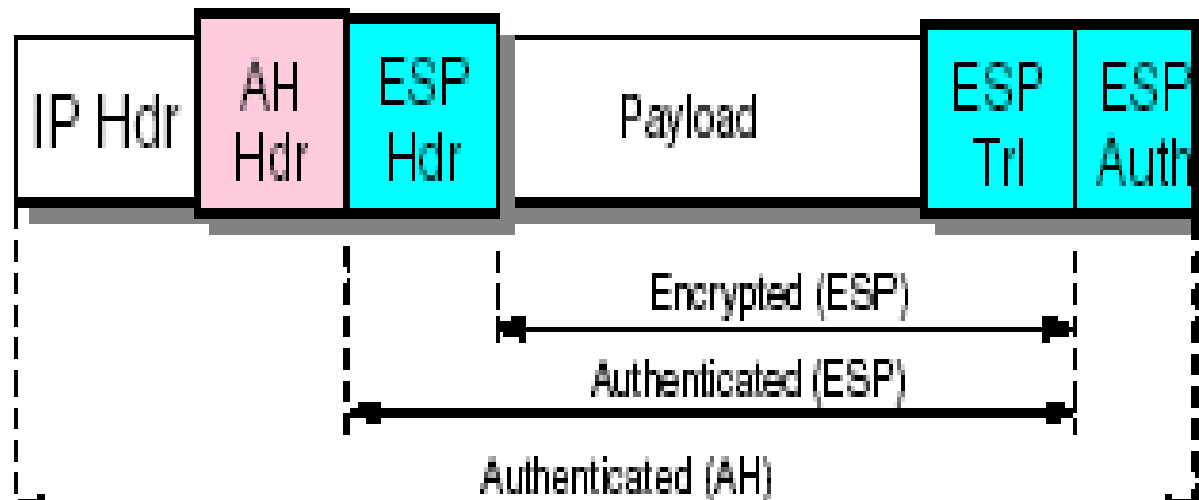
3

Giao thức IKE

# Kết hợp AH và ESP trong chế độ Transport

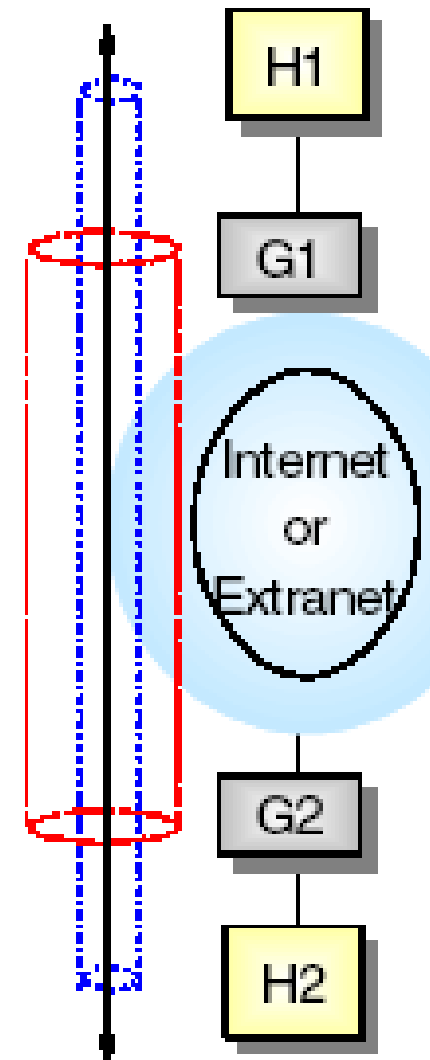
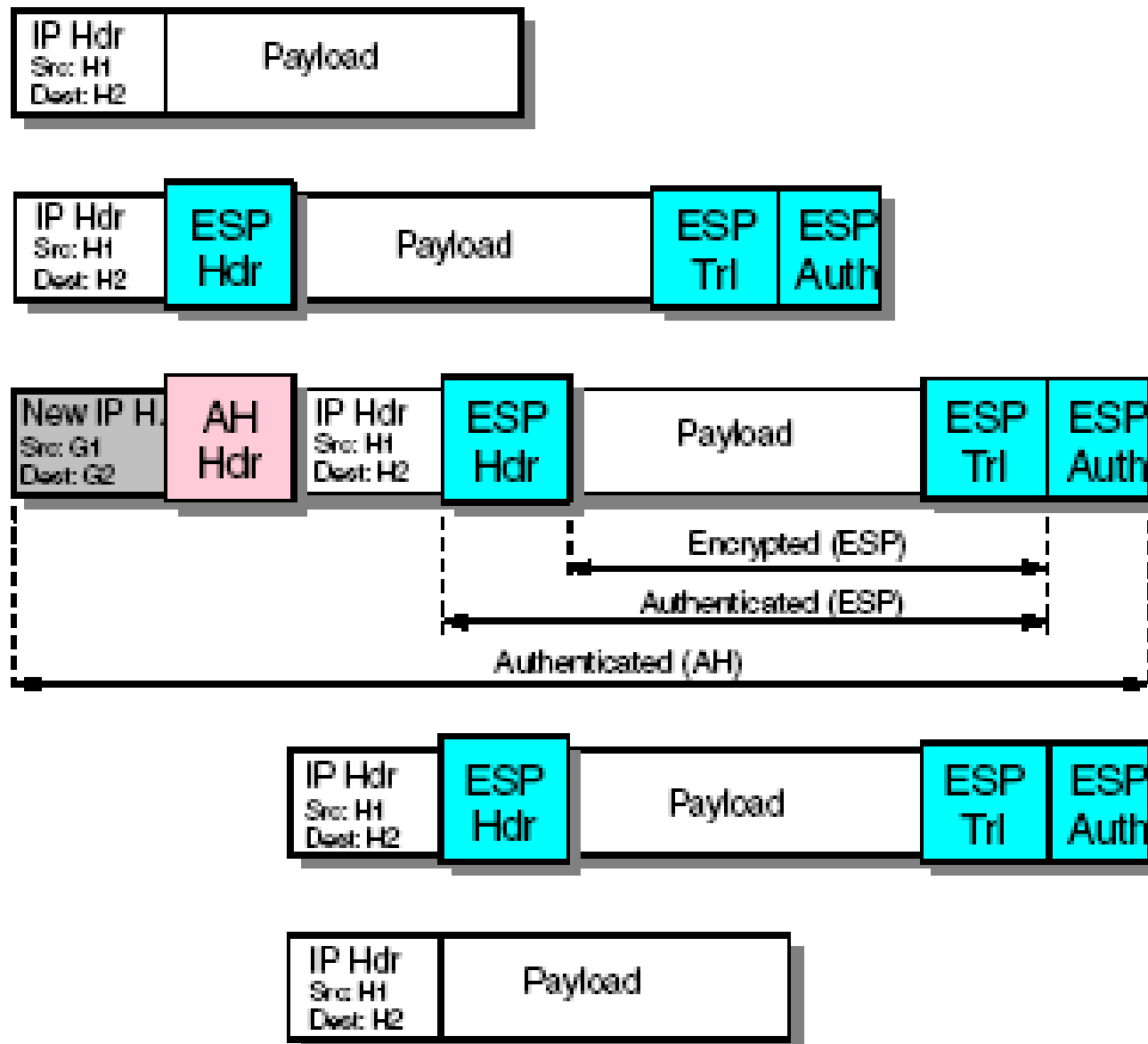
- Gói tin sẽ được đảm bảo an toàn 2 lớp: lớp bên ngoài là AH, lớp bên trong là ESP.



## IPSec Transport Mode



 AH in transport mode  
 ESP in transport mode

# Kết hợp AH và ESP trong chế độ Tunnel



 AH in tunnel mode  
 ESP in transport mode

1

Giao thức ESP

2

Kết hợp AH & ESP

3

Giao thức IKE

# Giao thức trao đổi khoá Internet (IKE)

- Bản thân IPSec không có khả năng thiết lập SA
- Do đó quá trình được chia làm 2 phần:
  - Giao thức IKE tạo, thoả thuận các SA
  - IPSec xử lý ở mức gói

# Giao thức trao đổi khoá Internet (IKE)

- Giao thức **IKE** (**I**nternet **K**ey **E**xchange – RFC 2409):
  - Là giao thức để quản lý, trao đổi khóa trong IPSec
  - Cho phép **thương lượng** và **tạo tự động** các **IPSec SA** giữa các bên liên lạc IPSec.
  - IKE cũng chịu trách nhiệm **xoá** các khóa, SA sau khi một phiên truyền tin kết thúc

# Lịch sử của IKE

- IKE được đưa ra đầu tiên vào năm 1998 bởi IETF
- Được xây dựng dựa trên nền tảng của ba giao thức:
  - Giao thức phân phối khóa Oakley (Key Distribution) – RFC 2412
  - Giao thức quản lý khóa ISAKMP (Key Management) – RFC 2408
  - SKEME (secure key exchange mechanism for Internet)
- IKE Có thể được sử dụng bên ngoài IPSec
- IKE hiện đã được phát triển với 2 phiên bản, phiên bản IKEv1 và IKEv2.



# Giao thức trao đổi khoá Internet (IKE)

- Thuật toán trao đổi khóa Diffie-Hellman

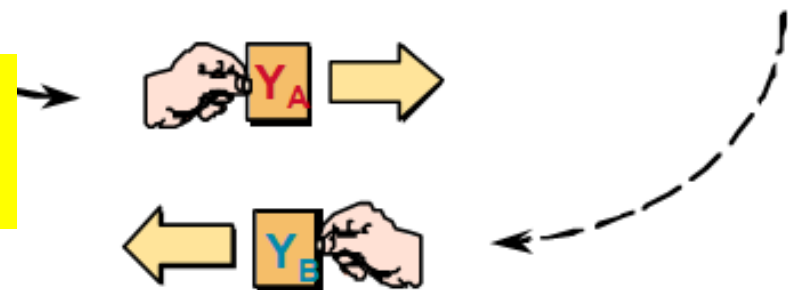
**Alice** Private Value,  $X_A$   
Public Value,  $Y_A$

$$Y_A = g^{X_A} \mod p$$

Private Value,  $X_B$  **Bob**  
Public Value,  $Y_B$

$$Y_B = g^{X_B} \mod p$$

$p$  – là số nguyên tố rất lớn  
 $g$  – là phần tử sinh của  $Z_p^*$



$$Y_B^{X_A} \mod p = g^{X_A X_B} \mod p = Y_A^{X_B} \mod p$$

(shared secret)

# Mối quan hệ giữa IPSec và IKE

- IPSec cần các SA để bảo vệ lưu lượng
- Nếu chưa có các SA, IPSec sẽ yêu cầu IKE cung cấp các **IPSec SA**.
- IKE mở một phiên quản lý với các bên tham gia, và thương lượng tất cả **các SA** và các **khóa** cho IPSec.
- IPSec bắt đầu thực hiện bảo vệ lưu lượng.

# Mối quan hệ giữa IPSec và IKE

1. Outbound packet from Alice to Bob. No SA.



Alice's Laptop



4. Packet is sent from Alice to Bob protected by IPsec SA.



Bob's Laptop

2. Alice's IKE begins negotiation with Bob's.

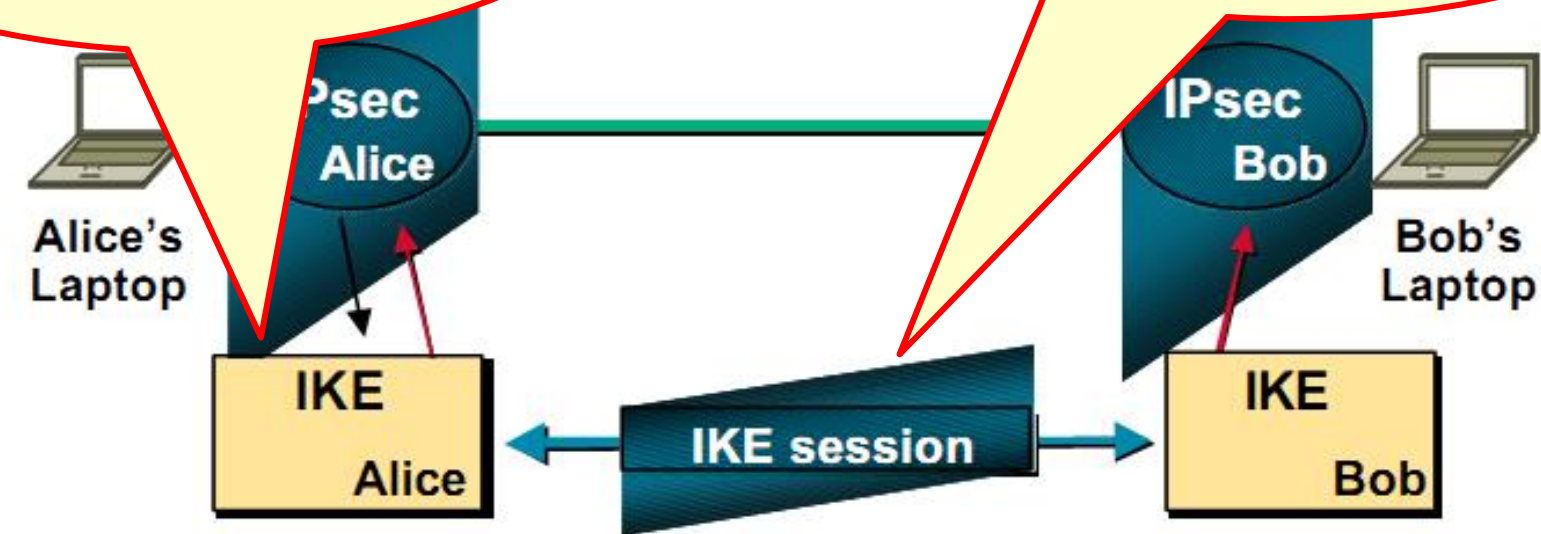
3. Negotiation complete. Alice and Bob now have complete SAs in place.

IKE session

# hệ giữa IP

Một phiên IKE chạy trên giao thức UDP với các cổng nguồn và đích được thiết lập = 500.

Kết quả của phiên IKE = các **IKE SA** (để bảo vệ phiên IKE hoạt động an toàn)



2. Alice's IKE begins negotiation with Bob.

3. Negotiation complete. Alice and Bob now have complete SAs in place.

Sau đó, IKE thiết lập tất cả các **IPSec SA** được yêu cầu.

# Giao thức trao đổi khoá Internet (IKE)

---

## 2 pha của IKE

# Giao thức trao đổi khoá Internet (IKE)

- **IKE hoạt động trên 2 pha:**

- **Pha 1:**

- + Mục tiêu: Thương lượng các tham số mật mã, chia sẻ khóa bí mật (Diffie-Hellman), xác thực các bên tham gia.

- => Thu được các IKE SA, các khóa mật

- **Pha 2:**

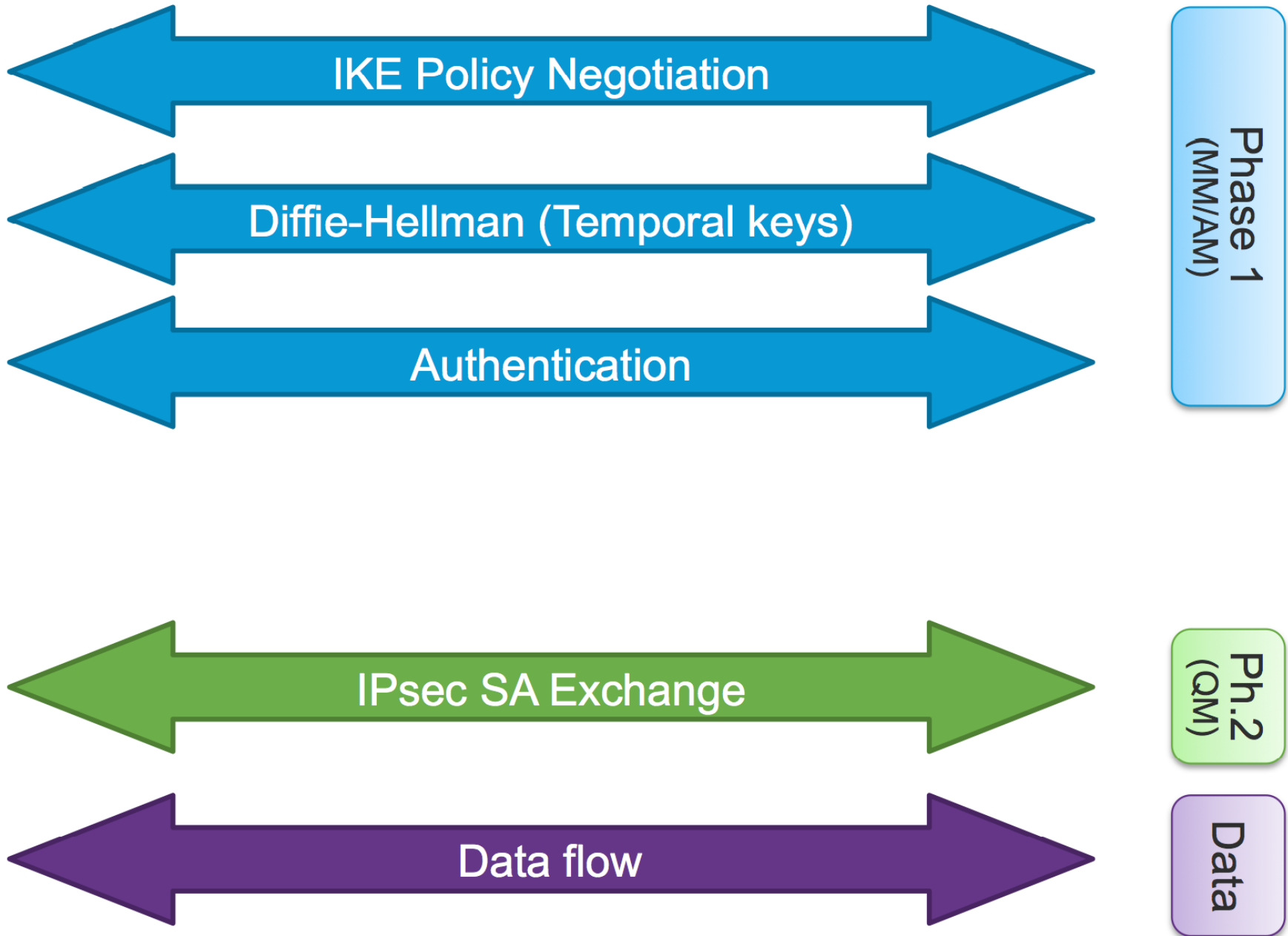
- + Mục tiêu: thỏa thuận được các **khóa** mật mã sử dụng để bảo vệ đường truyền cho các thực thể, và các **SA (IPSec SA)** cho trao đổi dữ liệu.

- => Thu được các IPSec SA, các khóa mật

# Giao thức trao đổi khoá Internet (IKE)

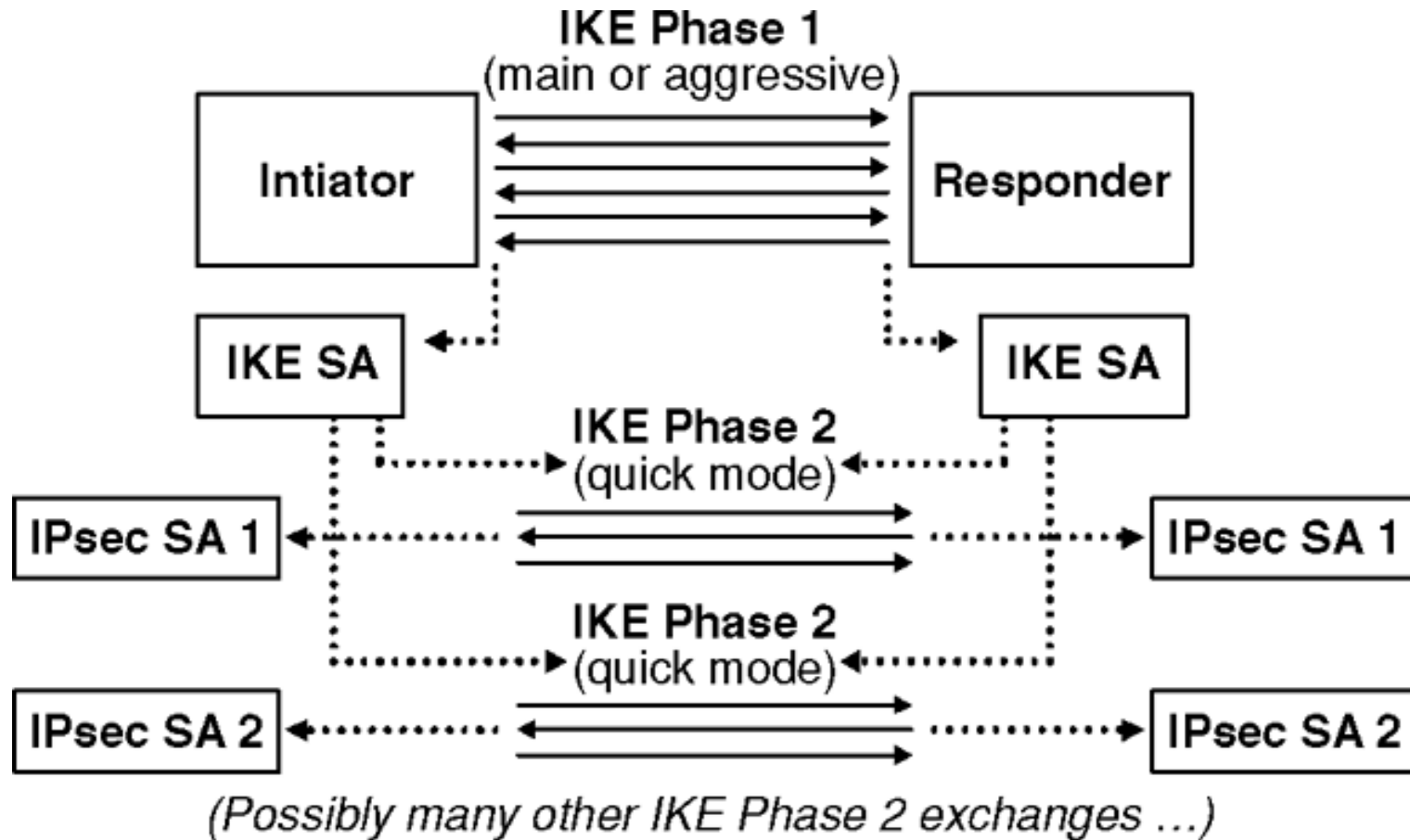
- **IKE gồm 2 pha:**
  - **IKE pha 1:**
    - Sử dụng *Main mode* hoặc *aggressive mode*.
    - Thương lượng IKE SA
      - » Để bảo vệ cho IKE pha thứ 2
  - **IKE pha 2:**
    - Sử dụng *Quick mode*
      - » Thương lượng các IPSec SAs

# Giao thức trao đổi khoá Internet (IKE)





# Giao thức trao đổi khoá Internet (IKE)



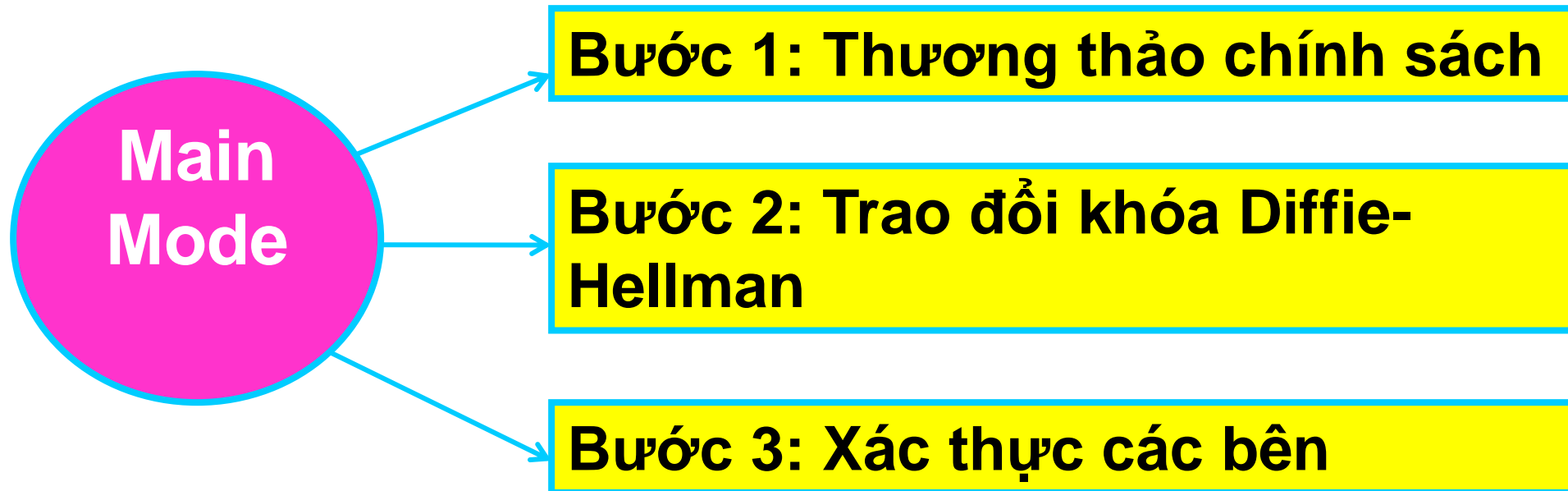
# Giao thức trao đổi khoá Internet (IKE)

---

## IKE pha 1

# Giao thức trao đổi khoá Internet (IKE)

- IKE pha 1 - Main mode:



# Giao thức trao đổi khoá Internet (IKE)

- IKE pha 1 - Main mode:

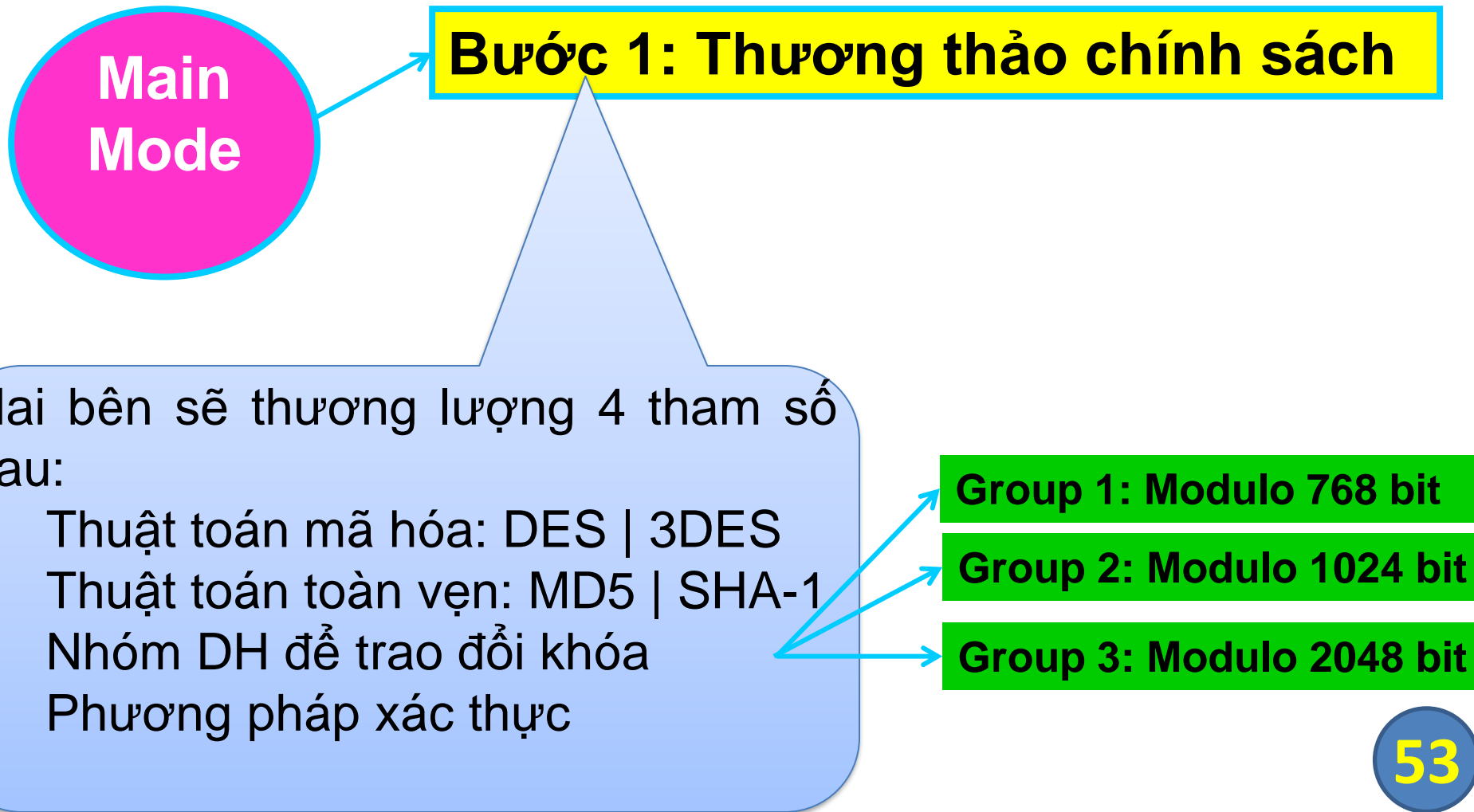
**Main  
Mode**

## **Bước 1: Thương thảo chính sách**

- Một bên sẽ đưa ra một danh sách các thuật toán.
- Bên nhận sẽ lựa chọn hoặc có yêu cầu khác.

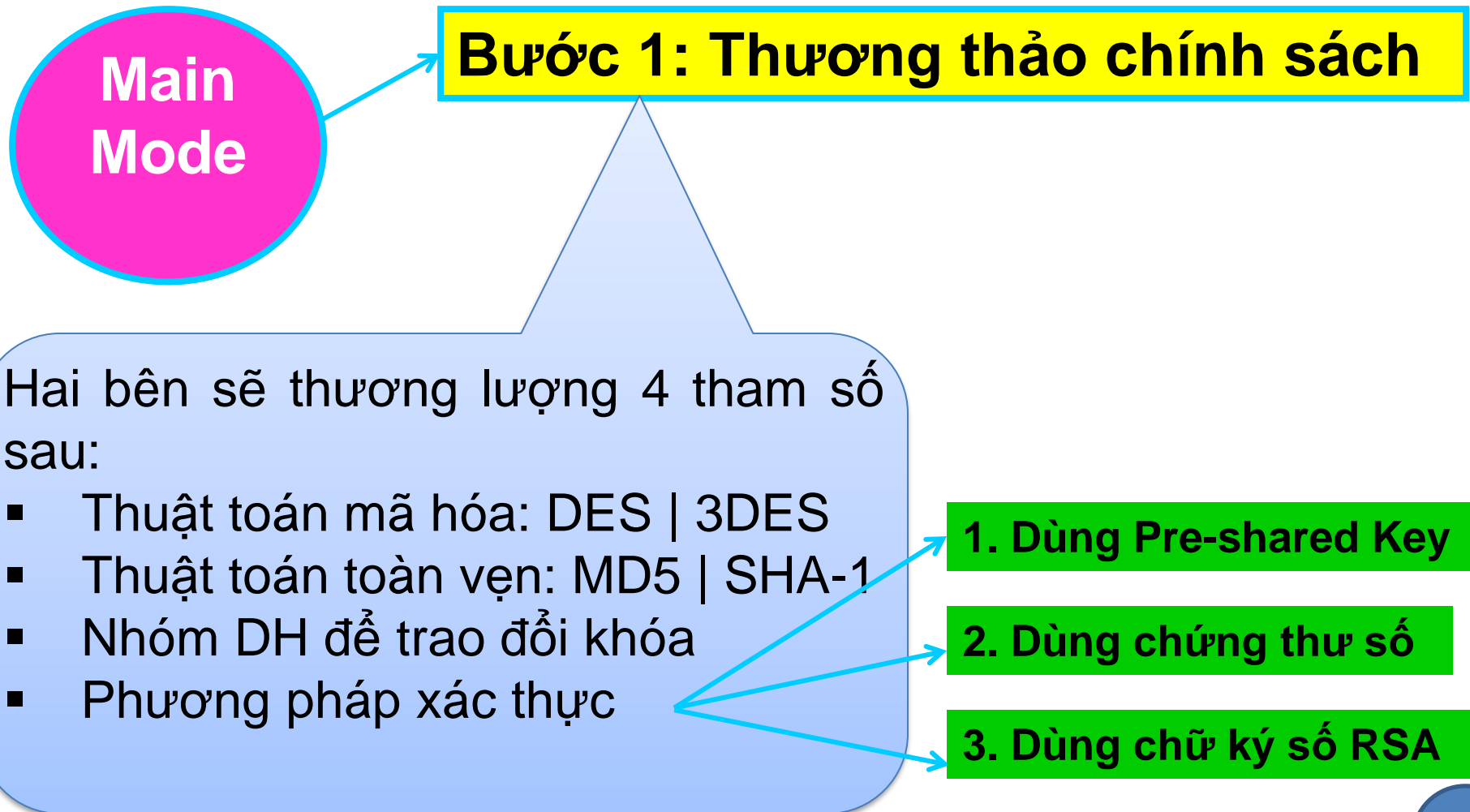
# Giao thức trao đổi khoá Internet (IKE)

- IKE pha 1 - Main mode:



# Giao thức trao đổi khoá Internet (IKE)

- IKE pha 1 - Main mode:



# Giao thức trao đổi khoá Internet (IKE)

- IKE pha 1 - Main mode:

**Main  
Mode**

**Bước 2: Trao đổi khóa Diffie-Hellman**

- Hai bên thực hiện trao đổi khóa bằng thuật toán Diffie-Hellman (với Group chọn ở Bước 1).
- Kết quả, hai bên có cùng một khóa chủ  $K_M$

# Giao thức trao đổi khoá Internet (IKE)

- IKE pha 1 - Main mode:

**Main  
Mode**

**Bước 3: Xác thực các bên**

Kết hợp:

- Kết quả bước 1: Thuật toán mã hóa, Thuật toán băm + Phương pháp xác thực
  - Kết quả bước 2: Khóa  $K_M$
- => **Identity payload** được băm sau đó được mã hóa bằng  $K_M$

**Identity payload =**  
**Identity type** + port  
+ protocol



# Giao thức trao đổi khoá Internet (IKE)

- Một số kiểu ID IPSec

- VPN Gateway IDs are exchanged in IKE Main Mode:

IPsec ID type	FreeS/WAN ipsec.conf example
ID_IPV4_ADDR	<code>rightid=11.22.33.44</code>
ID_FQDN	<code>rightid=@gateway.kool.net</code>
ID_USER_FQDN	<code>rightid=antje@kool.net</code>
ID_DER_ASN1_DN	<code>rightid="C=DE, O=Kool AG, CN=antje@kool.net"</code>
ID_KEY_ID	<code>rightid=@#736f6e696377616c6c #sonicwall</code>

- Client / Client Subnet IDs are exchanged in IKE Quick Mode:

ID_IPV4_ADDR	<code>right=11.22.33.44</code>
ID_IPV4_ADDR_SUBNET	<code>rightsubnet=10.1.0.0/22</code>

# Giao thức trao đổi khoá Internet (IKE)

- IKE pha 1 - Main mode:

**Main  
Mode**



```
graph LR; MM((Main Mode)) --> B1[Bước 1: Thương thảo chính sách]; MM --> B2[Bước 2: Trao đổi khóa Diffie-Hellman]; MM --> B3[Bước 3: Xác thực các bên];
```

**Bước 1: Thương thảo chính sách**

**Bước 2: Trao đổi khóa Diffie-Hellman**

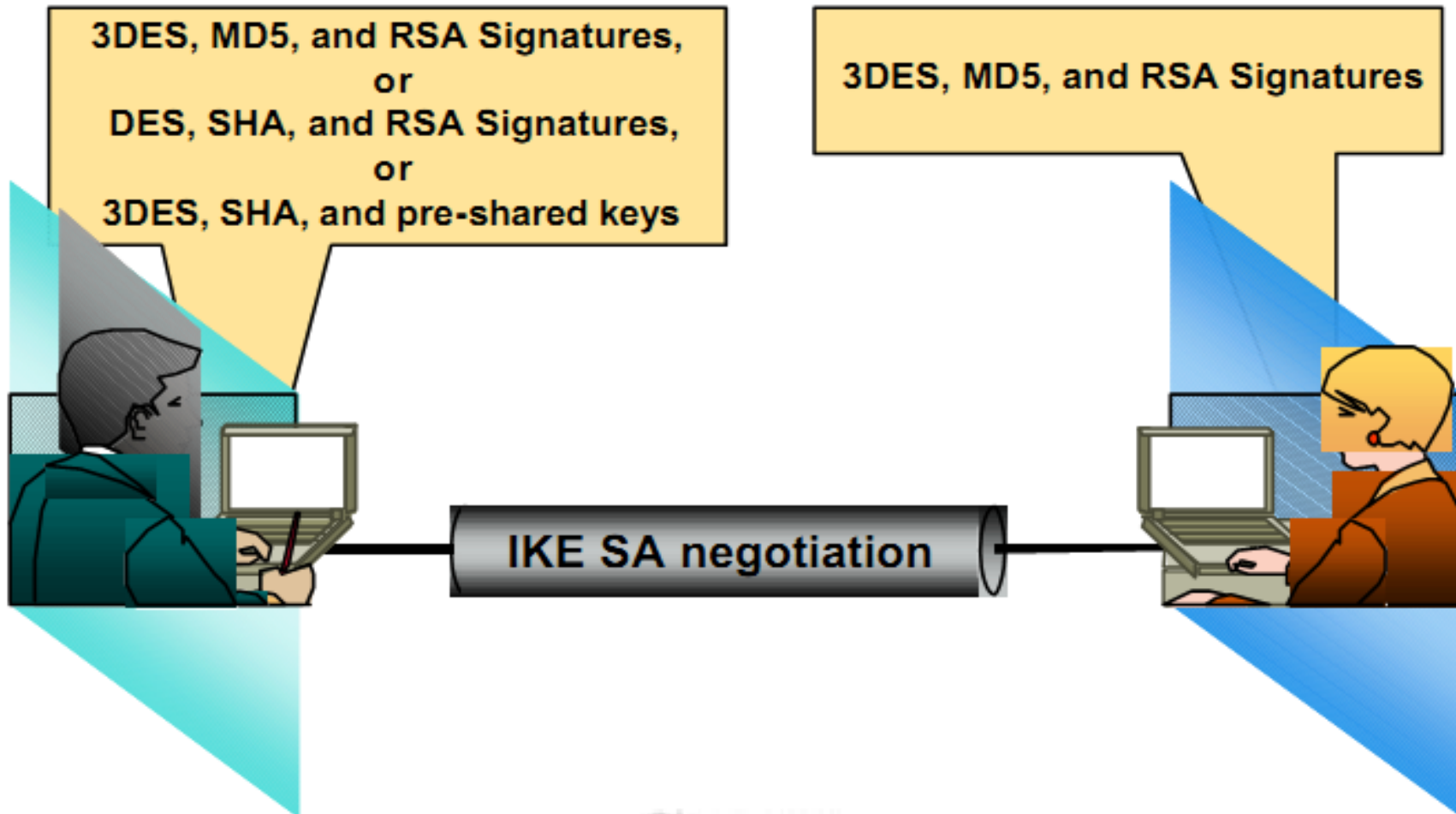
**Bước 3: Xác thực các bên**

## Kết quả pha 1:

- Hai bên thương lượng được IKE SA (thuật toán mã hóa, thuật toán xác thực, phương pháp xác thực)
- Khóa mật  $K_M$
- Xác thực được nhau.

# Giao thức trao đổi khoá Internet (IKE)

- IKE pha 1 - Ví dụ:**



# IKE pha 1 - Main mode

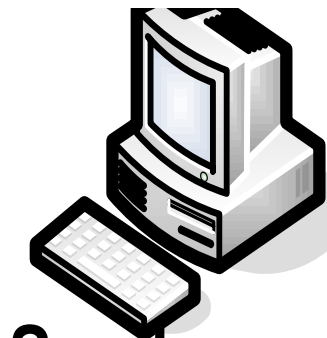
- Xác minh và bảo vệ định danh của các bên liên lạc
- Trong chế độ này, 6 thông điệp được trao đổi
  - 2 message đầu: thương lượng tham số mật mã
  - 2 message giữa: trao đổi khoá DH và Nonce
  - 2 message cuối: xác thực các bên

# IKE pha 1 - Main mode

- Mô hình

## Tiêu đề IKE:

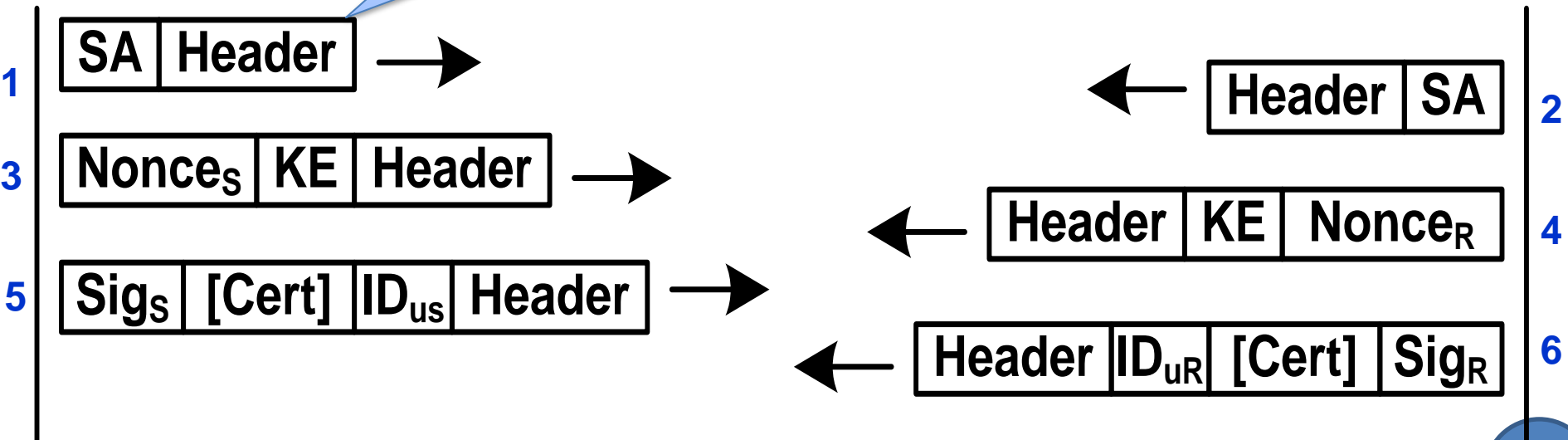
- Có thể là số 0
- Hay là một message digest



Sender



Recipient



# IKE pha 1 - Main mode

- Mô hình

SA – một ds thuộc tính an toàn:

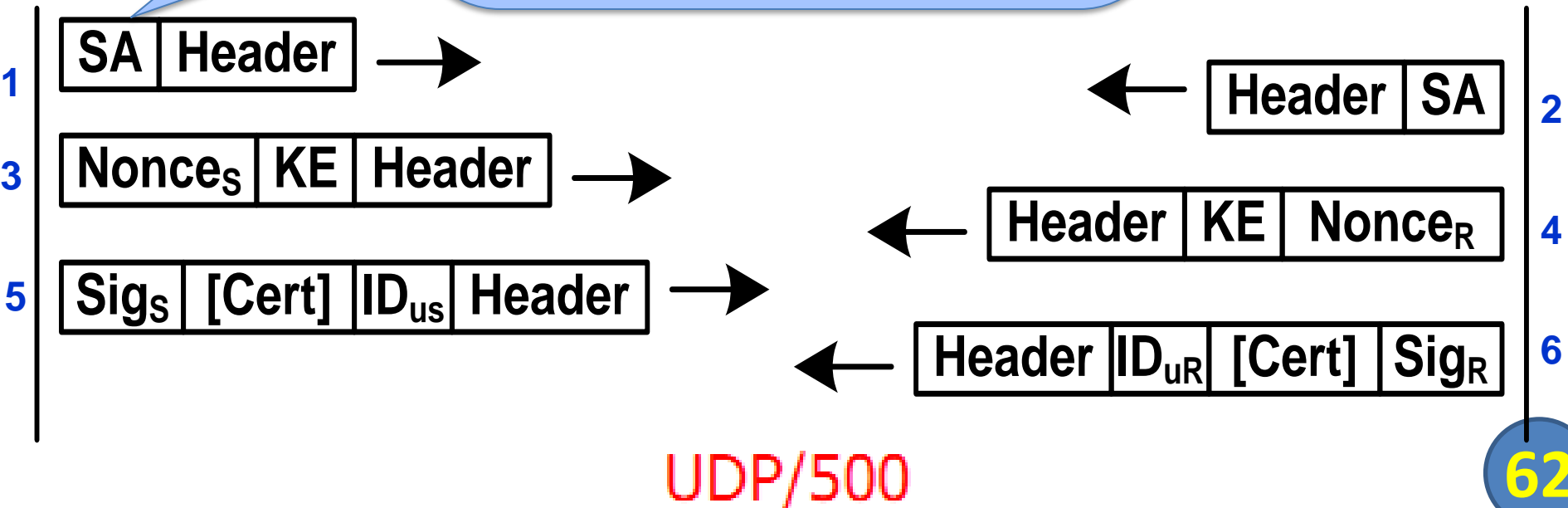
- HMAC?
- DH Group?
- Encryption algorithm, Key length?
- Authen method?...



Sender



Recipient

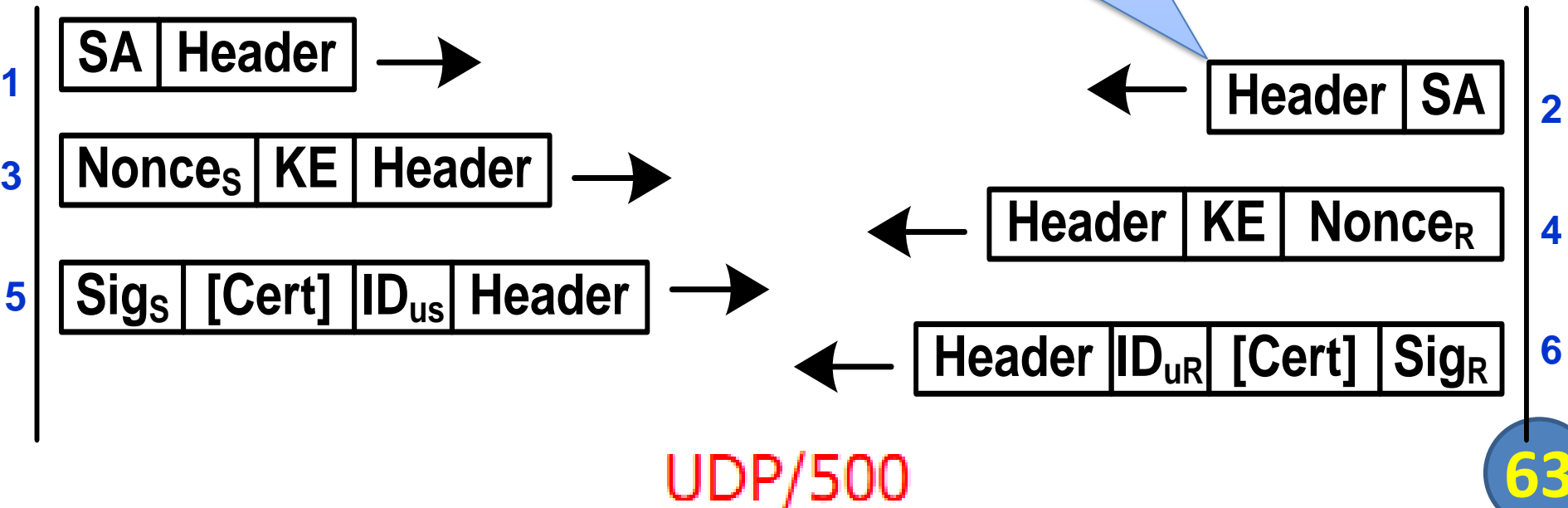
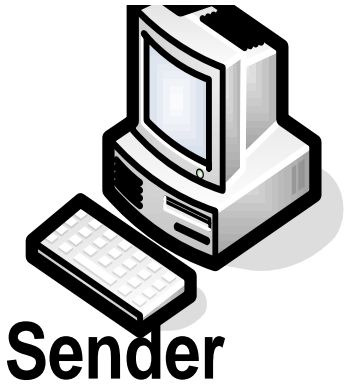


# IKE pha 1 - Main mode

- Mô hình

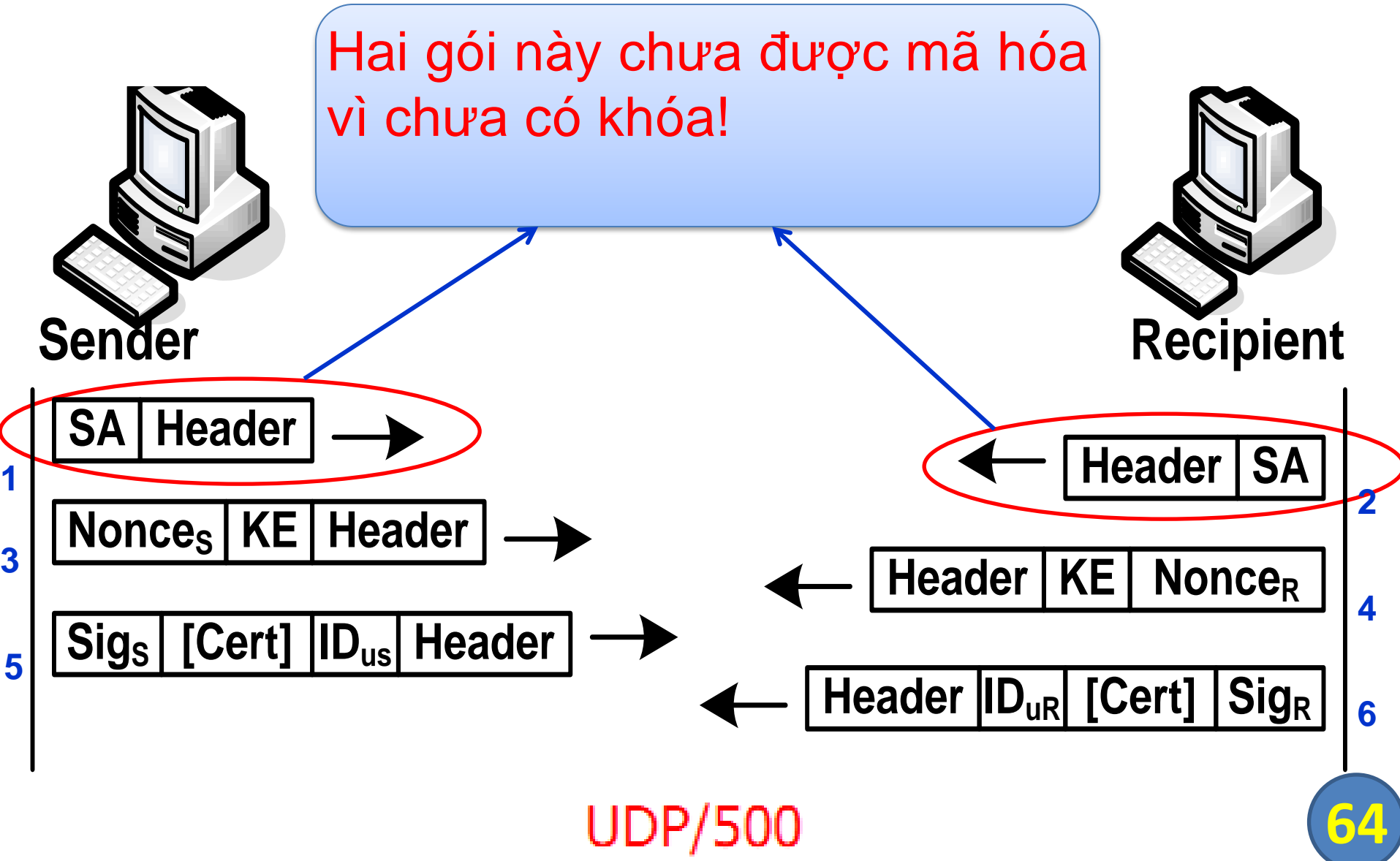
**Bên nhận:**

- Lựa chọn các thuộc tính an toàn trong ds SA nhận được.
- Gửi lại cho bên gửi



# IKE pha 1 - Main mode

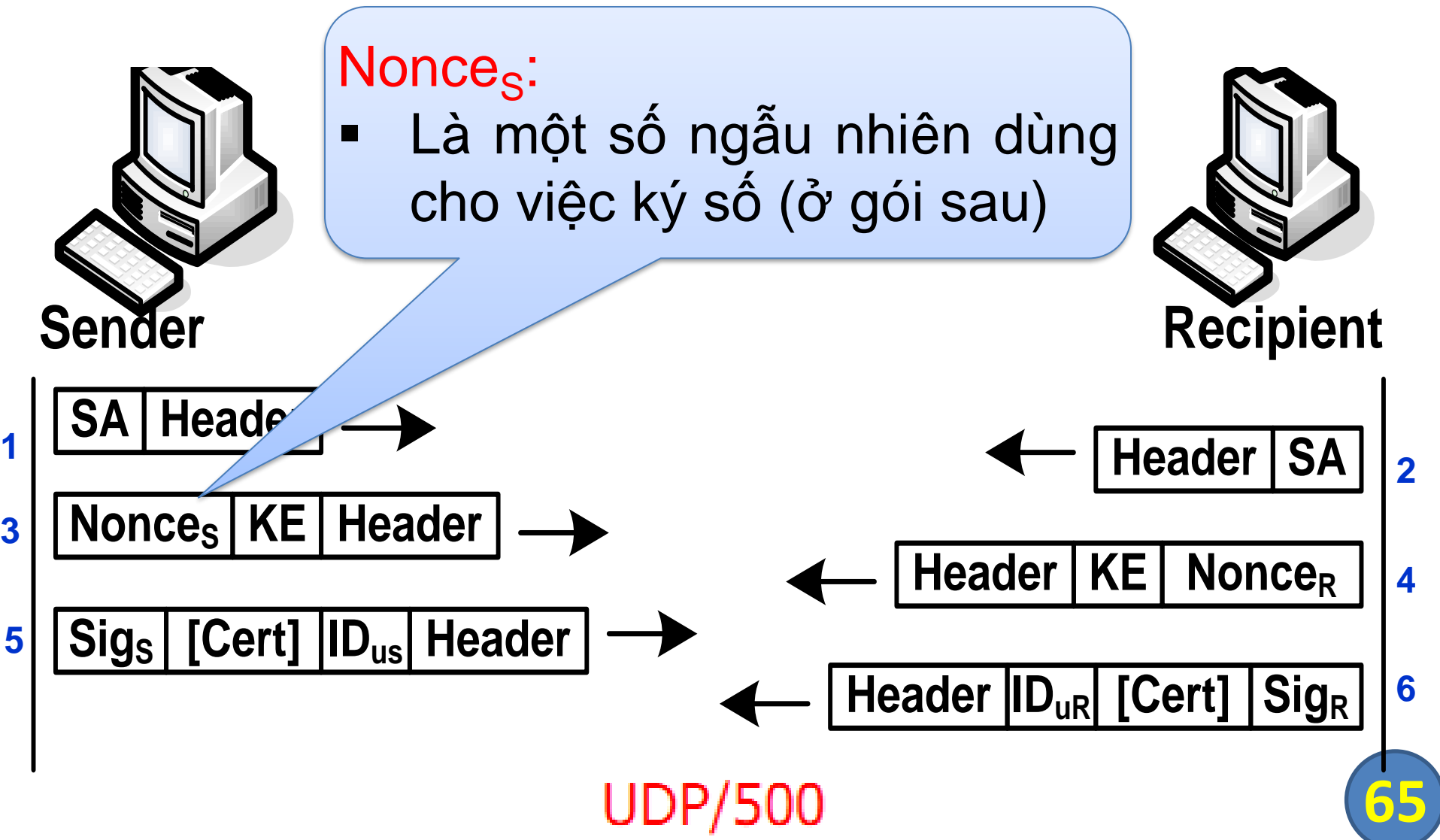
- Mô hình





# IKE pha 1 - Main mode

- Mô hình

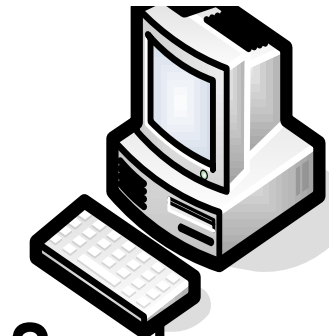


# IKE pha 1 - Main mode

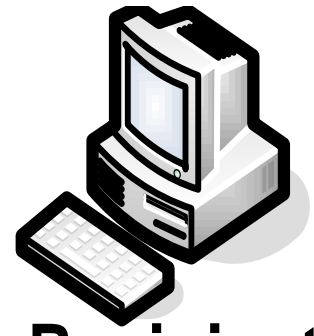
- Mô hình

## KE (Key Exchange) :

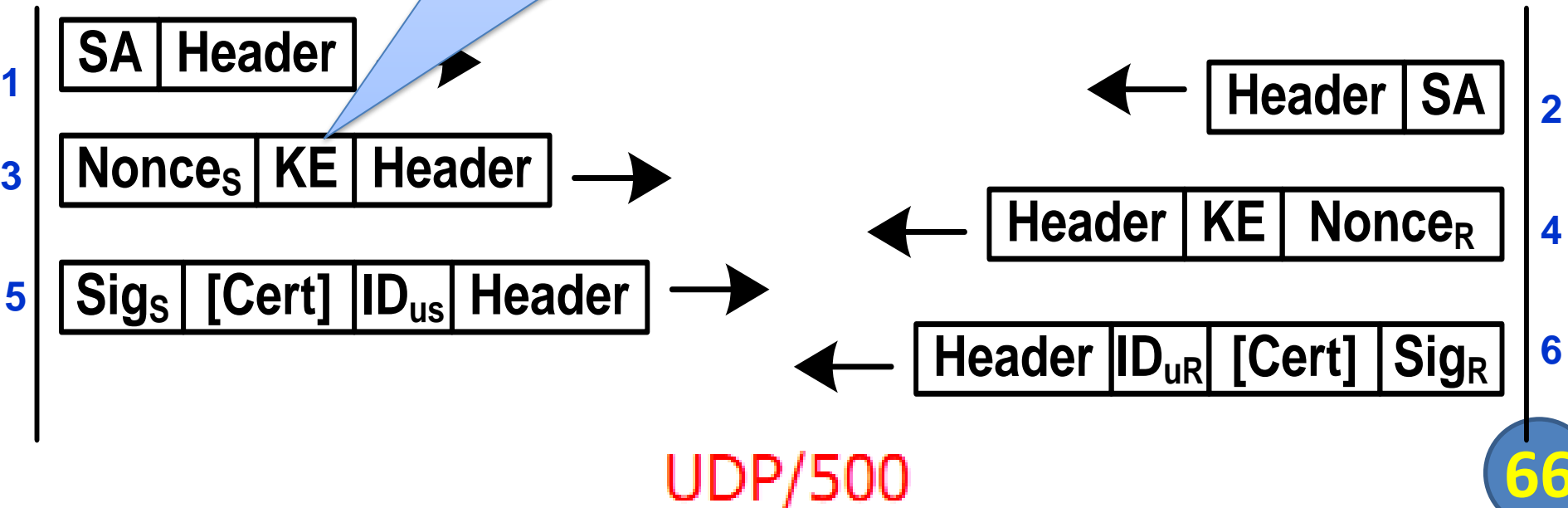
- Là khóa công khai để dùng cho trao đổi khóa DH
- VD:  $y_A = g^{x_A}$



Sender



Recipient

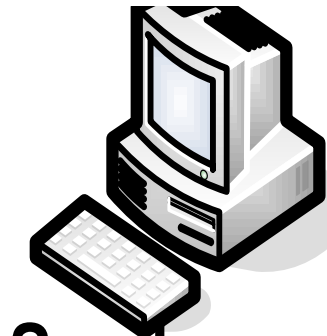


# IKE pha 1 - Main mode

- Mô hình

## Header:

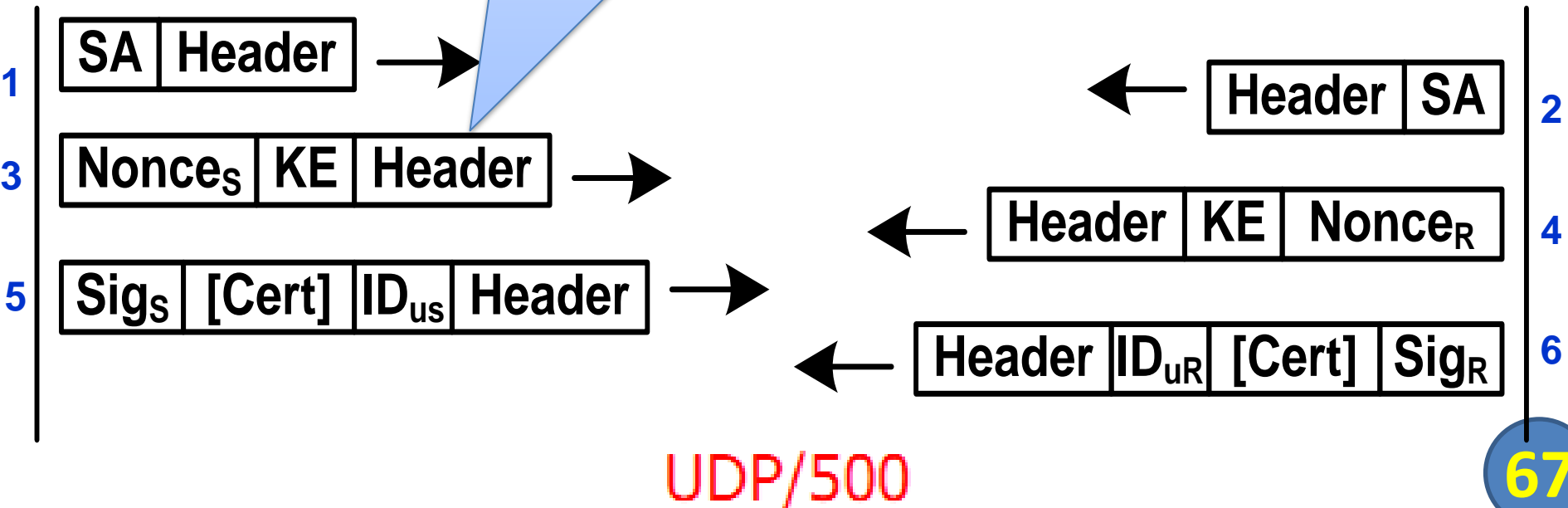
- Có thể chứa tên CA (trung tâm chứng thực) mà bên gửi yêu cầu
- Hoặc (0): any CA



Sender

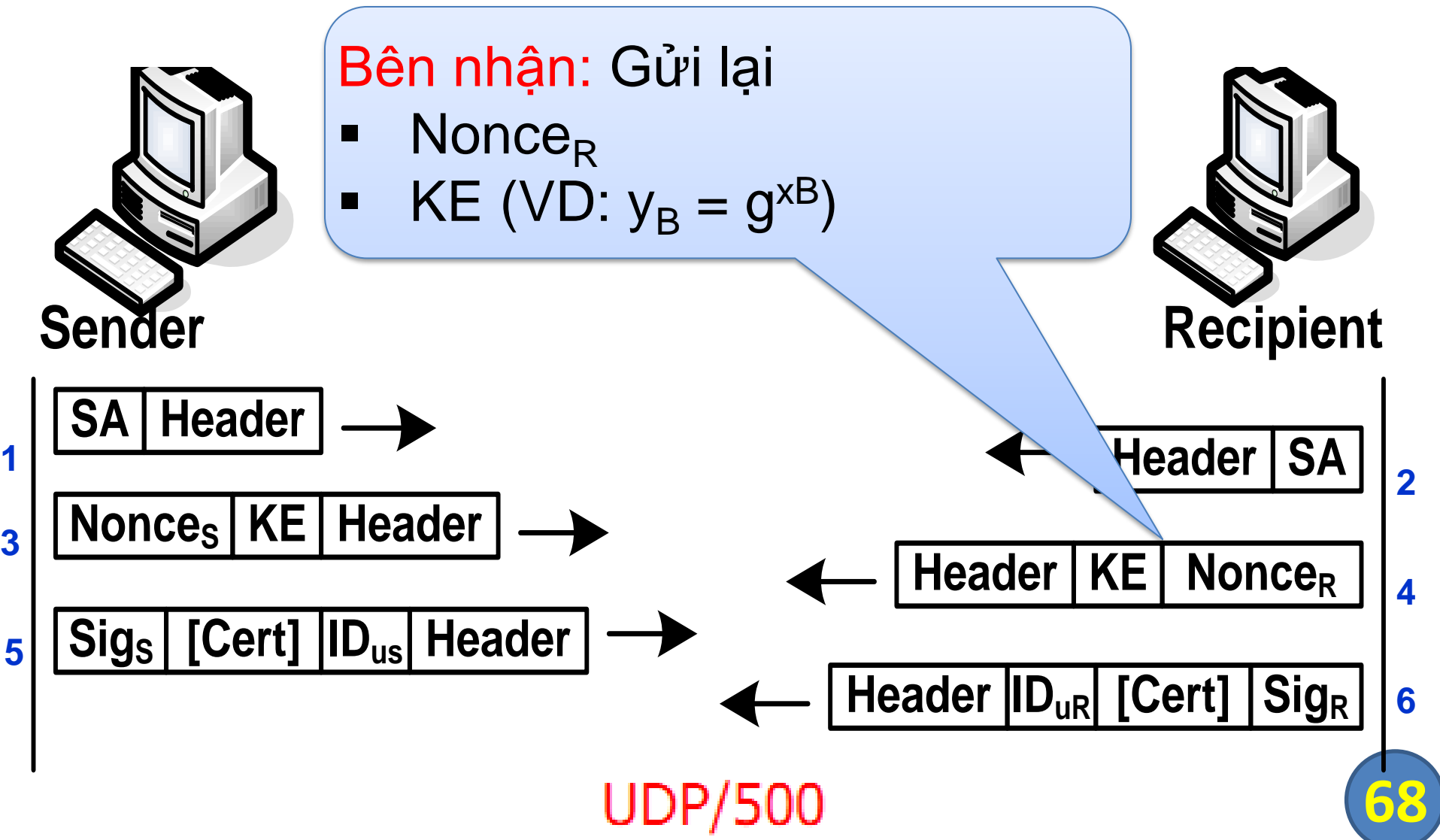


Recipient



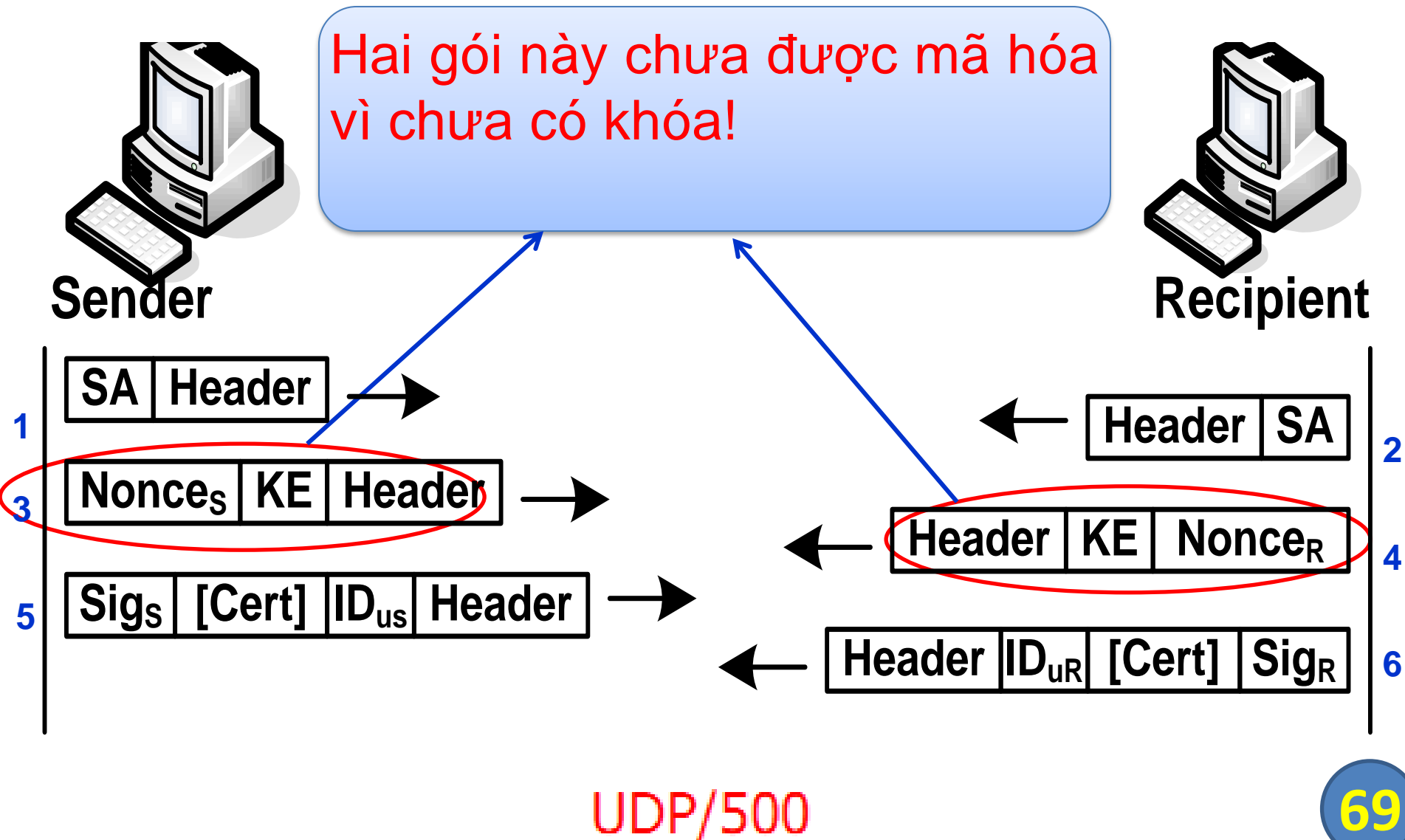
# IKE pha 1 - Main mode

- Mô hình



# IKE pha 1 - Main mode

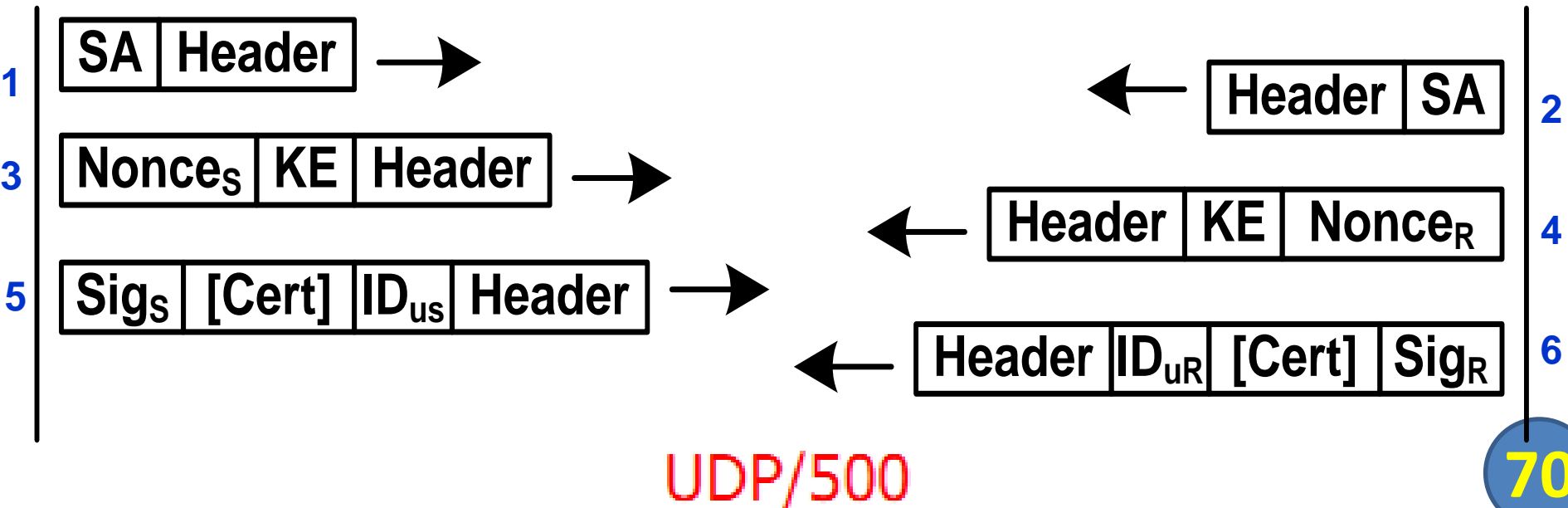
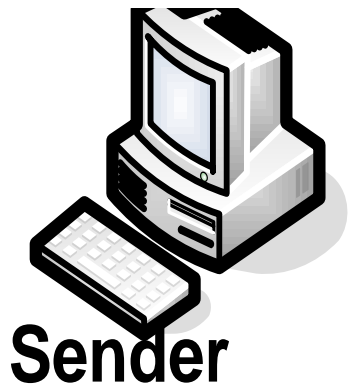
- Mô hình



# IKE pha 1 - Main mode

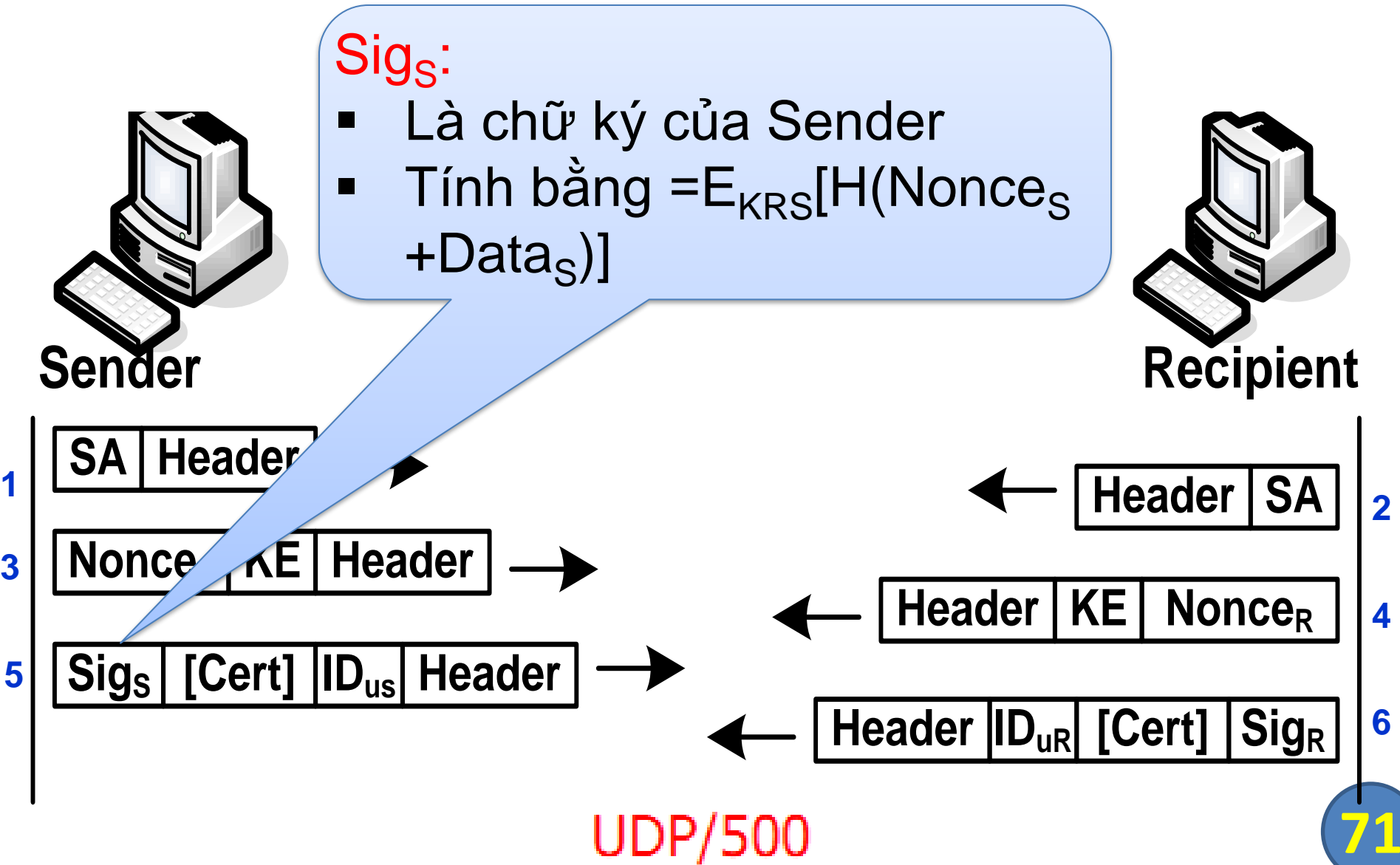
- Mô hình

Sau 2 gói (3),(4), cả hai bên đều có khóa bí mật chung là  $K_M$  để mã hóa (VD:  
 $K_M = g^{xA \cdot xB} \bmod p$ )



# IKE pha 1 - Main mode

- Mô hình

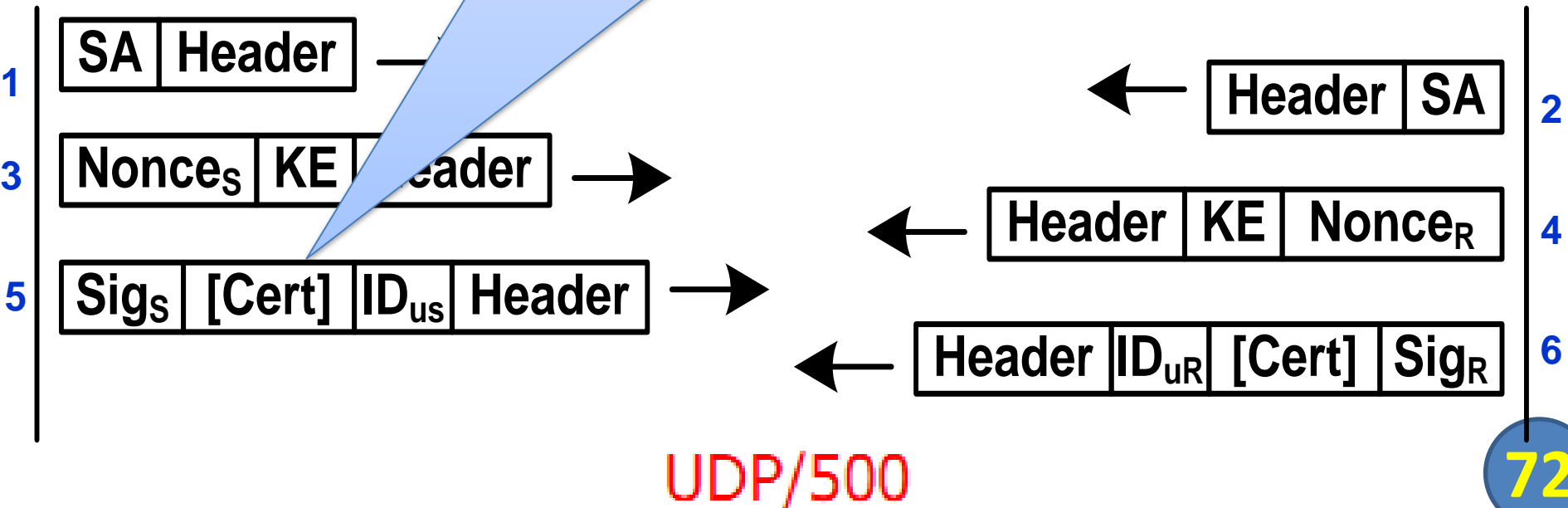
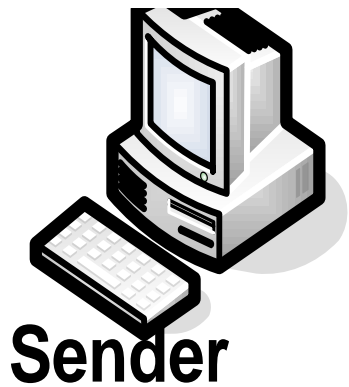


# IKE pha 1 - Main mode

- Mô hình

## Cert:

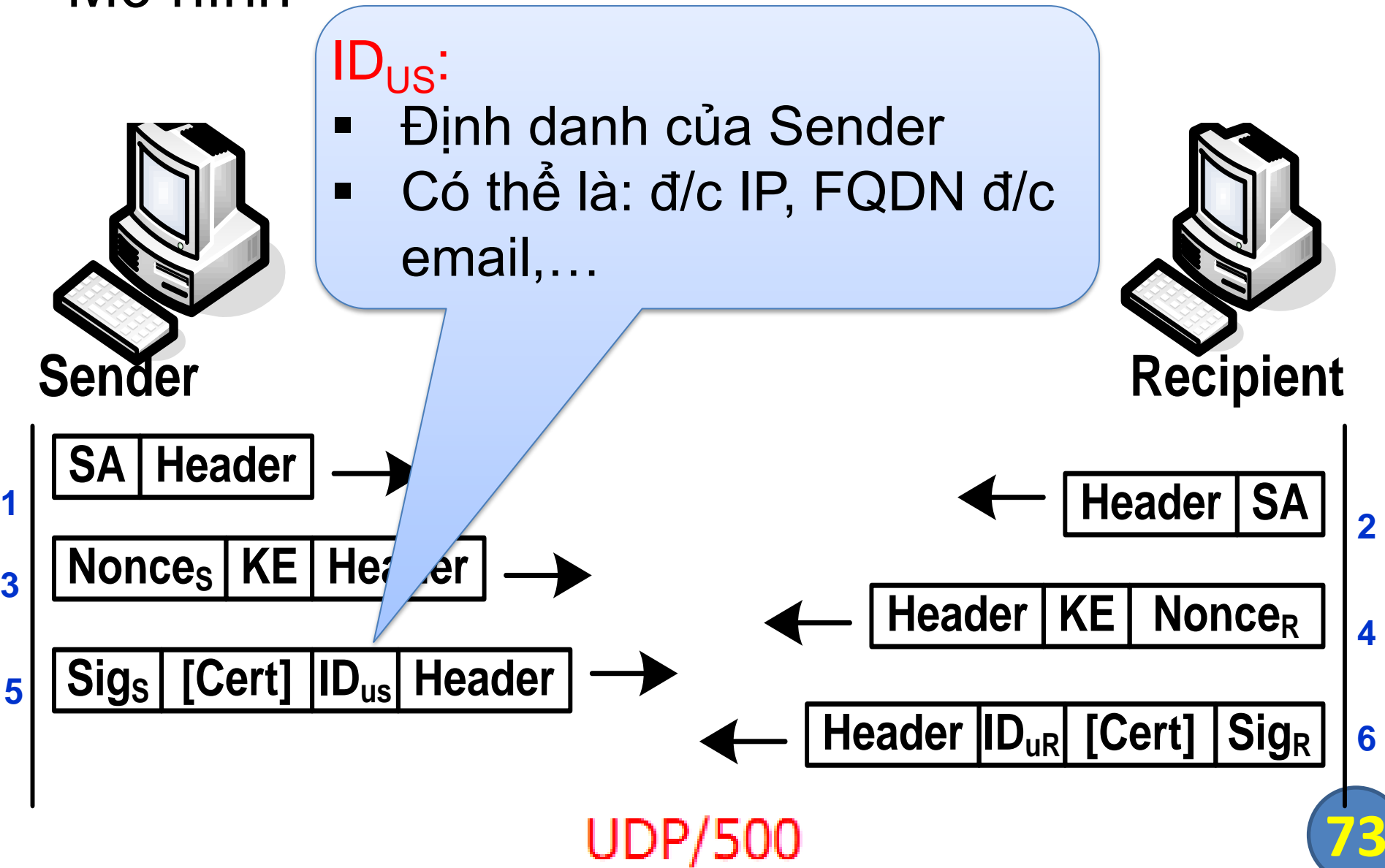
- Là chứng thư số của Sender
- Chứa khóa công khai của Sender **KUS** (tương ứng với KRS)





# IKE pha 1 - Main mode

- Mô hình

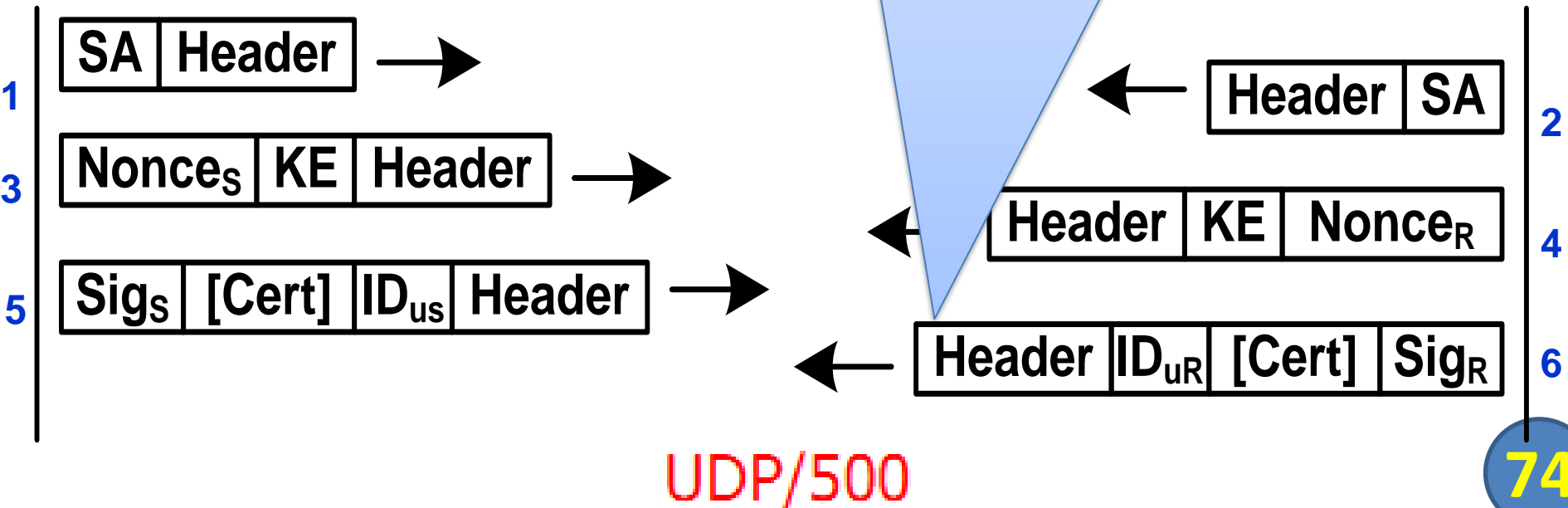
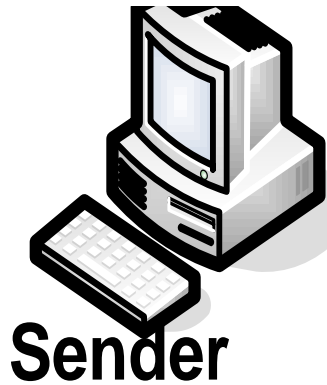


# IKE pha 1 - Main mode

- Mô hình

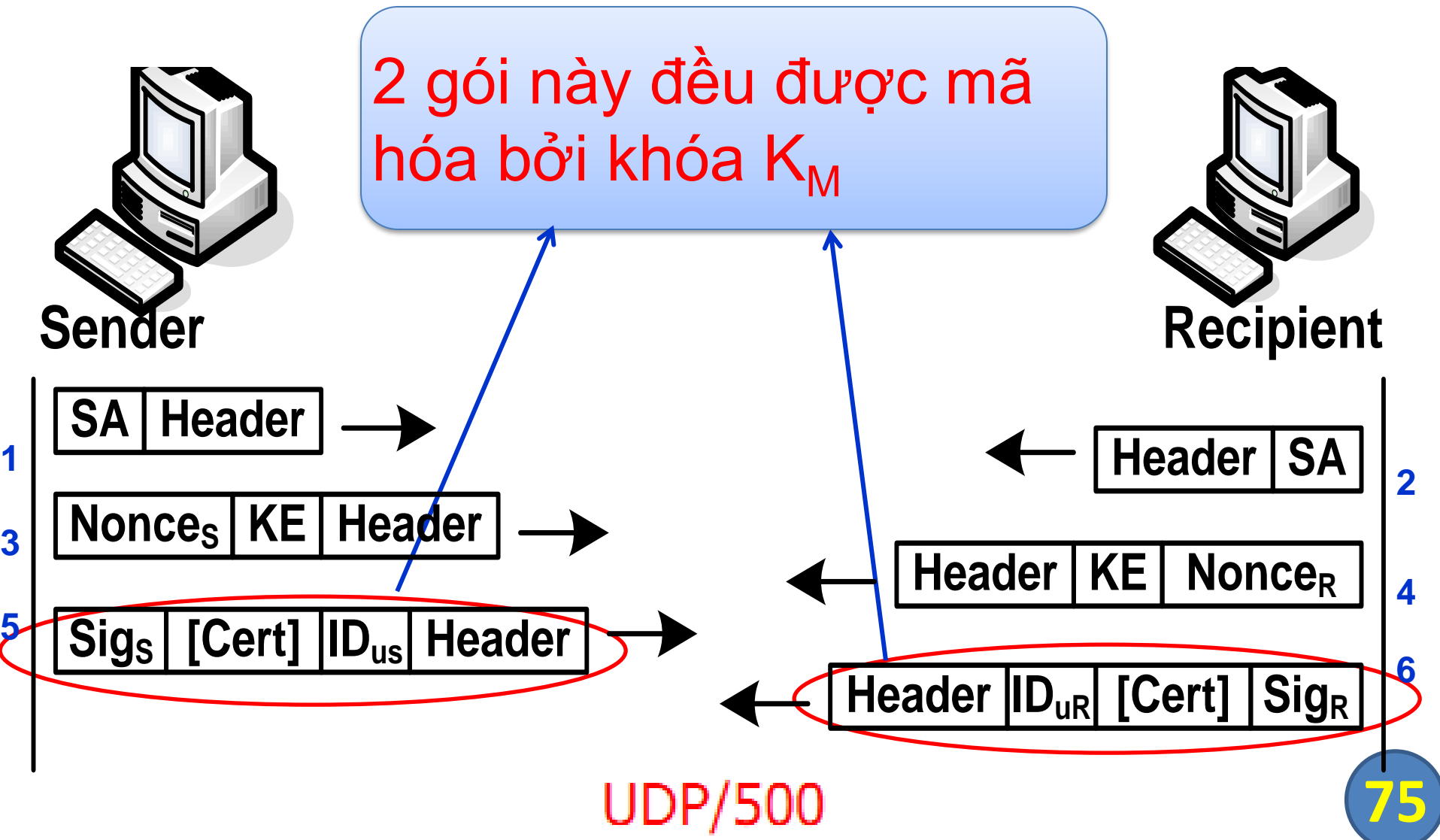
**Bên nhận:** Kiểm tra chữ ký của Sender

- Lấy khóa công khai KUS của Sender trong [Cert<sub>S</sub>]
- Kiểm tra chữ ký:  
$$\text{Sig}_S = E_{K_{RS}}[H(\text{Nonce}_S + \text{Data}_S)]$$



# IKE pha 1 - Main mode

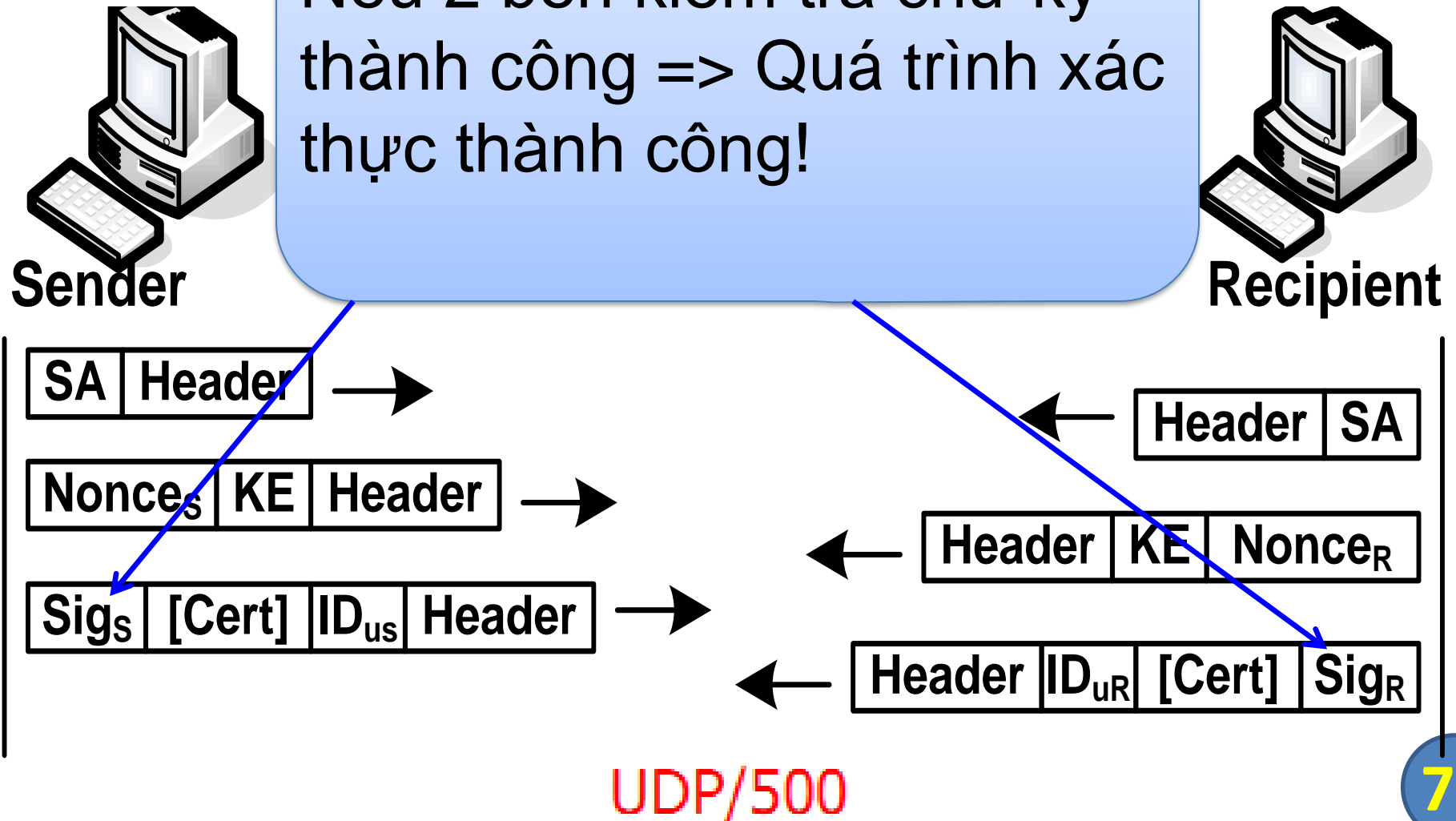
- Mô hình



# IKE pha 1 - Main mode

- Mô hình

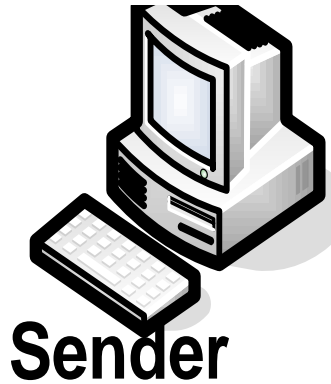
Nếu 2 bên kiểm tra chữ ký thành công => Quá trình xác thực thành công!



# IKE pha 1 - Main mode

- Mô hình **Kết thúc IKE Pha 1:**

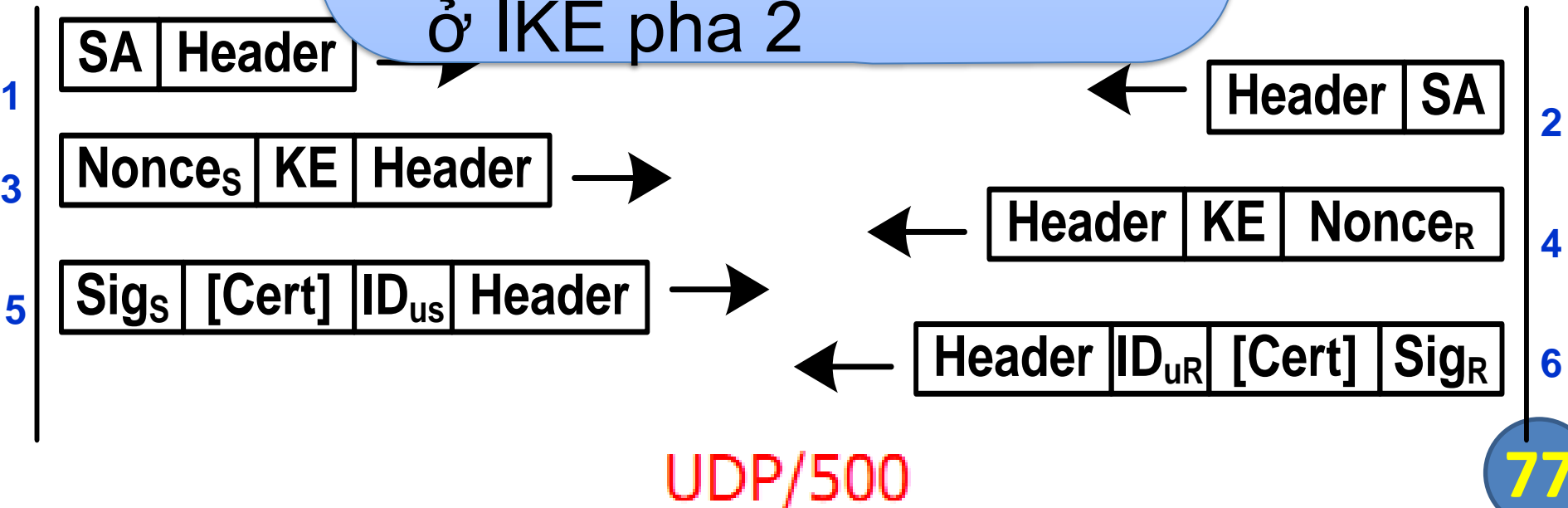
- Hai bên xác thực được nhau.
- Thu được các tham số, thuật toán mật mã, khóa  $K_M$  để mã hóa các gói tin ở IKE pha 2



Sender



Recipient



# Giao thức trao đổi khoá Internet (IKE)

---

## IKE pha 1 - Agressive Mode

# IKE pha 1 - Agressive Mode

- Agressive Mode
  - Được thiết lập tương tự như trong Main Mode
  - Khác ở chỗ là chỉ có 3 thông điệp được trao đổi
    - Nhanh hơn chế độ Main Mode./.

# IKE pha 1 - Agressive Mode

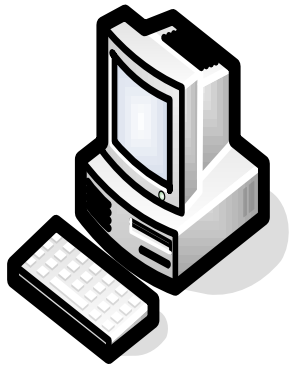
- Message 1: dùng để hỗ trợ chính sách an toàn, dữ liệu tạo khoá, số ngẫu nhiên dùng cho việc ký số và định danh.
- Message 2: để đáp lại thông điệp thứ nhất. Nó xác thực người nhận và thống nhất chính sách an toàn, dữ liệu tạo khoá.
- Message 3: dùng để xác thực người gửi./.



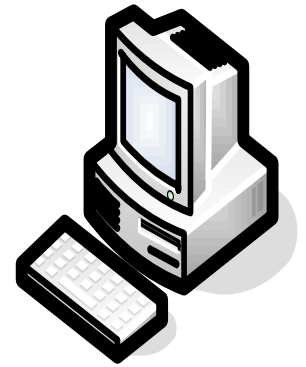
# IKE pha 1 - Agressive Mode

- Mô hình

Tương tự như Main Mode, nhưng các bước được rút gọn hơn, thông điệp dài hơn



Sender



Recipient

ID <sub>us</sub>	Nonce <sub>s</sub>	[KE]	SA	Header
------------------	--------------------	------	----	--------

 →

← 

Header	SA	[KE]	Nonce <sub>R</sub>	[ID <sub>ur</sub> ]	[Cert]	Sig <sub>R</sub>
--------	----	------	--------------------	---------------------	--------	------------------

Sig <sub>R</sub>	[Cert]	Header
------------------	--------	--------

 →

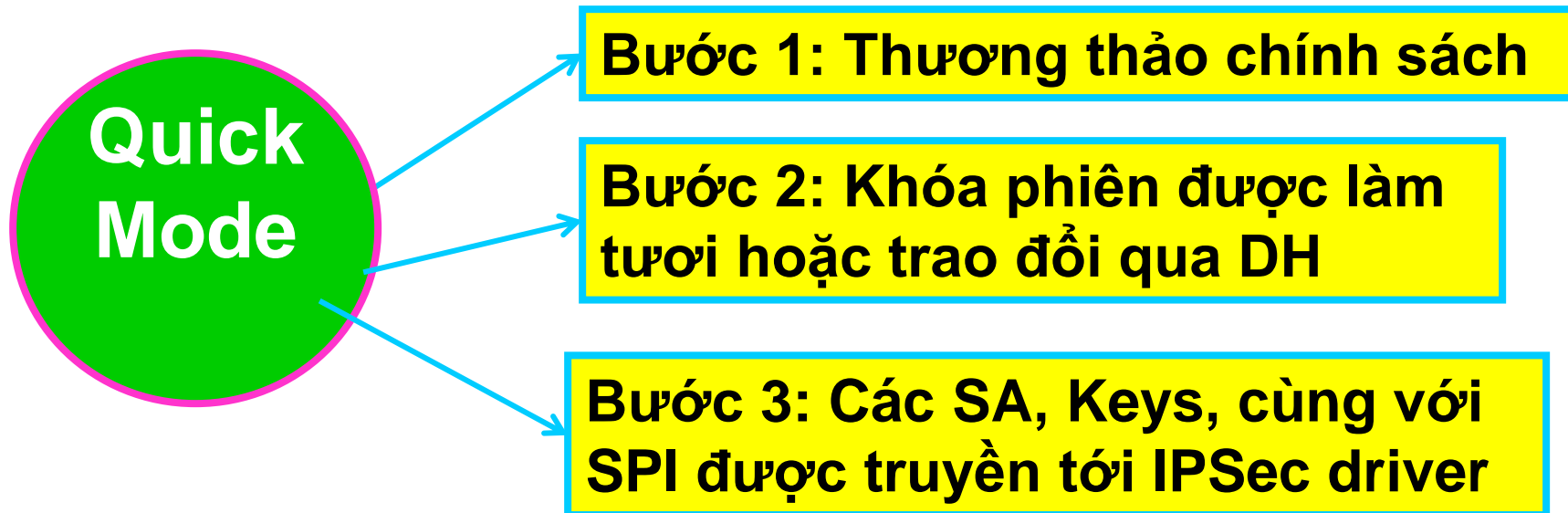
# Giao thức trao đổi khoá Internet (IKE)

---

## IKE pha 2

# Giao thức trao đổi khoá Internet (IKE)

- IKE pha 2 - Quick mode:



# Giao thức trao đổi khoá Internet (IKE)

- IKE pha 2 - Quick mode:

**Quick  
Mode**

**Bước 1: Thương thảo chính sách**

- Một bên sẽ đưa ra một danh sách các thuật toán mật mã (SA).
- Bên nhận sẽ lựa chọn hoặc có yêu cầu khác.

# Giao thức trao đổi khoá Internet (IKE)

- IKE pha 2 - Quick mode:

**Quick  
Mode**

**Bước 1: Thương thảo chính sách**

Sau khi thương lượng xong. 2 SA được thiết lập cho mỗi bên

- Một SA cho lưu lượng INBOUND
- Một SA cho lưu lượng OUTBOUND

# Giao thức trao đổi khoá Internet (IKE)

- IKE pha 2 - Quick mode:



**Bước 2: Khóa phiên được làm tươi hoặc trao đổi qua DH**

Các khóa này làm nhiệm vụ cho: xác thực, toàn vẹn, mã hóa (nếu cần) trong phiên IPSec.

Có hai lựa chọn:

- Làm tươi khóa  $K_M$  thu được bằng DH trong pha 1
- Thực hiện trao đổi khóa DH lần 2, để thu được  $K_s$  (rekey)

# Giao thức trao đổi khoá Internet (IKE)

- IKE pha 2 - Quick mode:



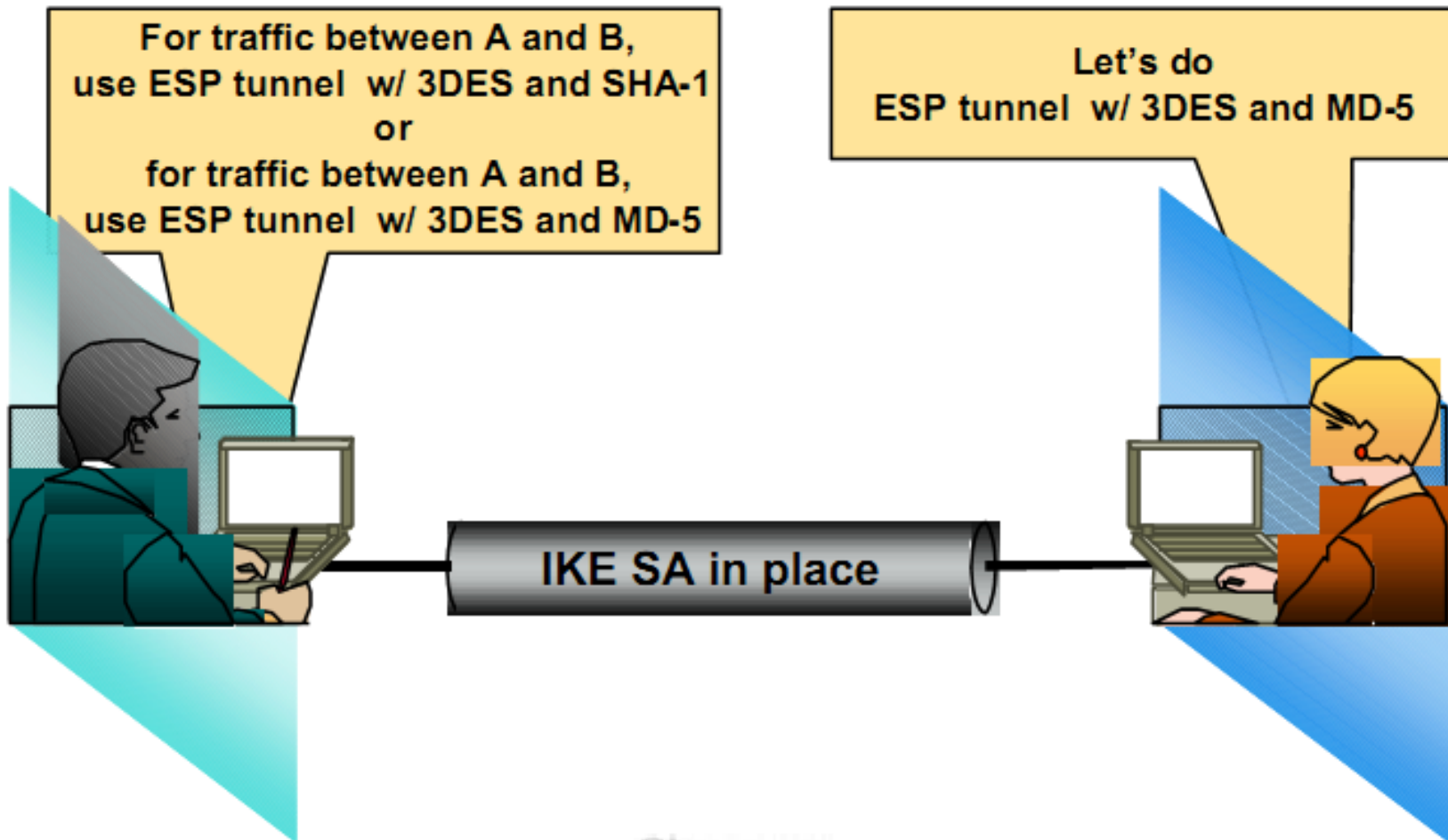
**Bước 3: Các SA, Keys, cùng với SPI được truyền tới IPSec driver**

## Kết quả pha 2:

- Là một cặp SA mới (inbound & outbound) được dùng để bảo vệ lưu lượng IP
- Mỗi SA có SPI và key riêng của nó
- Các khóa mới được tạo cho: xác thực, toàn vẹn hay mã hóa.
- Sau khi cặp SA mới được tạo ra, cặp SA cũ bị xóa, và lưu lượng được bảo vệ với cặp SA mới

# Giao thức trao đổi khoá Internet (IKE)

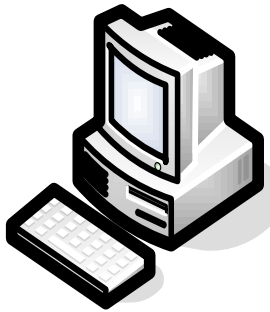
- IKE pha 2 - Ví dụ: (thương lượng IPsec SA)



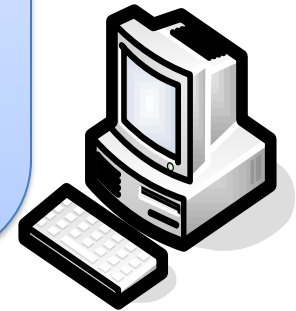


# IKE pha 2 - Quick mode

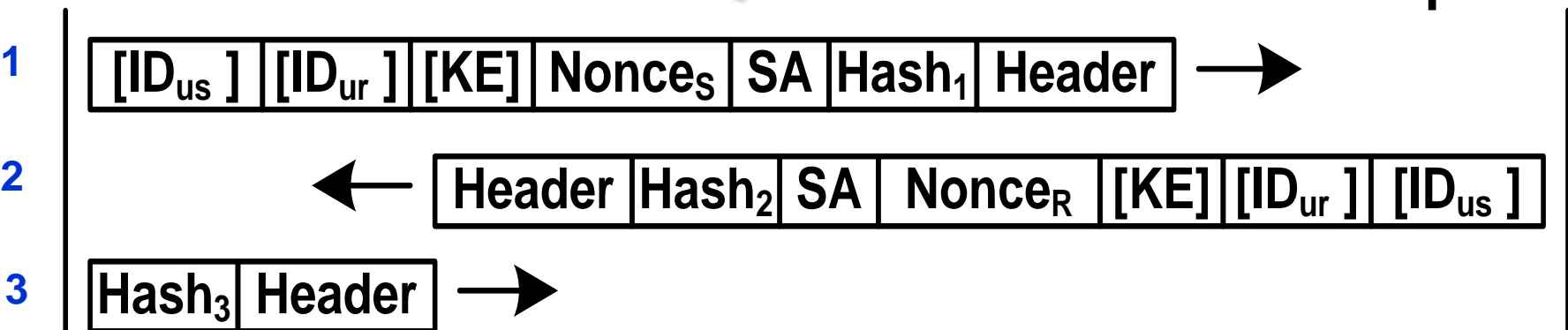
- Mô hình 3 gói làm nhiệm vụ:
  - Thương lượng SA
  - Rekey hoặc refresh khóa bằng DH.
  - Xác thực lẫn nhau



Sender



Recipient



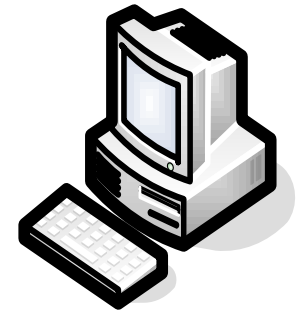
# IKE pha 2 - Quick mode

- Mô hình

Cả 3 gói đều được mã hóa bằng khóa  $K_M$  ở pha 1



Sender



Recipient

1



2



3



# IKE pha 2 - Quick mode

- Mô hình

SA (ds các thuộc tính an toàn):

- cipher?
- HMAC?
- Key length?
- IPSec protocol? (AH, ESP)?, ...



Sender



Recipient

1



2



3

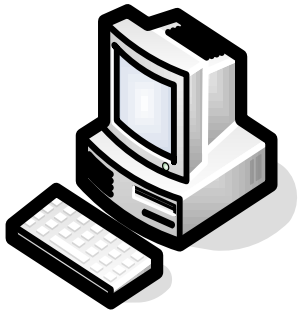


# IKE pha 2 - Quick mode

- Mô hình

**KE:**

- Khóa công khai để trao đổi khóa DH (lần 2)
- Tạo ra một khóa phiên mới  $K_S$  cho xác thực, mã hóa lưu lượng IP thực (phiên IPSec)
- Có thể không cần khóa này, lấy luôn khóa  $K_M$  ở pha trước



Sender



Recipient

1

[ID<sub>us</sub>] [ID<sub>ur</sub>] [KE] Nonce<sub>s</sub> SA Hash<sub>1</sub> Header →

2

← Header Hash<sub>2</sub> SA Nonce<sub>R</sub> [KE] [ID<sub>ur</sub>] [ID<sub>us</sub>]

3

Hash<sub>3</sub> Header →

# IKE pha 2 - Quick mode

- Mô hình

## Kết thúc pha 2:

- Tạo ra 2 SA:
  - + Một SA Inbound
  - + Một SA Outbound
- Mỗi SA có SPI và key riêng.



Sender



Recipient

1

[ID<sub>us</sub>] [ID<sub>ur</sub>] [KE] Nonce<sub>s</sub> SA Hash<sub>1</sub> Header →

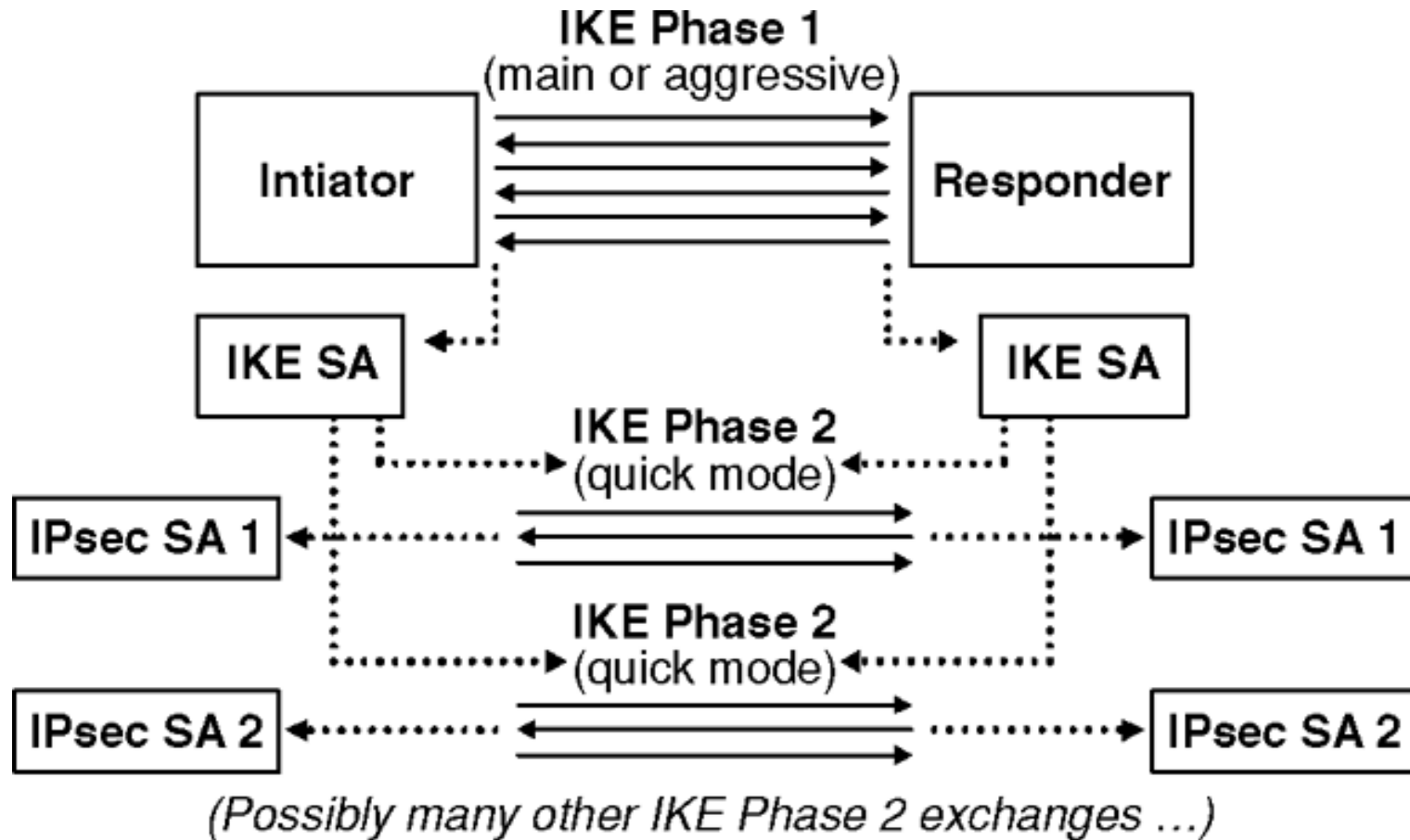
2

← Header Hash<sub>2</sub> SA Nonce<sub>R</sub> [KE] [ID<sub>ur</sub>] [ID<sub>us</sub>]

3

Hash<sub>3</sub> Header →

# Giao thức trao đổi khoá Internet (IKE)



# Ví dụ về hai pha IKE

**IPSec Setup**

Keying Mode: IKE with Preshared Key ▾

**Phase 1:**

Encryption: 3DES ▾

Authentication: MD5 ▾

Group: 768-bit ▾

Key Lifetime: 28800 sec

**Phase 2:**

Encryption: 3DES ▾

Authentication: SHA1 ▾

Perfect Forward Secrecy: Enable ▾

Preshared Key: name

Group: 768-bit ▾

Key Lifetime: 3600 sec

# Nhược điểm của IKE v1

- Có quá nhiều tùy chọn an toàn dẫn tới việc khó quản lý, kiểm soát.
- Thực hiện nhiều bước trao đổi, phức tạp, dẫn đến tính hiệu quả kém, khó đánh giá và phân tích độ an toàn.
- Không hỗ trợ phương pháp xác thực mở rộng EAP.
- Bị tấn công phản xạ ngay cả ở chế độ Main mode xác thực sử dụng khóa chia sẻ trước hoặc sử dụng chứng chỉ số [].



# Tổng quan về giao thức IKEv2

- Ra đời 10/2005 trong RFC 4306.
- Được phát triển nhằm giải quyết những vấn đề của IKEv1
- IKEv2 cũng bao gồm hai pha.
  - Pha 1 gồm hai thủ tục IKE\_SA\_INIT và IKE\_AUTH.
  - Pha 2 gồm thủ tục CREAT\_CHILD\_SA và/ hoặc có thêm thủ tục INFORMATION.
- (Sinh viên tham khảo thêm trong giáo trình, trên mạng)

# Ứng dụng của giao thức IKE.

- Được sử dụng rất rộng rãi.
  - Sử dụng như một phần của bộ giao thức IPSec - có thể triển khai trên Windows Server 2008, Windows Server 2012, Windows Server 2016, Windows Server 2019.
  - Trong các sản phẩm mã nguồn mở.
    - OpenIkev2.
    - StrongSwan.
    - Openswan.
    - Racoon và Racoon2 của dự án KAM.
    - ...

# Giao thức IPSec (Tổng kết)

Giới thiệu về IPSec

Tổ hợp an toàn (SA) trong IPSec

 Giao thức xác thực tiêu đề (AH)

Giao thức ESP

Sự kết hợp của AH và ESP

Giao thức trao đổi khóa IKE

