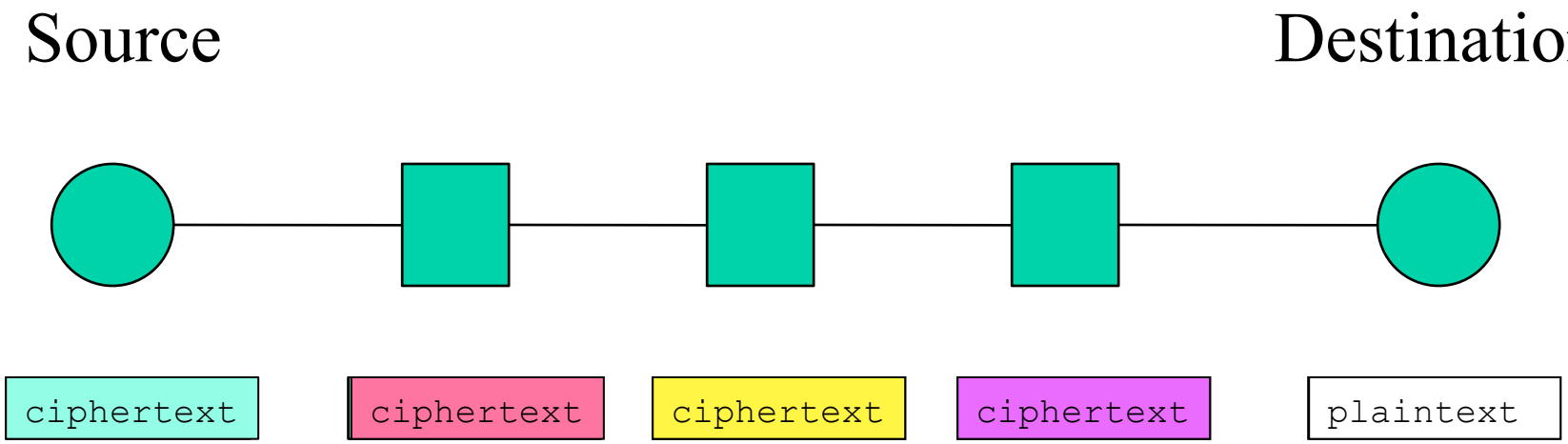
# Encryption and Network Security

- Cryptography is widely used to protect networks
- Relies on encryption algorithms and protocols discussed previously
- Can be applied at different places in the network stack
- With different effects and costs

# Link Level Encryption



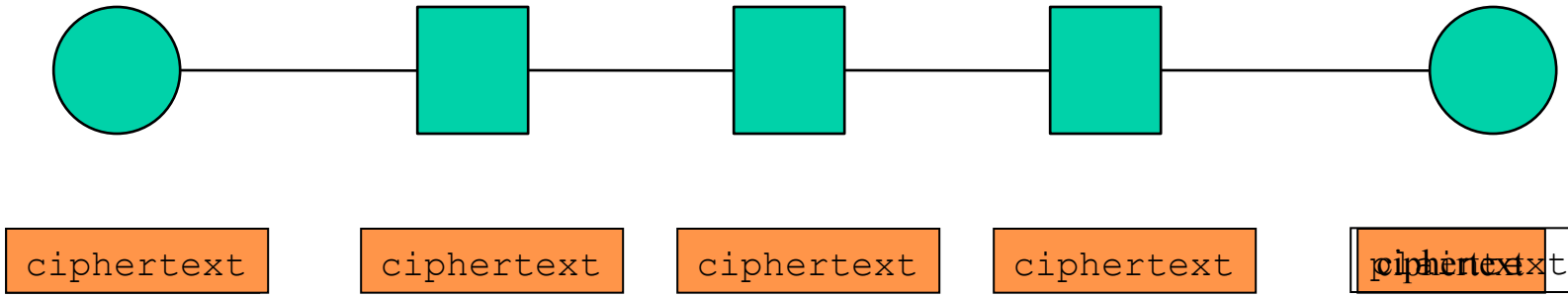

Let's say we want to send a message using encryption

Different keys (maybe even different ciphers) used at each hop

# End-to-End Encryption

Source

Destination




Cryptography only at the end points

Only the end points see the plaintext

Normal way network cryptography done

When would link encryption be better?

# Where Are the Endpoints, Anyway?

- If you do end-to-end encryption, where are the endpoints?
- The network layer end points?
- The transport layer end points?
- The application layer end points?
- Maybe not even end machine to end machine (e.g., VPNs)?
- Has serious implications for where you do cryptography
  - And keying and trust issues

# IPsec

- Standard for applying cryptography at the network layer of IP stack
- Provides various options for encrypting and authenticating packets
  - On end-to-end basis
  - Without concern for transport layer (or higher)

# What IPsec Covers

- Message integrity
- Message authentication
- Message confidentiality

# What Isn't Covered

- Non-repudiation
- Digital signatures
- Key distribution
- Traffic analysis
- Handling of security associations
- Some of these covered in related standards

# Some Important Terms for IPsec

- Security Association - “A Security Association (SA) is a simplex ‘connection’ that affords security services to the traffic carried by it.”
  - Basically, a secure one-way channel
- SPI (Security Parameters Index) – Combined with destination IP address and IPsec protocol type, uniquely identifies an SA

# General Structure of IPsec

- Really designed for end-to-end encryption
  - Though could do link level
- Designed to operate with either IPv4 or IPv6
- Meant to operate with a variety of different ciphers
- And to be neutral to key distribution methods
- Has sub-protocols
  - E.g., Encapsulating Security Payload

# Encapsulating Security Payload (ESP) Protocol

- Encrypt the data and place it within the ESP
- The ESP has normal IP headers
- Can be used to encrypt just the payload of the packet
- Or the entire IP packet

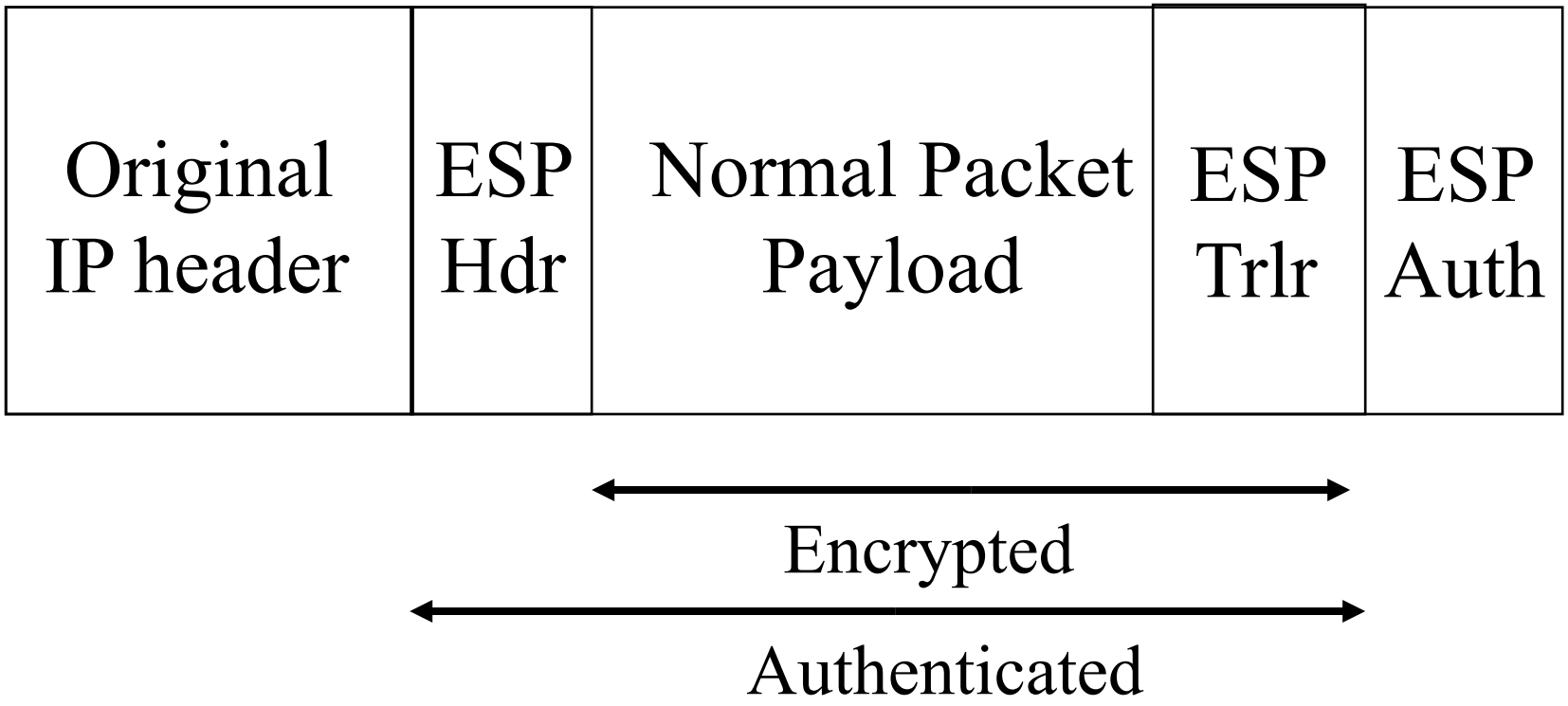
# ESP Modes

- Transport mode
  - Encrypt just the transport-level data in the original packet
  - No IP headers encrypted
- Tunnel mode
  - Original IP datagram is encrypted and placed in ESP
  - Unencrypted headers wrapped around ESP

# ESP in Transport Mode

- Extract the transport-layer frame
  - E.g., TCP, UDP, etc.
- Encapsulate it in an ESP
- Encrypt it
- The encrypted data is now the last payload of a cleartext IP datagram

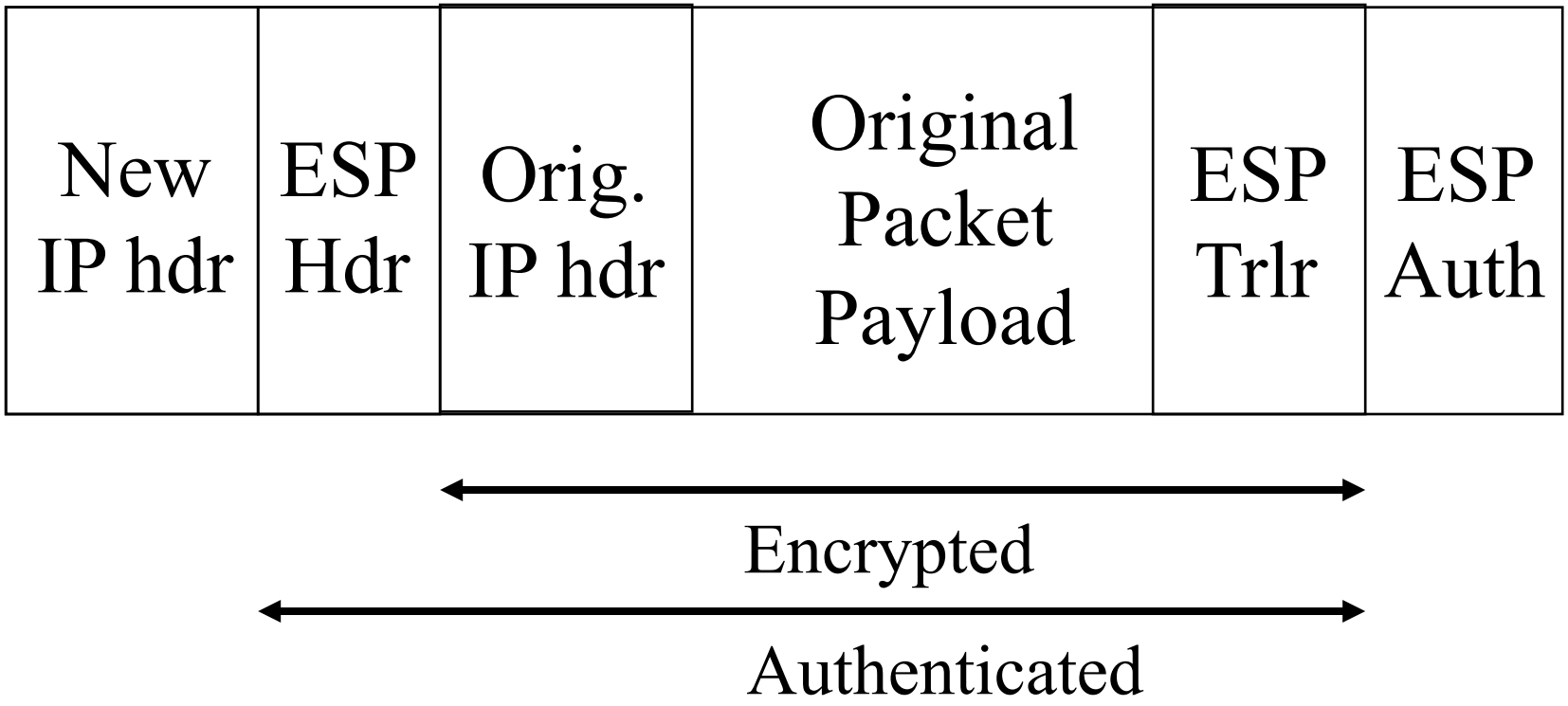
# ESP Transport Mode



# Using ESP in Tunnel Mode

- Encrypt the IP datagram
  - The entire datagram
- Encapsulate it in a cleartext IP datagram
- Routers not understanding IPsec can still handle it
- Receiver reverses the process

# ESP Tunnel Mode



# Uses and Implications of Tunnel Mode

- Typically used when there are security gateways between sender and receiver
  - And/or sender and receiver don't speak IPsec
- Outer header shows security gateway identities
  - Not identities of real parties
- Can thus be used to hide some traffic patterns

# What IPsec Requires

- Protocol standards
  - To allow messages to move securely between nodes
- Supporting mechanisms at hosts running IPsec
  - E.g., a Security Association Database
- Lots of plug-in stuff to do the cryptographic heavy lifting

# The Protocol Components

- Pretty simple
- Necessary to interoperate with non-IPsec equipment
- So everything important is inside an individual IP packet's payload
- No inter-message components to protocol
  - Though some security modes enforce inter-message invariants at endpoints

# The Supporting Mechanisms

- Methods of defining security associations
- Databases for keeping track of what's going on with other IPsec nodes
  - To know what processing to apply to outgoing packets
  - To know what processing to apply to incoming packets

# Plug-In Mechanisms

- Designed for high degree of generality
- So easy to plug in:
  - Different crypto algorithms
  - Different hashing/signature schemes
  - Different key management mechanisms

# Status of IPsec

- Accepted Internet standard
- Widely implemented and used
  - Supported in Windows 2000, XP, Vista, Windows 7, Windows 8
  - In Linux 2.6 (and later) kernel
- The architecture doesn't require everyone to use it
- RFC 3602 on using AES in IPsec still listed as "proposed"
- AES will become default for ESP in IPsec