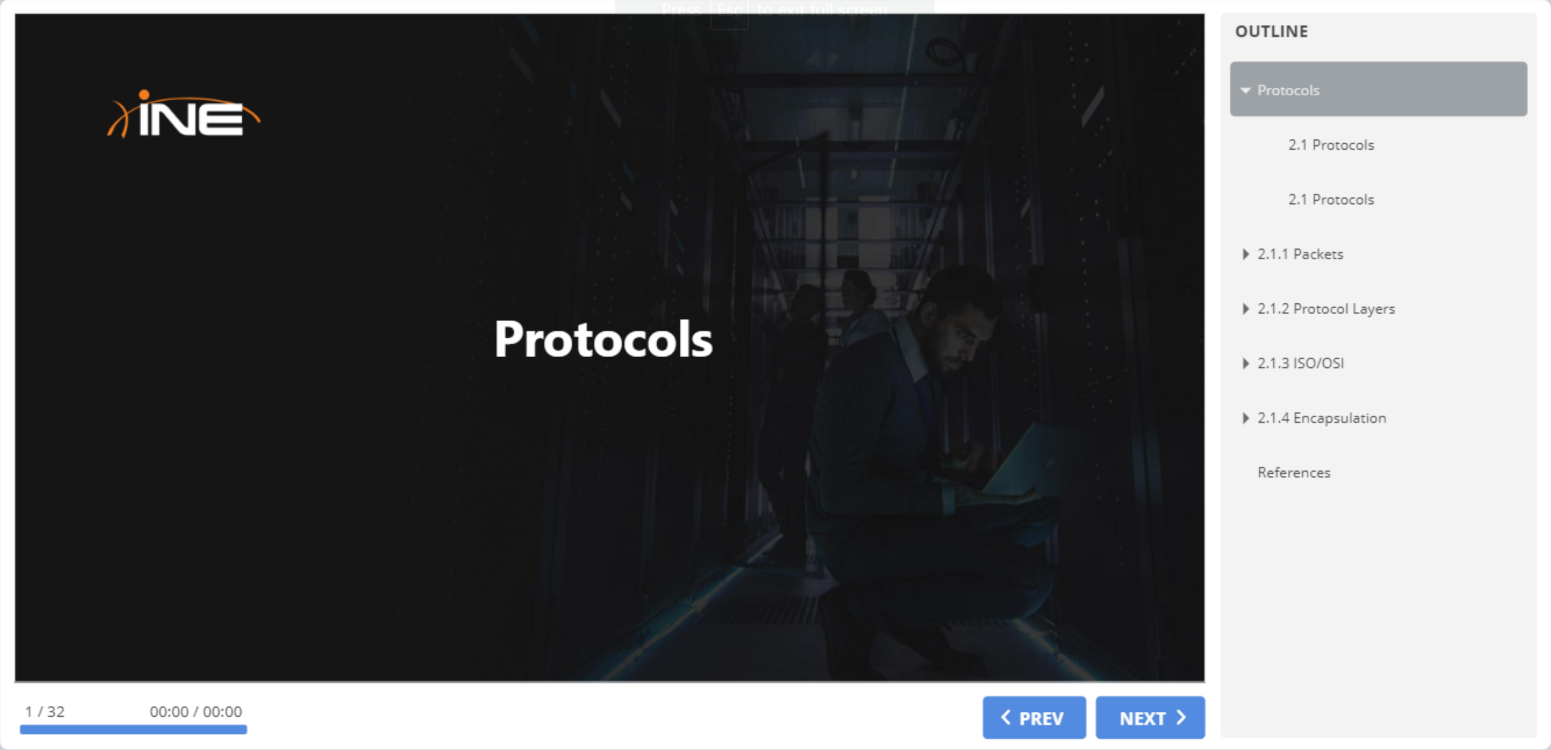
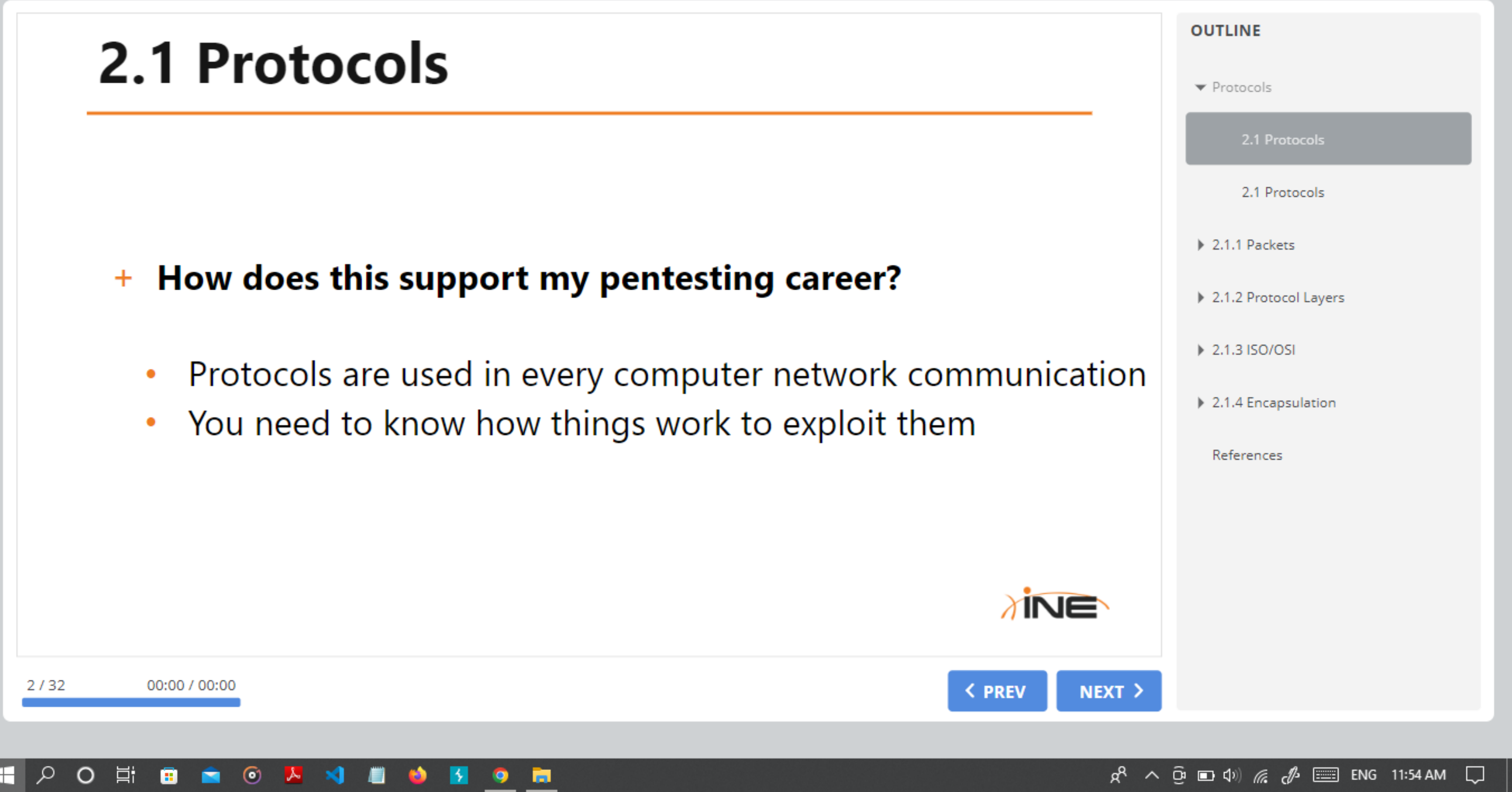


Protocols

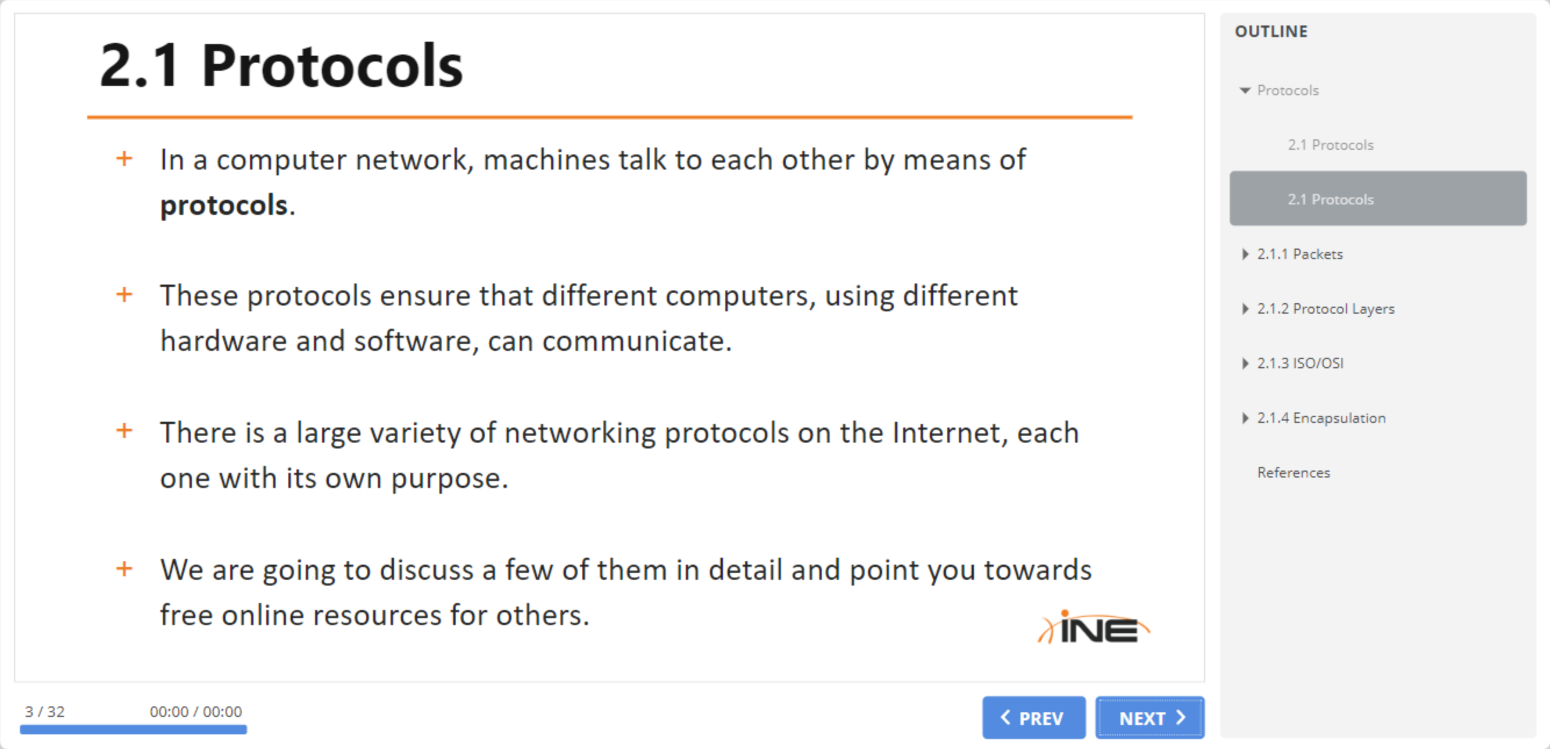
Press Esc to exit full screen
INE
OUTLINE
Protocols
2.1 Protocols
2.1 Protocols
2.1.1 Packets
2.1.2 Protocol Layers
2.1.3 ISO/OSI
2.1.4 Encapsulation
References
1 / 32
00:00 / 00:00
PREV
NEXT



2.1 Protocols

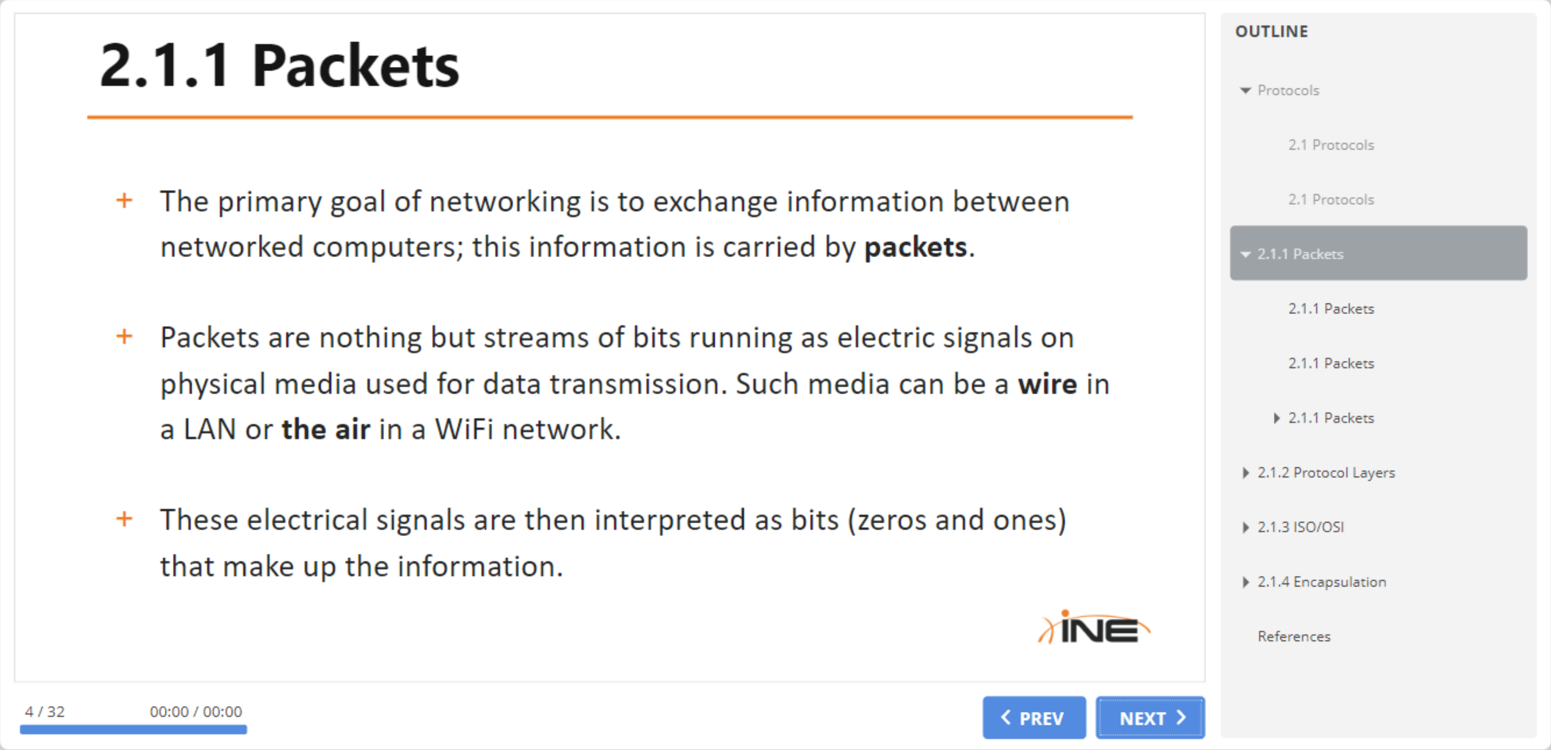
+ How does this support my pentesting career?

- Protocols are used in every computer network communication
- You need to know how things work to exploit them



2.1 Protocols

- In a computer network, machines talk to each other by means of **protocols**.
- These protocols ensure that different computers, using different hardware and software, can communicate.
- There is a large variety of networking protocols on the Internet, each one with its own purpose.
- We are going to discuss a few of them in detail and point you towards free online resources for others.



2.1.1 Packets

- The primary goal of networking is to exchange information between networked computers; this information is carried by **packets**.
- Packets are nothing but streams of bits running as electric signals on physical media used for data transmission. Such media can be a **wire** in a LAN or **the air** in a WiFi network.
- These electrical signals are then interpreted as bits (zeros and ones) that make up the information.

OUTLINE

Protocols

2.1 Protocols

2.1 Protocols

2.1.1 Packets

2.1.1 Packets

2.1.1 Packets

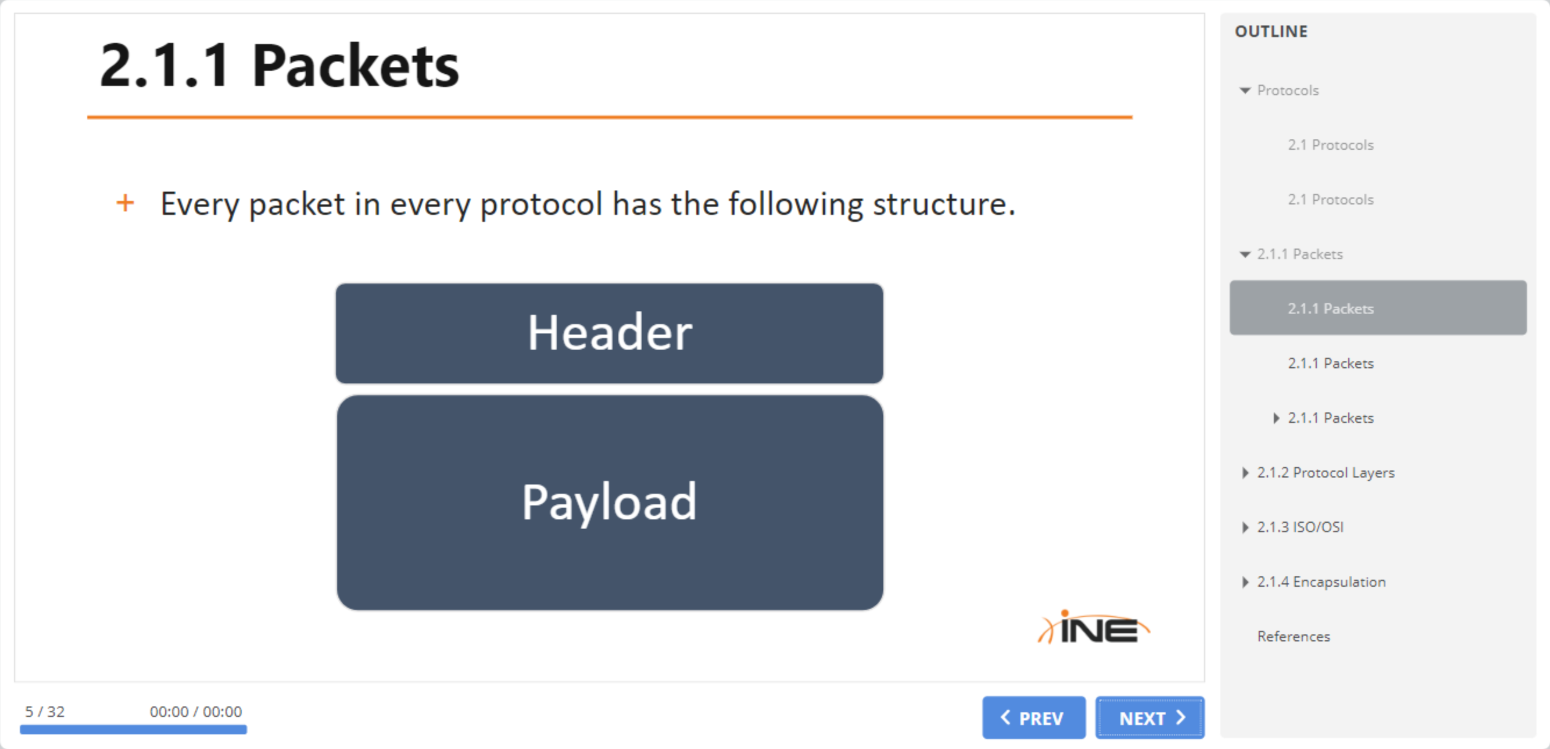
2.1.1 Packets

2.1.2 Protocol Layers

2.1.3 ISO/OSI

2.1.4 Encapsulation

References

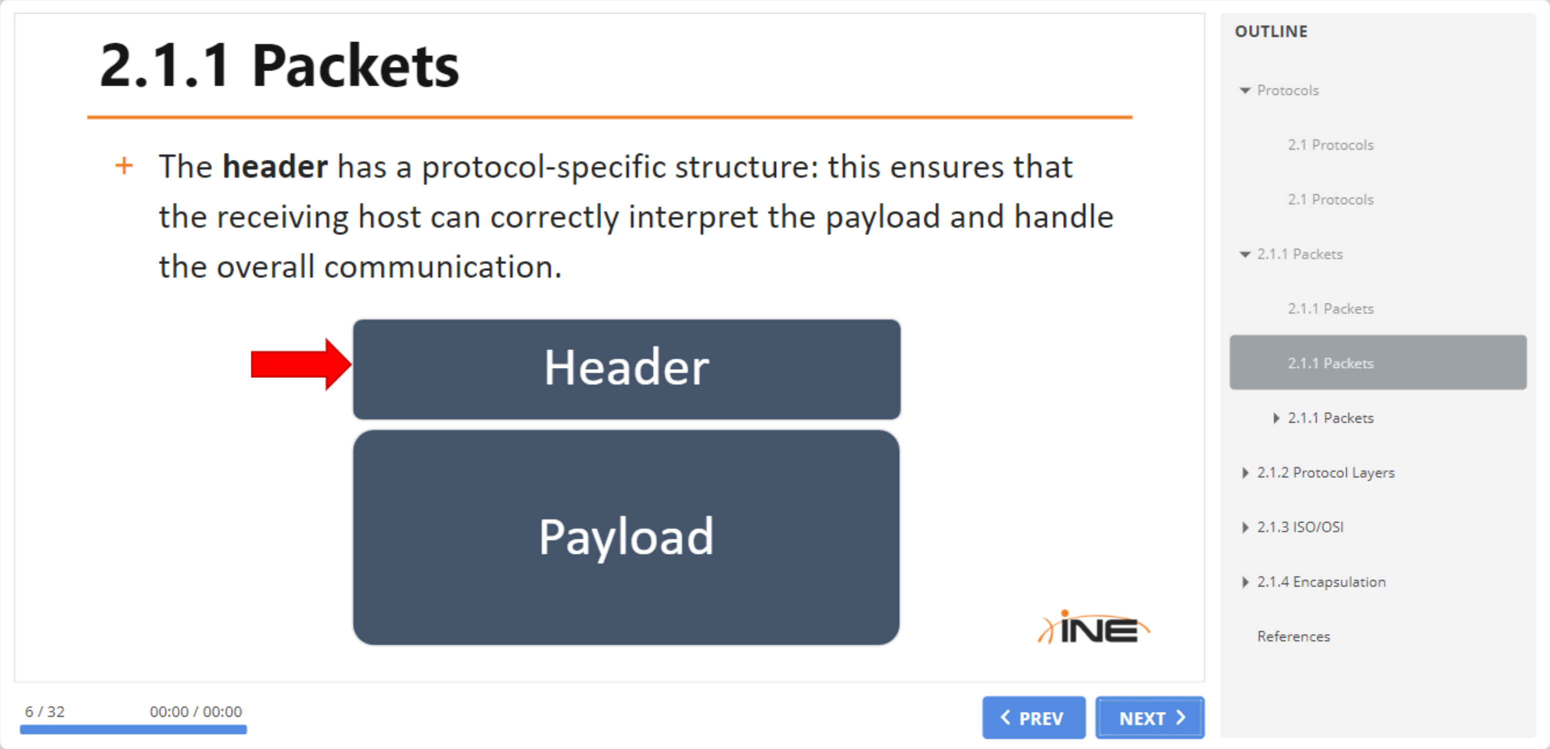


2.1.1 Packets

- Every packet in every protocol has the following structure.

Header

Payload



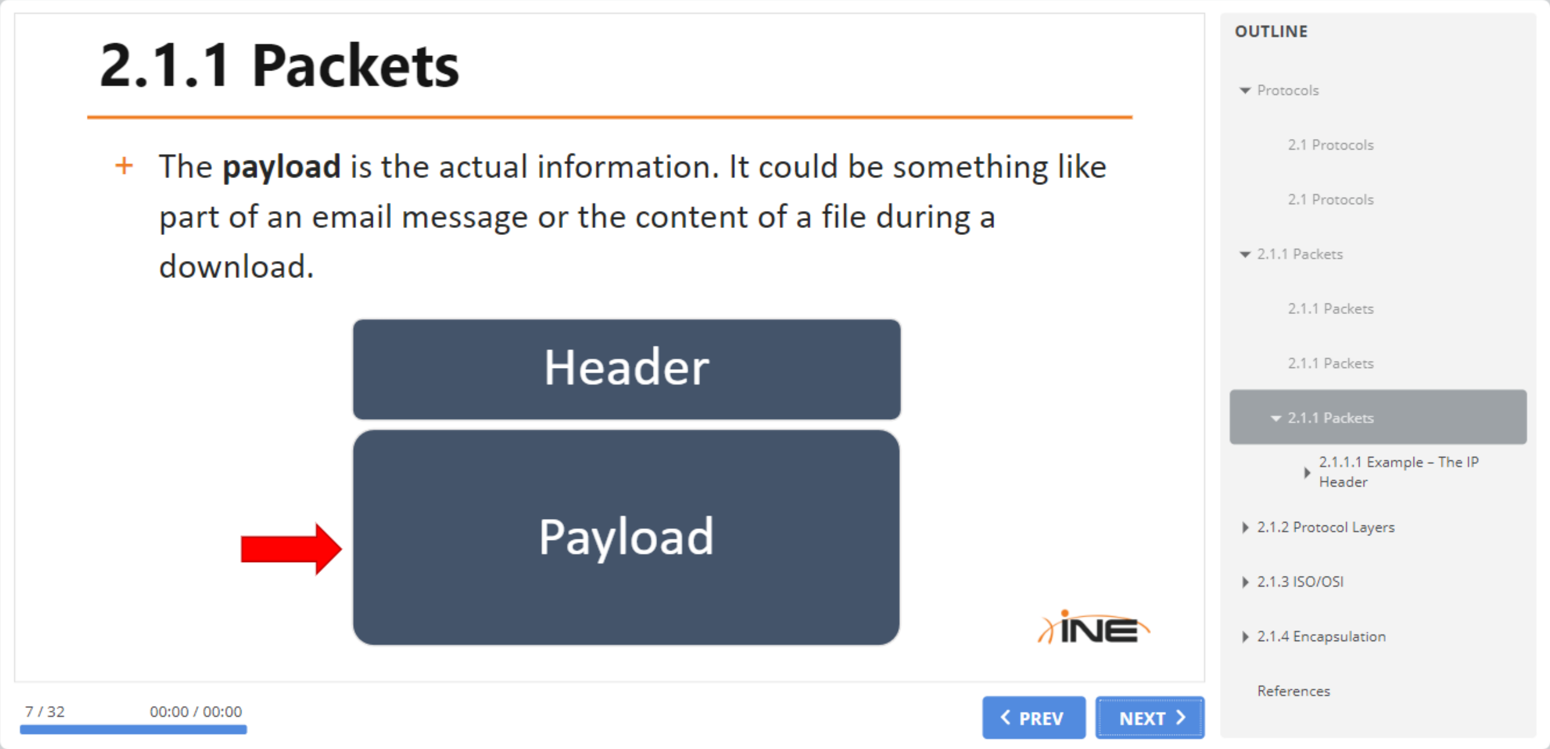
2.1.1 Packets

- The **header** has a protocol-specific structure: this ensures that the receiving host can correctly interpret the payload and handle the overall communication.

Header

Payload

OUTLINE
Protocols
2.1 Protocols
2.1 Protocols
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.2 Protocol Layers
2.1.3 ISO/OSI
2.1.4 Encapsulation
References

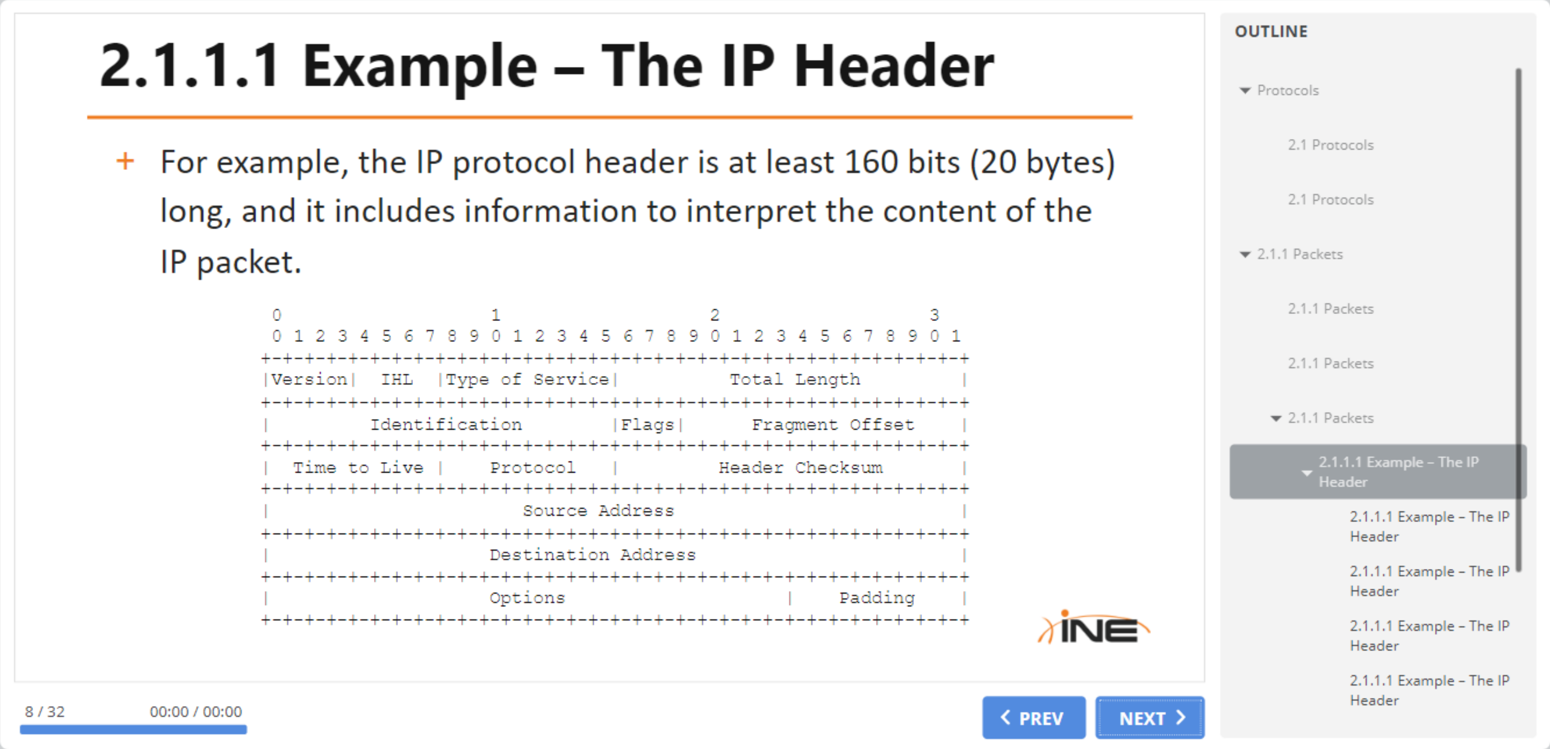


2.1.1 Packets

- The **payload** is the actual information. It could be something like part of an email message or the content of a file during a download.

OUTLINE

Protocols
2.1 Protocols
2.1 Protocols
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.1.1 Example – The IP Header
2.1.2 Protocol Layers
2.1.3 ISO/OSI
2.1.4 Encapsulation
References

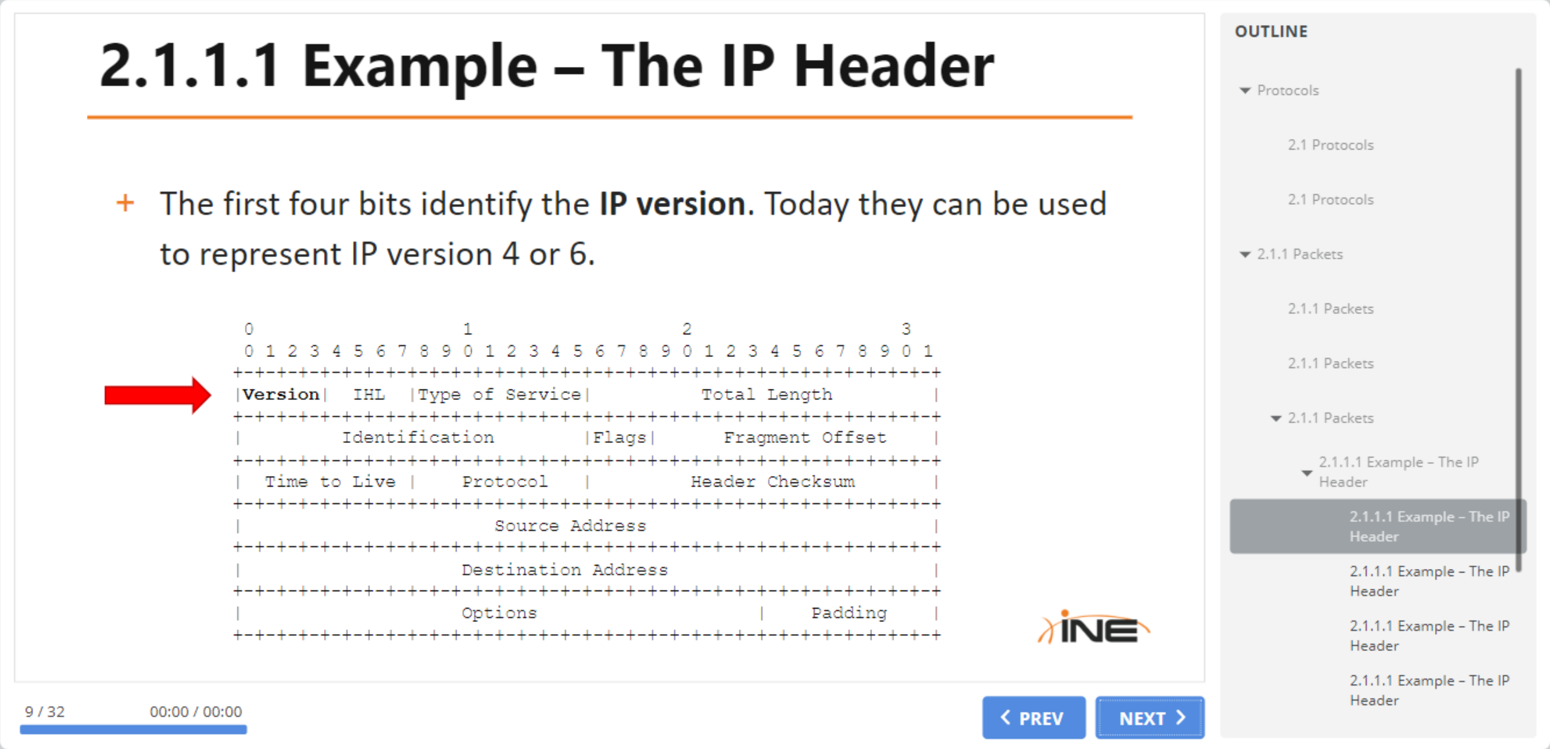


2.1.1.1 Example – The IP Header

+ For example, the IP protocol header is at least 160 bits (20 bytes) long, and it includes information to interpret the content of the IP packet.

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |Version|  IHL  |Type of Service|          Total Length         |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |         Identification        |Flags|      Fragment Offset    |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |  Time to Live |    Protocol   |         Header Checksum       |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                       Source Address                          |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    Destination Address                        |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    Options                    |    Padding    |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Protocols
2.1 Protocols
2.1 Protocols
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header

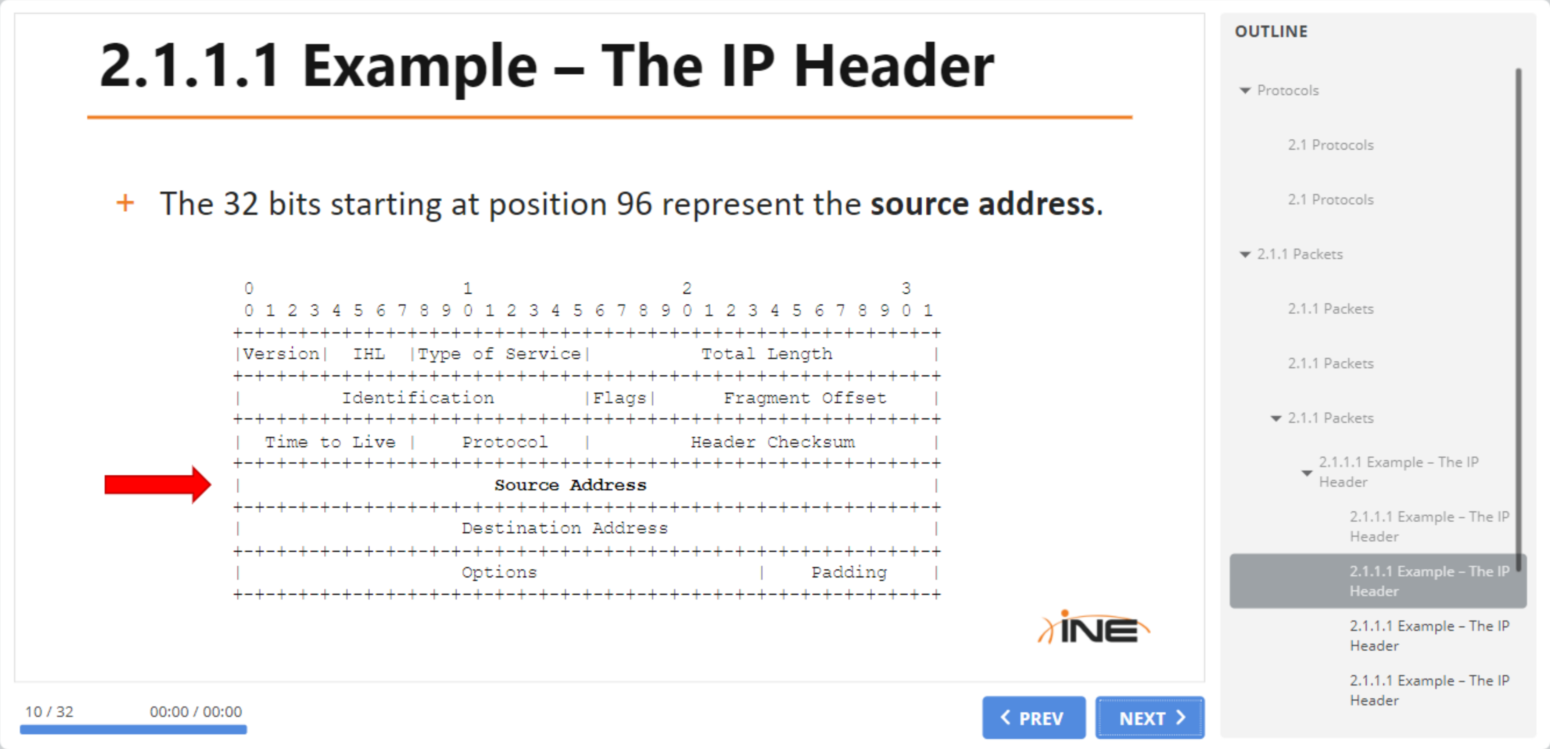


2.1.1.1 Example – The IP Header

- The first four bits identify the **IP version**. Today they can be used to represent IP version 4 or 6.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version|  IHL  |Type of Service|          Total Length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Identification        |Flags|      Fragment Offset    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Time to Live |    Protocol   |         Header Checksum       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Source Address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Destination Address                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Options                    |    Padding    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

INE
9 / 32
00:00 / 00:00
PREV
NEXT
OUTLINE
Protocols
2.1 Protocols
2.1 Protocols
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header

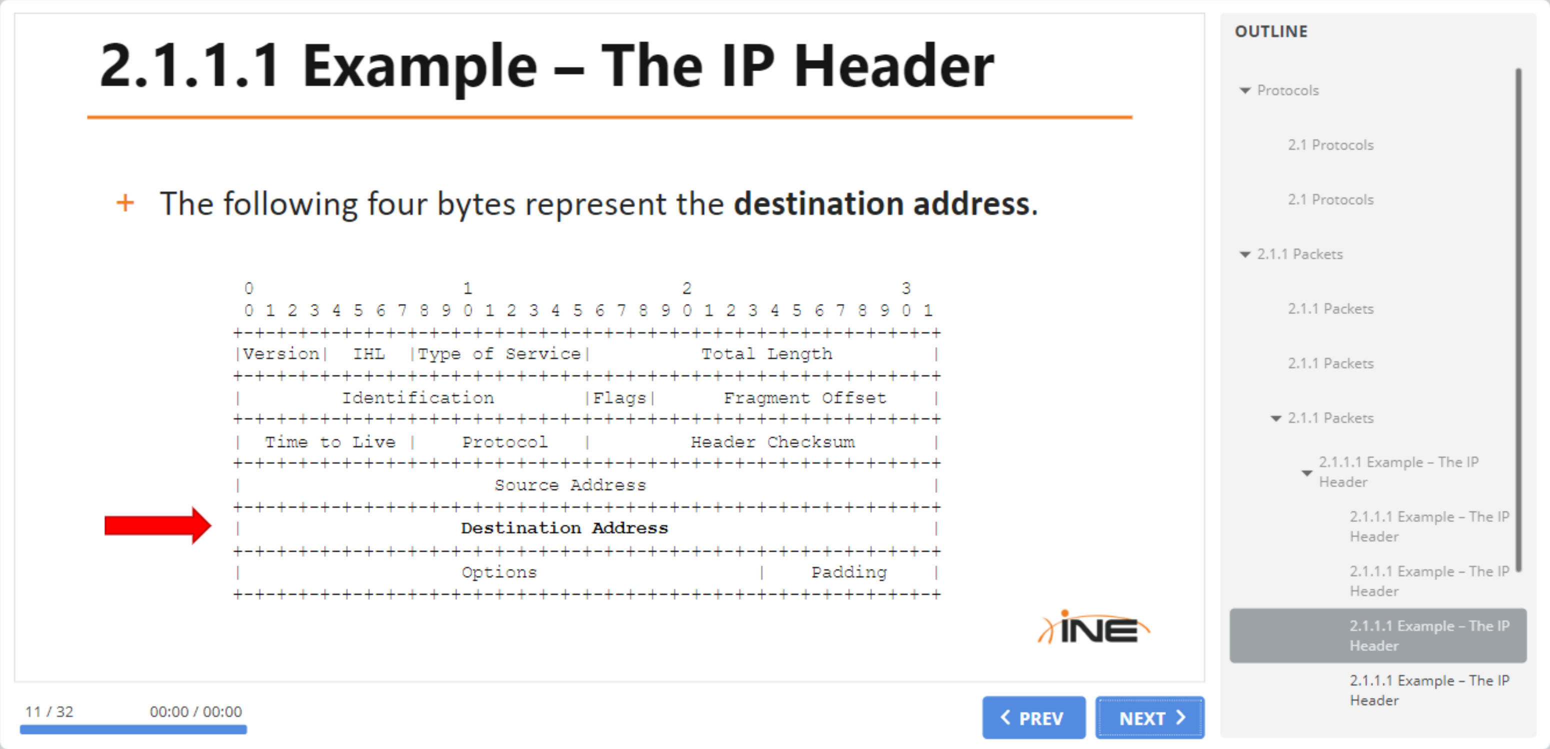


2.1.1.1 Example – The IP Header

+ The 32 bits starting at position 96 represent the **source address.**

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version|  IHL  |Type of Service|          Total Length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Identification        |Flags|      Fragment Offset    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Time to Live |    Protocol   |         Header Checksum       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Source Address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Destination Address                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Options                    |    Padding    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

INE
10 / 32
00:00 / 00:00
PREV
NEXT
OUTLINE
Protocols
2.1 Protocols
2.1 Protocols
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header

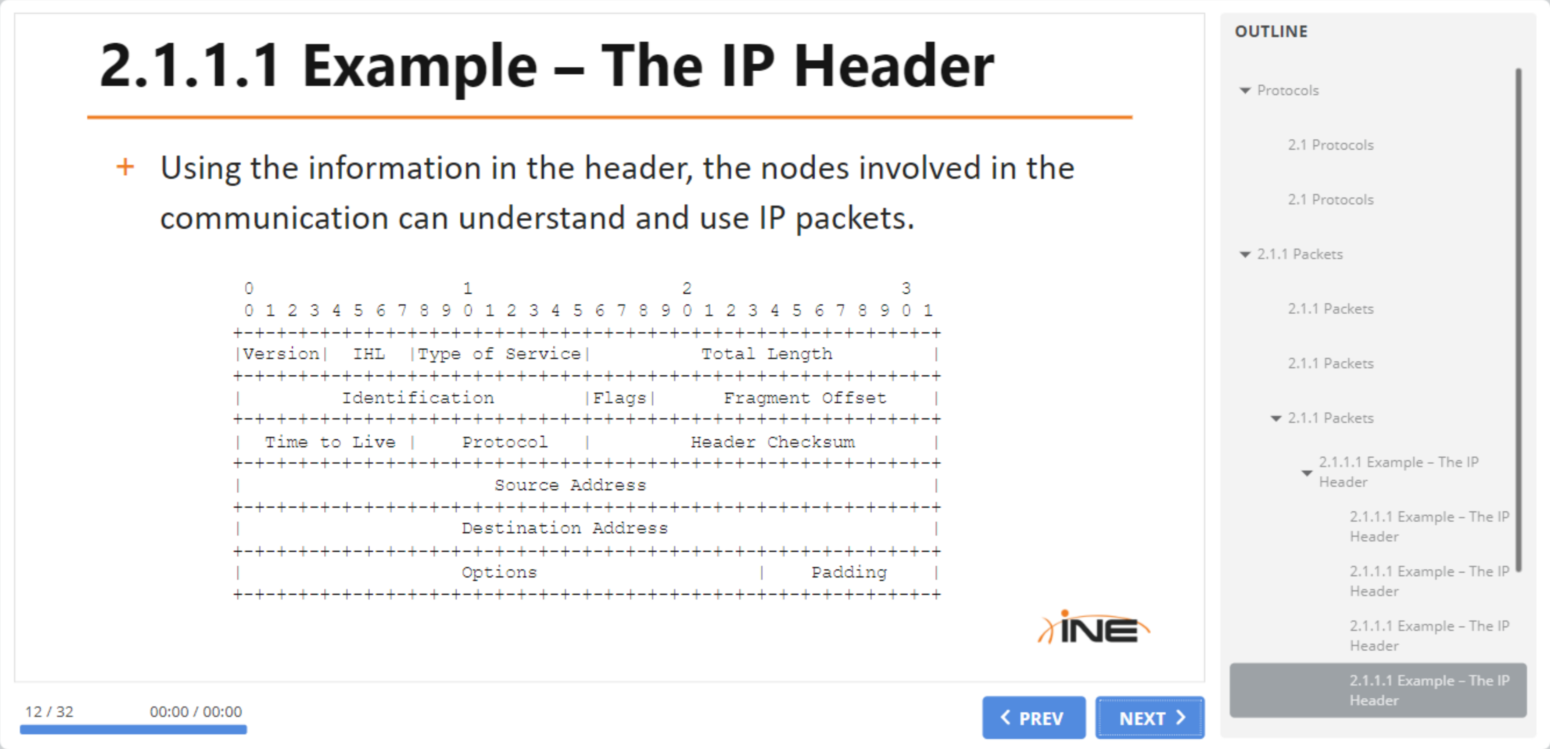


2.1.1.1 Example – The IP Header

- The following four bytes represent the **destination address.**

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version|  IHL  |Type of Service|          Total Length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Identification        |Flags|      Fragment Offset    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Time to Live |    Protocol   |         Header Checksum       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Source Address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Destination Address                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Options                    |    Padding    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

INE
11 / 32
00:00 / 00:00
PREV
NEXT
OUTLINE
Protocols
2.1 Protocols
2.1 Protocols
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header

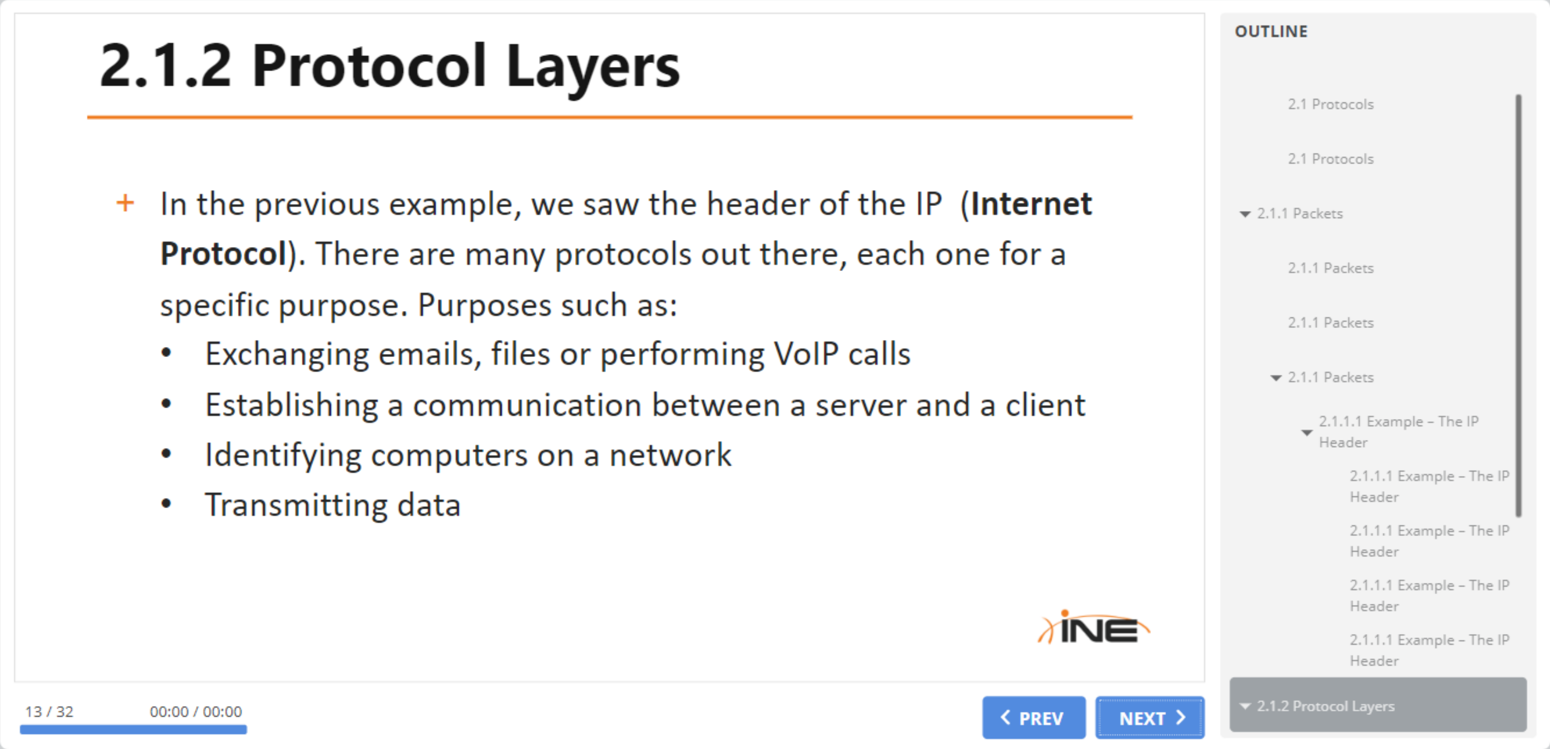


2.1.1.1 Example – The IP Header

- Using the information in the header, the nodes involved in the communication can understand and use IP packets.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version|  IHL  |Type of Service|          Total Length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Identification        |Flags|      Fragment Offset    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Time to Live |    Protocol   |         Header Checksum       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Source Address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Destination Address                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Options                    |    Padding    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

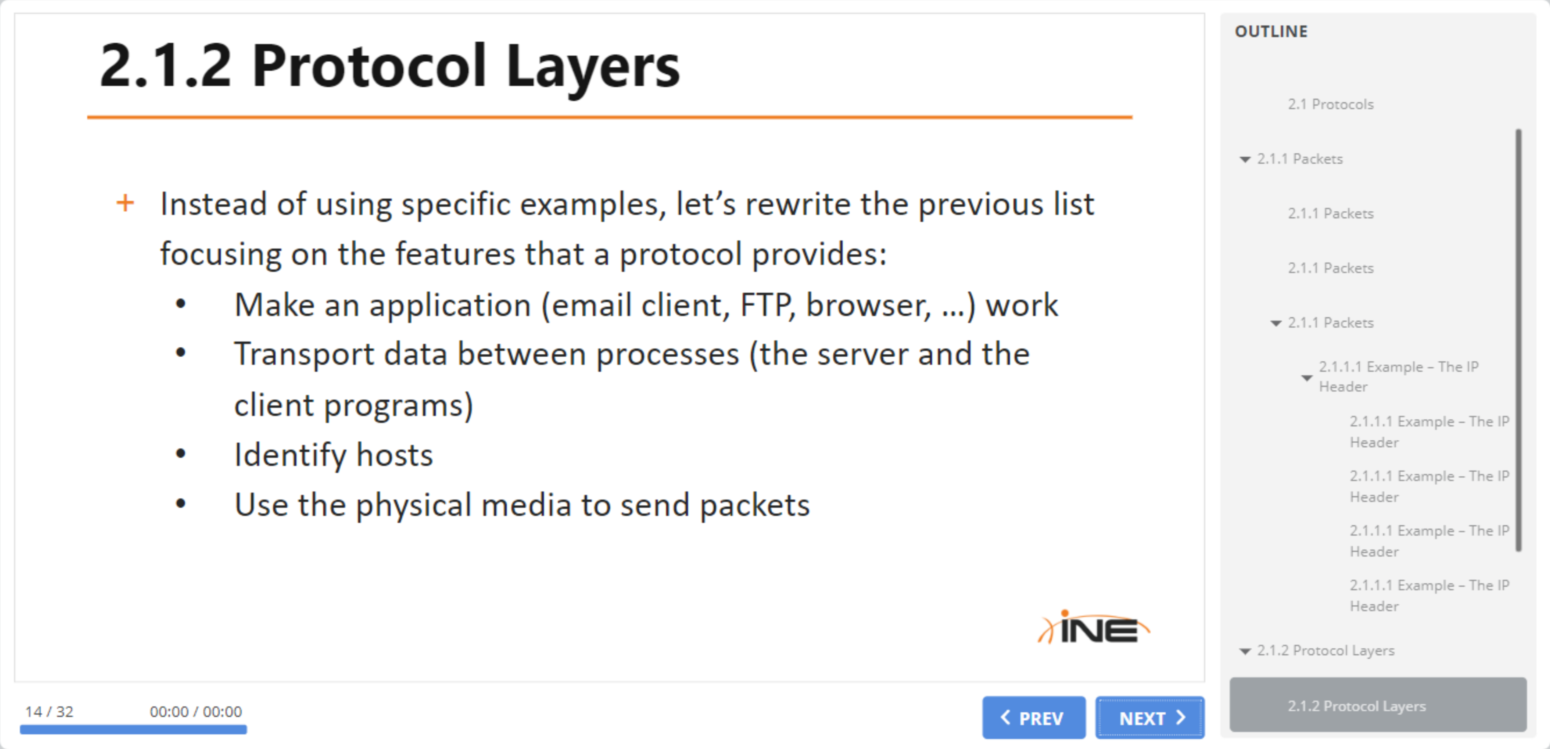
INE
12 / 32
00:00 / 00:00
PREV
NEXT
OUTLINE
Protocols
2.1 Protocols
2.1 Protocols
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header



2.1.2 Protocol Layers

- In the previous example, we saw the header of the IP (**Internet Protocol**). There are many protocols out there, each one for a specific purpose. Purposes such as:
  - Exchanging emails, files or performing VoIP calls
  - Establishing a communication between a server and a client
  - Identifying computers on a network
  - Transmitting data

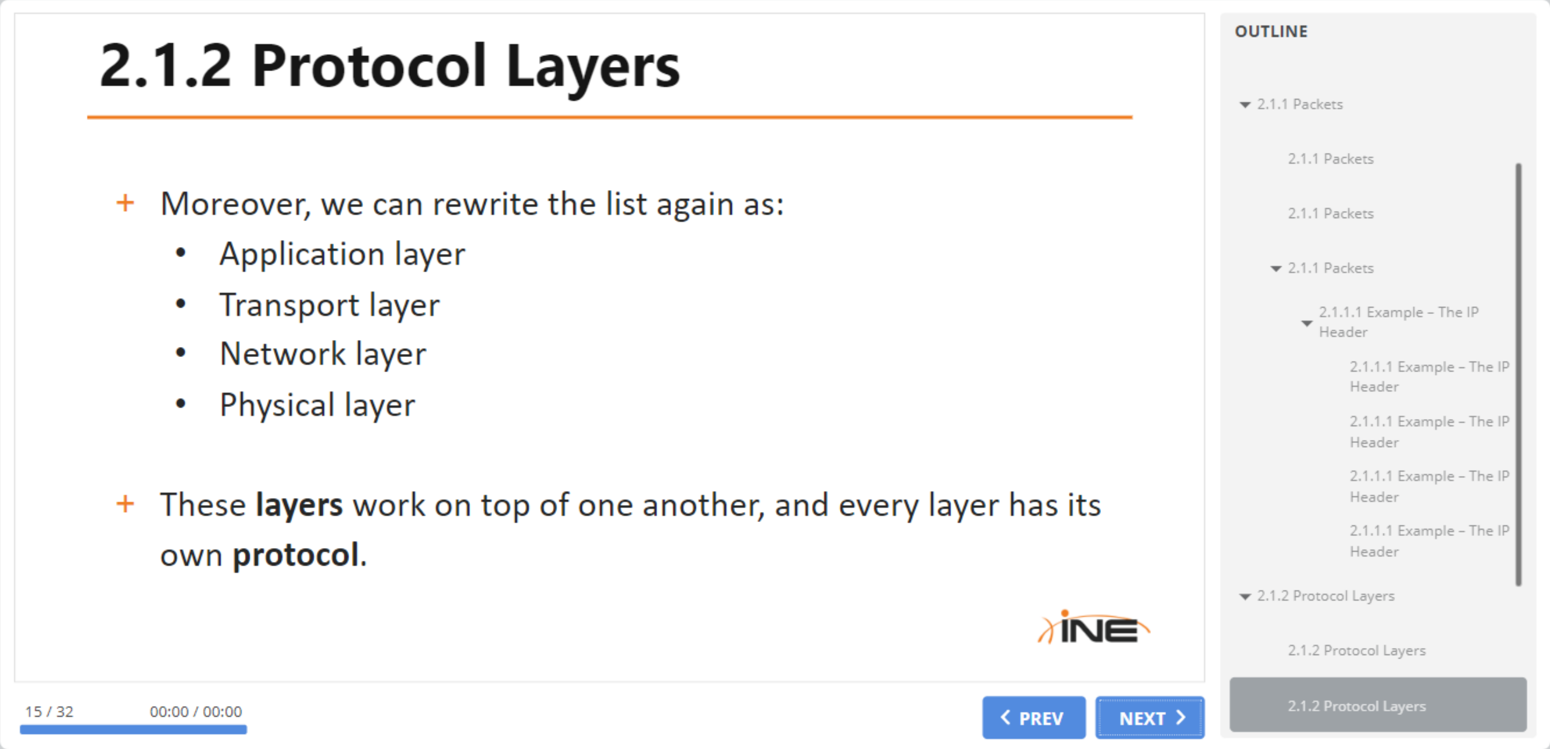
INE
OUTLINE
2.1 Protocols
2.1 Protocols
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.2 Protocol Layers
13 / 32
00:00 / 00:00
PREV
NEXT



2.1.2 Protocol Layers

- Instead of using specific examples, let’s rewrite the previous list focusing on the features that a protocol provides:
  - Make an application (email client, FTP, browser, …) work
  - Transport data between processes (the server and the client programs)
  - Identify hosts
  - Use the physical media to send packets

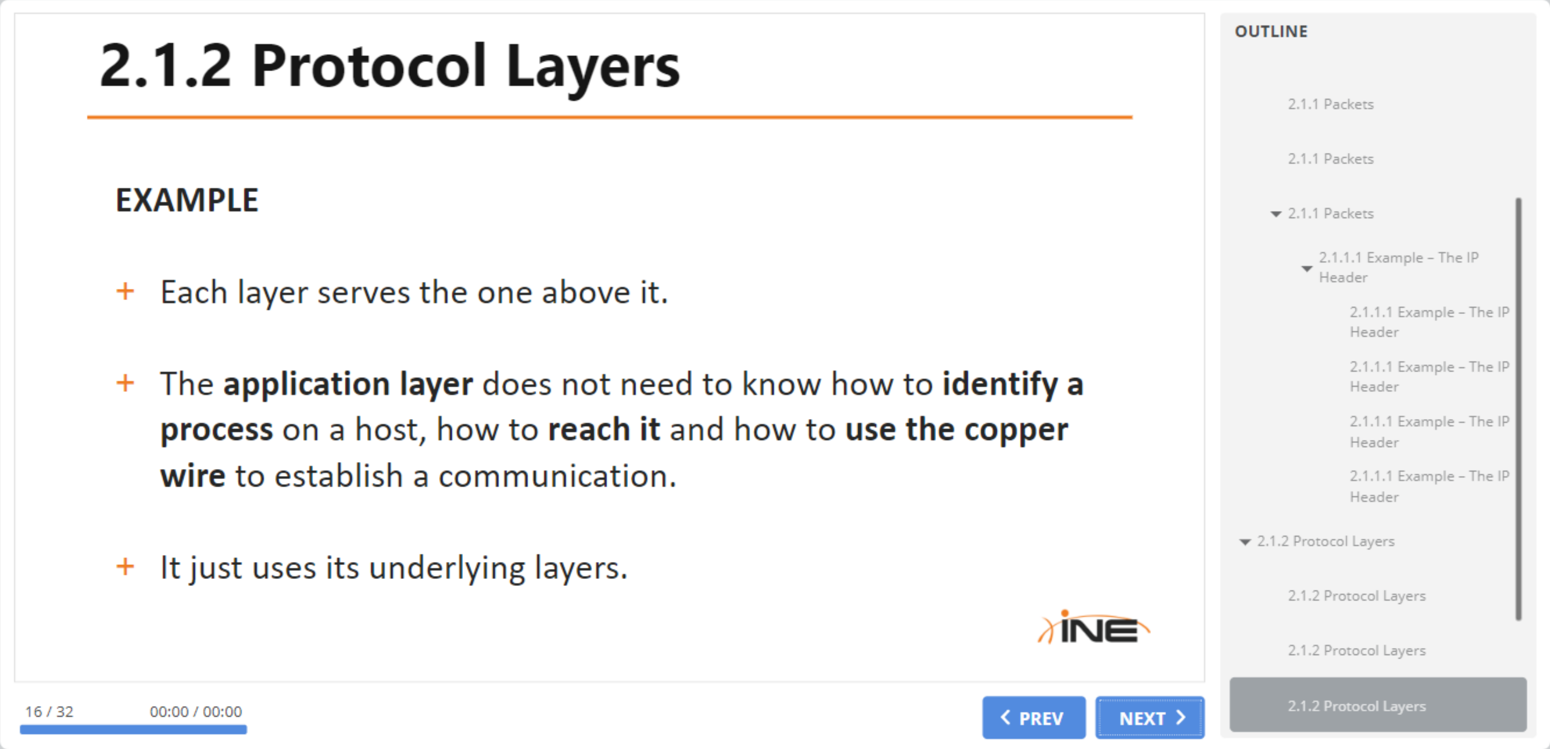
iNE
OUTLINE
2.1 Protocols
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.2 Protocol Layers
2.1.2 Protocol Layers
14 / 32
00:00 / 00:00
PREV
NEXT



2.1.2 Protocol Layers

- Moreover, we can rewrite the list again as:
  - Application layer
  - Transport layer
  - Network layer
  - Physical layer
- These **layers** work on top of one another, and every layer has its own **protocol**.

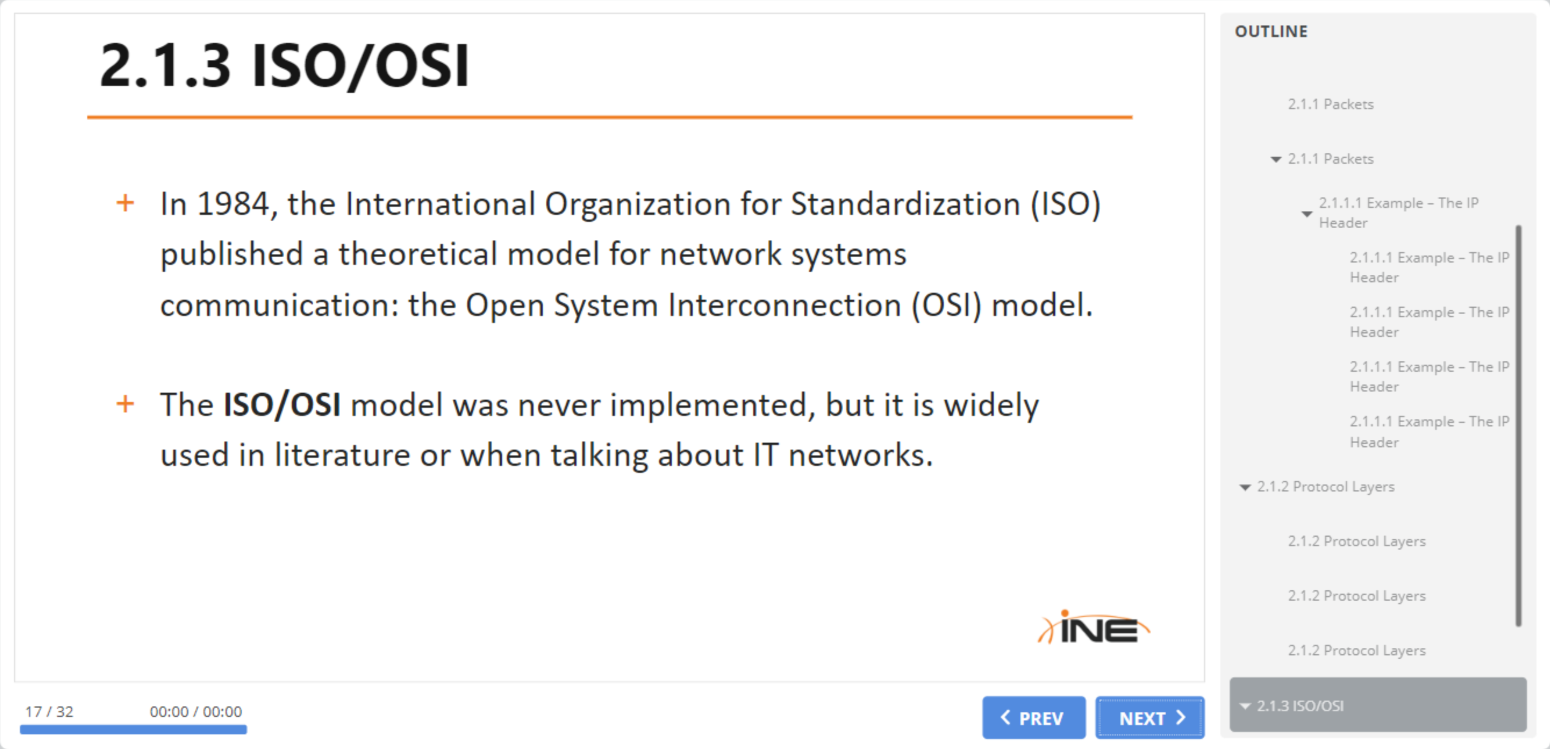
INE
15 / 32
00:00 / 00:00
PREV
NEXT
OUTLINE
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.1 Packets
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.2 Protocol Layers
2.1.2 Protocol Layers
2.1.2 Protocol Layers



2.1.2 Protocol Layers

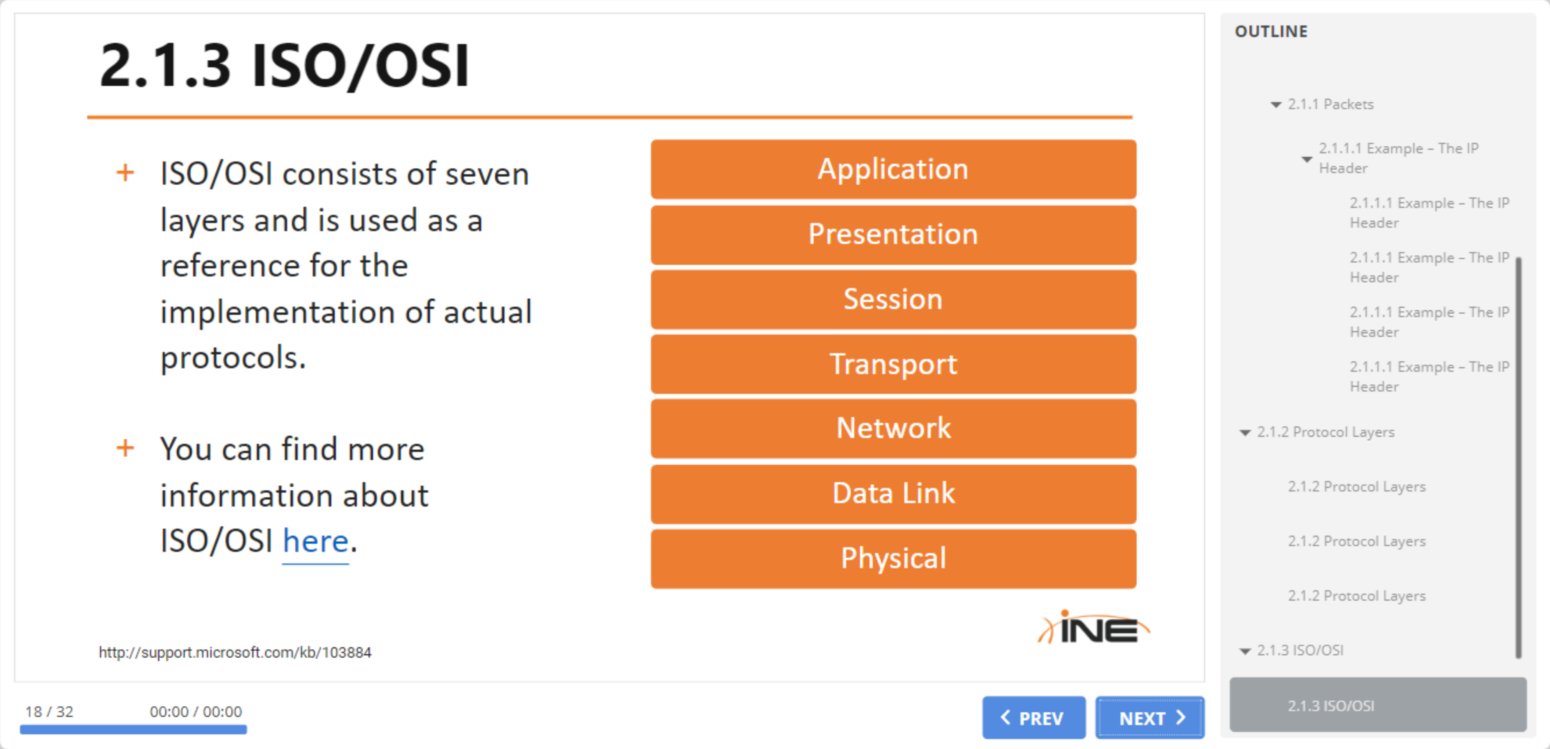
EXAMPLE

- Each layer serves the one above it.
- The **application layer** does not need to know how to **identify a process** on a host, how to **reach it** and how to **use the copper wire** to establish a communication.
- It just uses its underlying layers.



2.1.3 ISO/OSI

- In 1984, the International Organization for Standardization (ISO) published a theoretical model for network systems communication: the Open System Interconnection (OSI) model.
- The **ISO/OSI** model was never implemented, but it is widely used in literature or when talking about IT networks.



2.1.3 ISO/OSI

- ISO/OSI consists of seven layers and is used as a reference for the implementation of actual protocols.
- You can find more information about ISO/OSI here.

Application

Presentation

Session

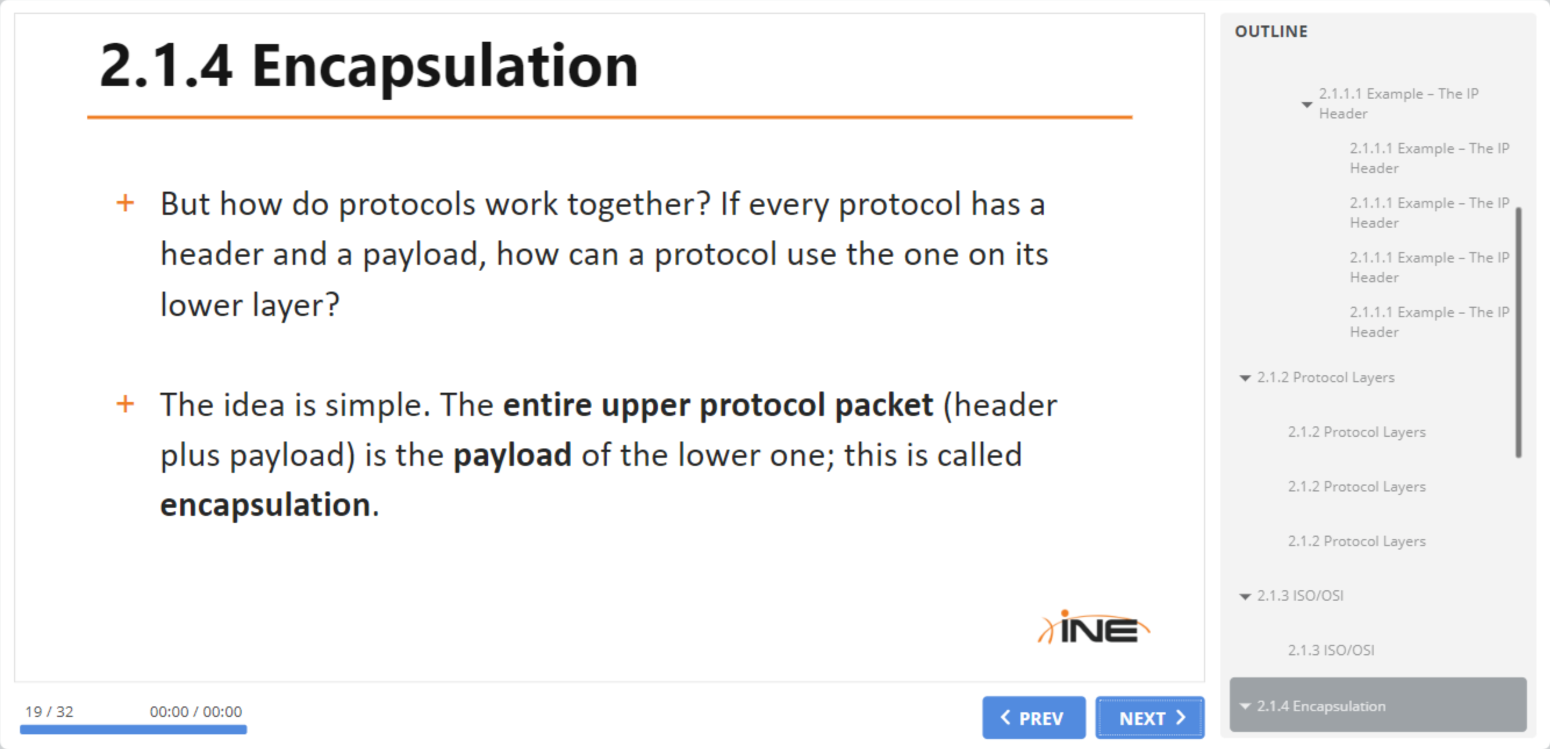
Transport

Network

Data Link

Physical

http://support.microsoft.com/kb/103884

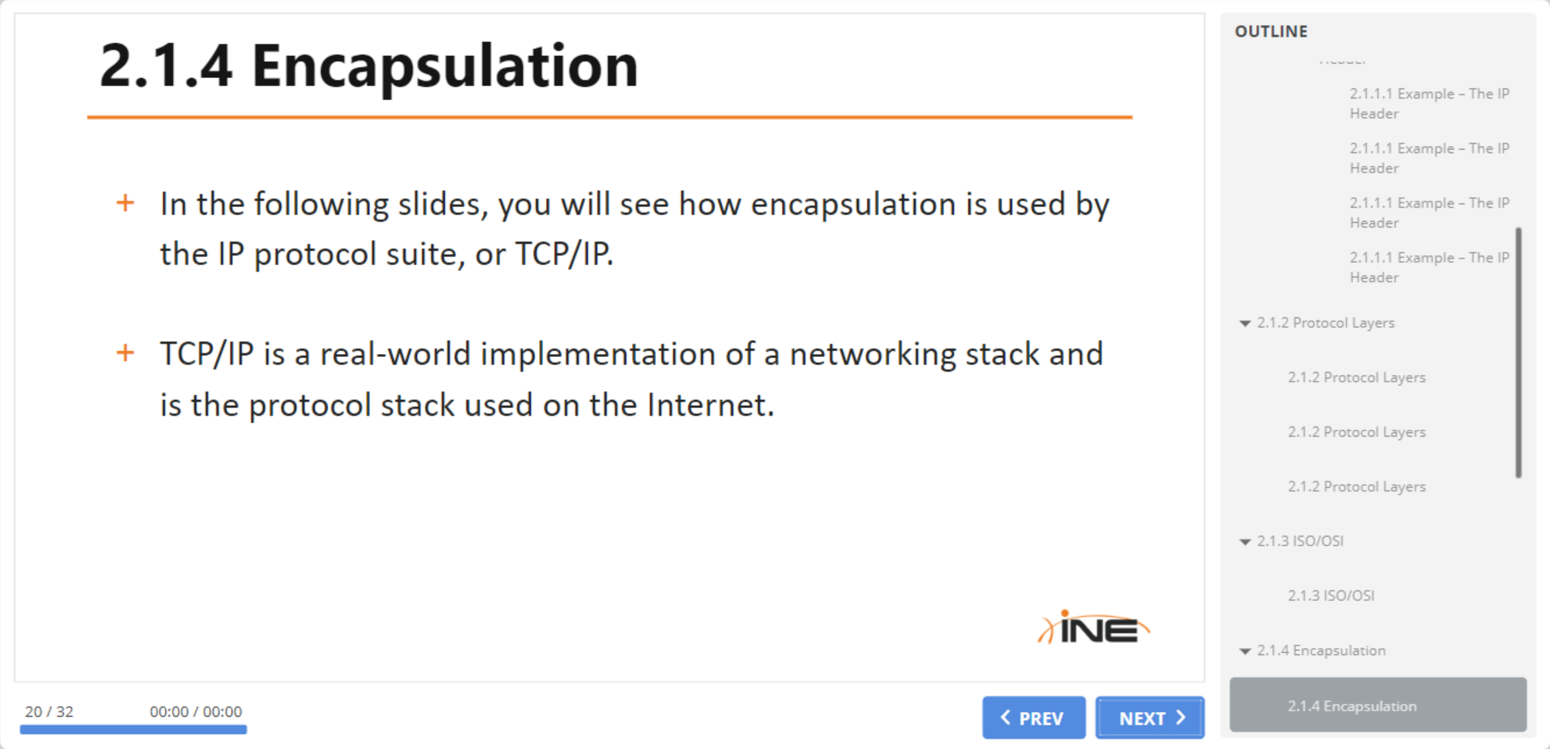


2.1.4 Encapsulation

- But how do protocols work together? If every protocol has a header and a payload, how can a protocol use the one on its lower layer?
- The idea is simple. The **entire upper protocol packet** (header plus payload) is the **payload** of the lower one; this is called **encapsulation**.

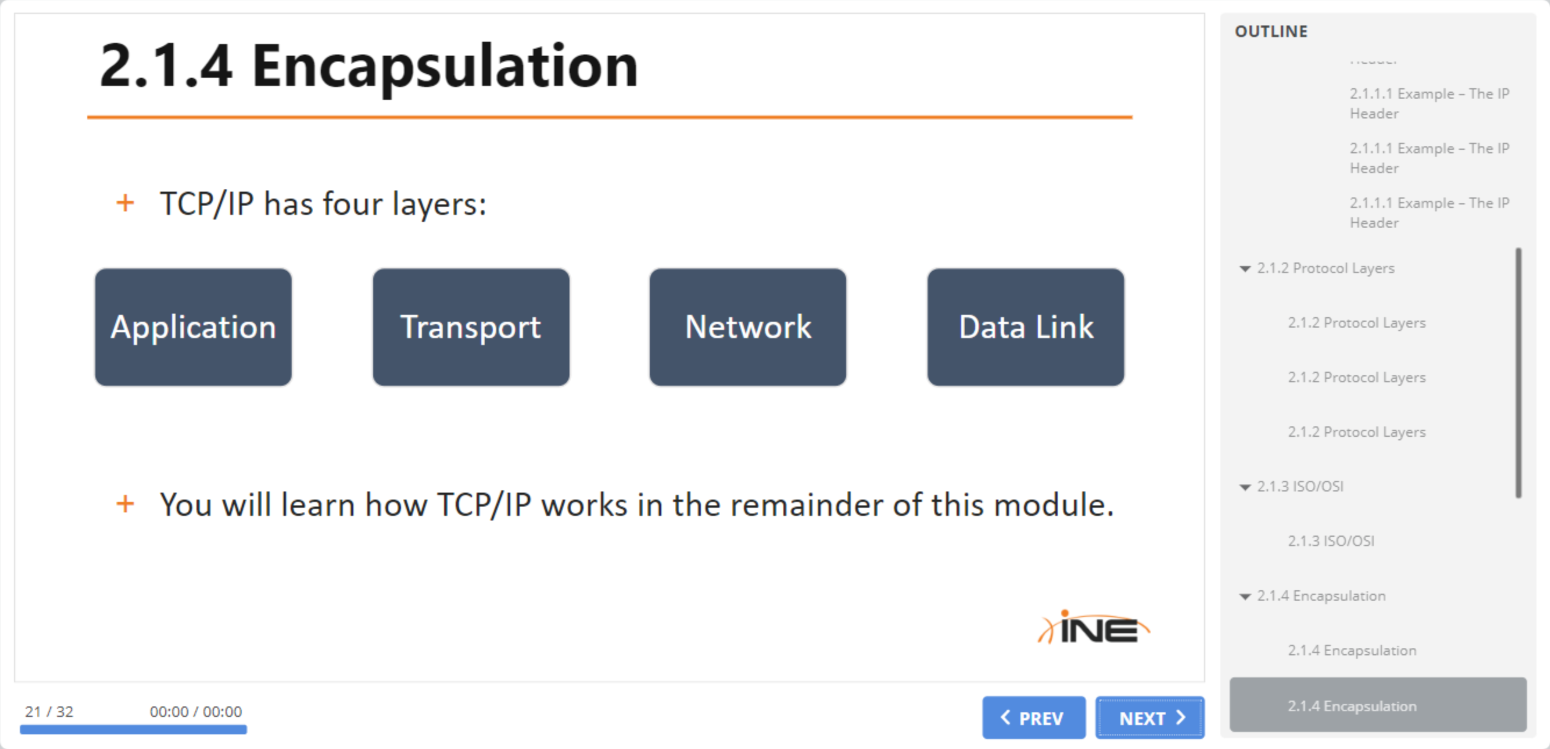
OUTLINE

2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.2 Protocol Layers
2.1.2 Protocol Layers
2.1.2 Protocol Layers
2.1.2 Protocol Layers
2.1.3 ISO/OSI
2.1.3 ISO/OSI
2.1.4 Encapsulation



2.1.4 Encapsulation

- In the following slides, you will see how encapsulation is used by the IP protocol suite, or TCP/IP.
- TCP/IP is a real-world implementation of a networking stack and is the protocol stack used on the Internet.



2.1.4 Encapsulation

- TCP/IP has four layers:

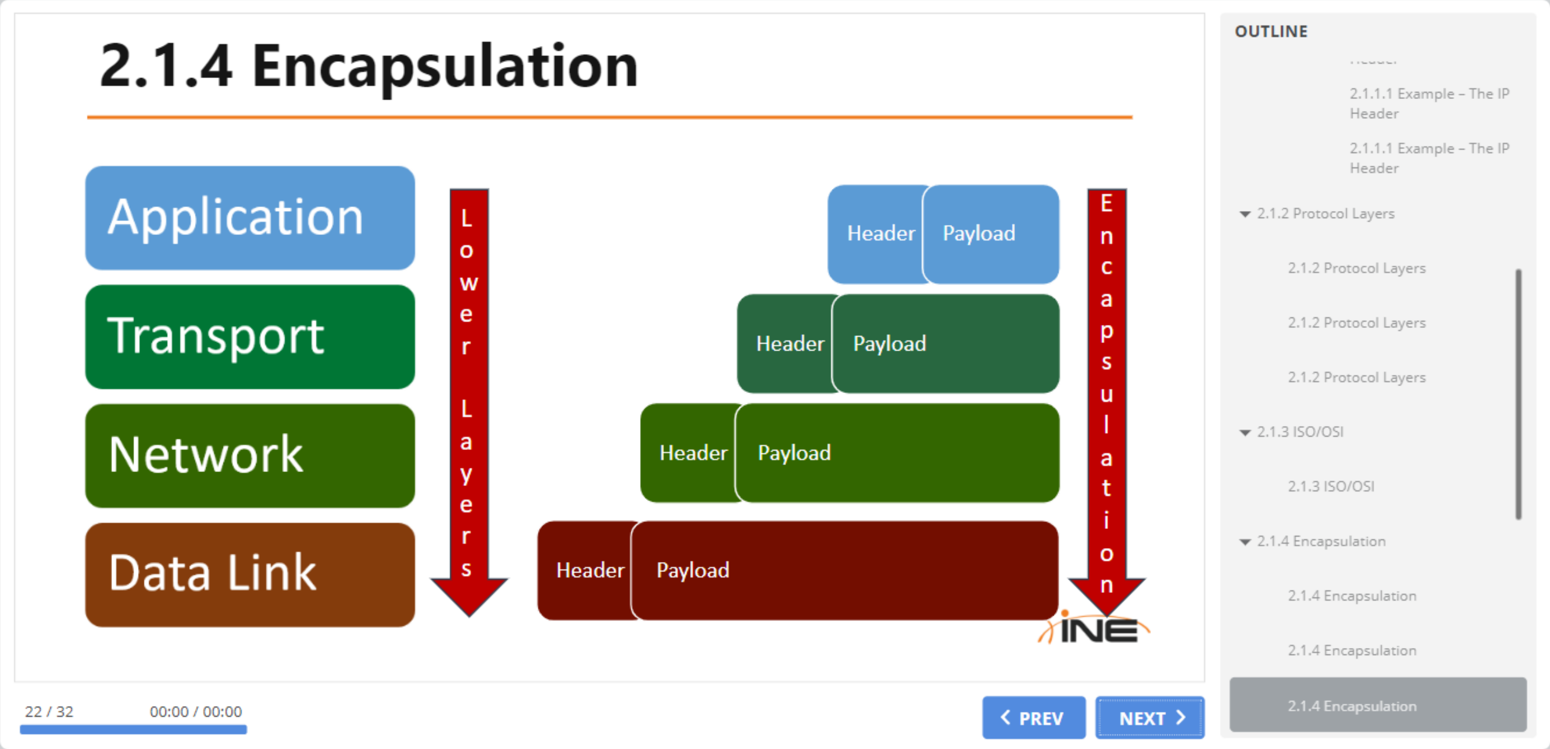
Application

Transport

Network

Data Link

- You will learn how TCP/IP works in the remainder of this module.



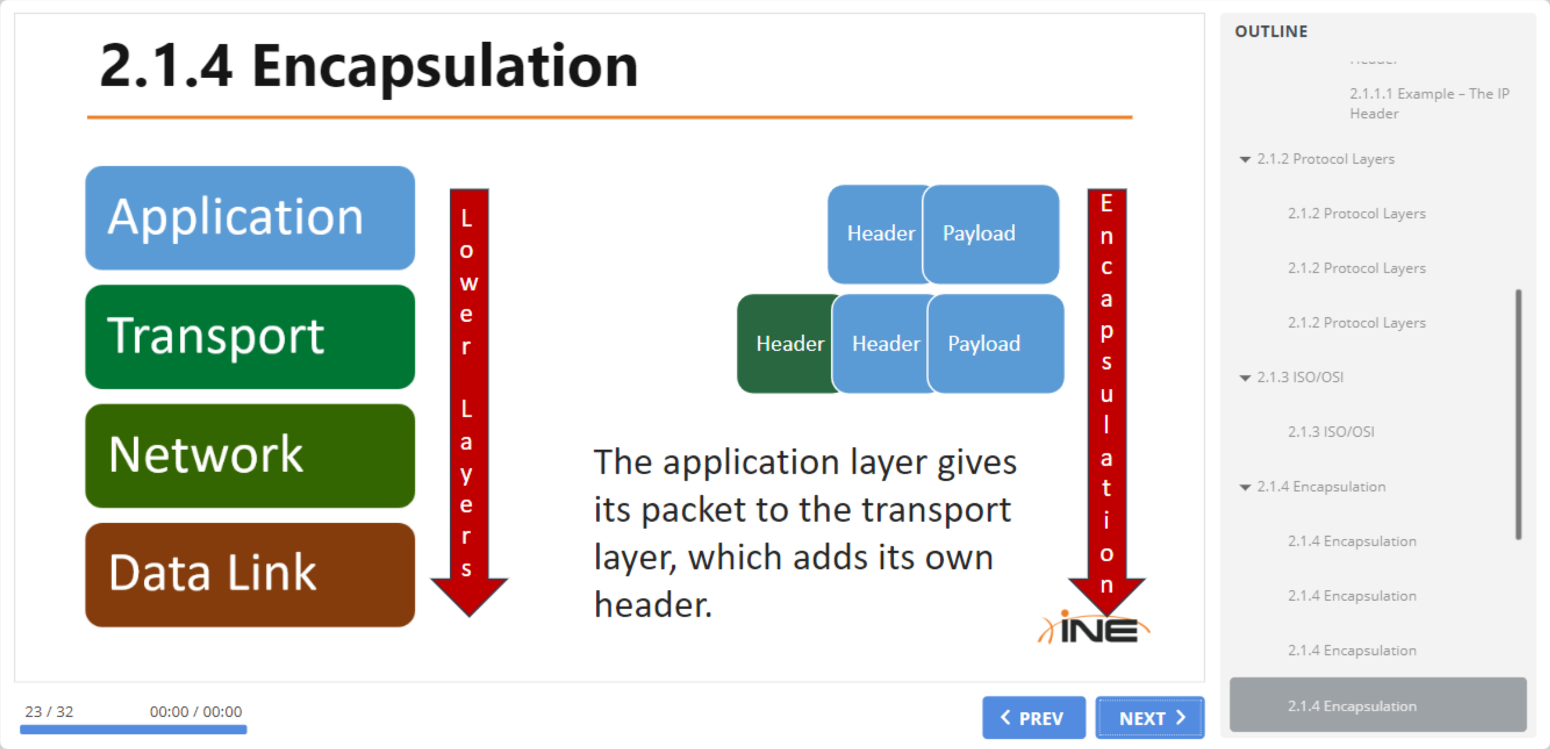
2.1.4 Encapsulation

22 / 32

00:00 / 00:00

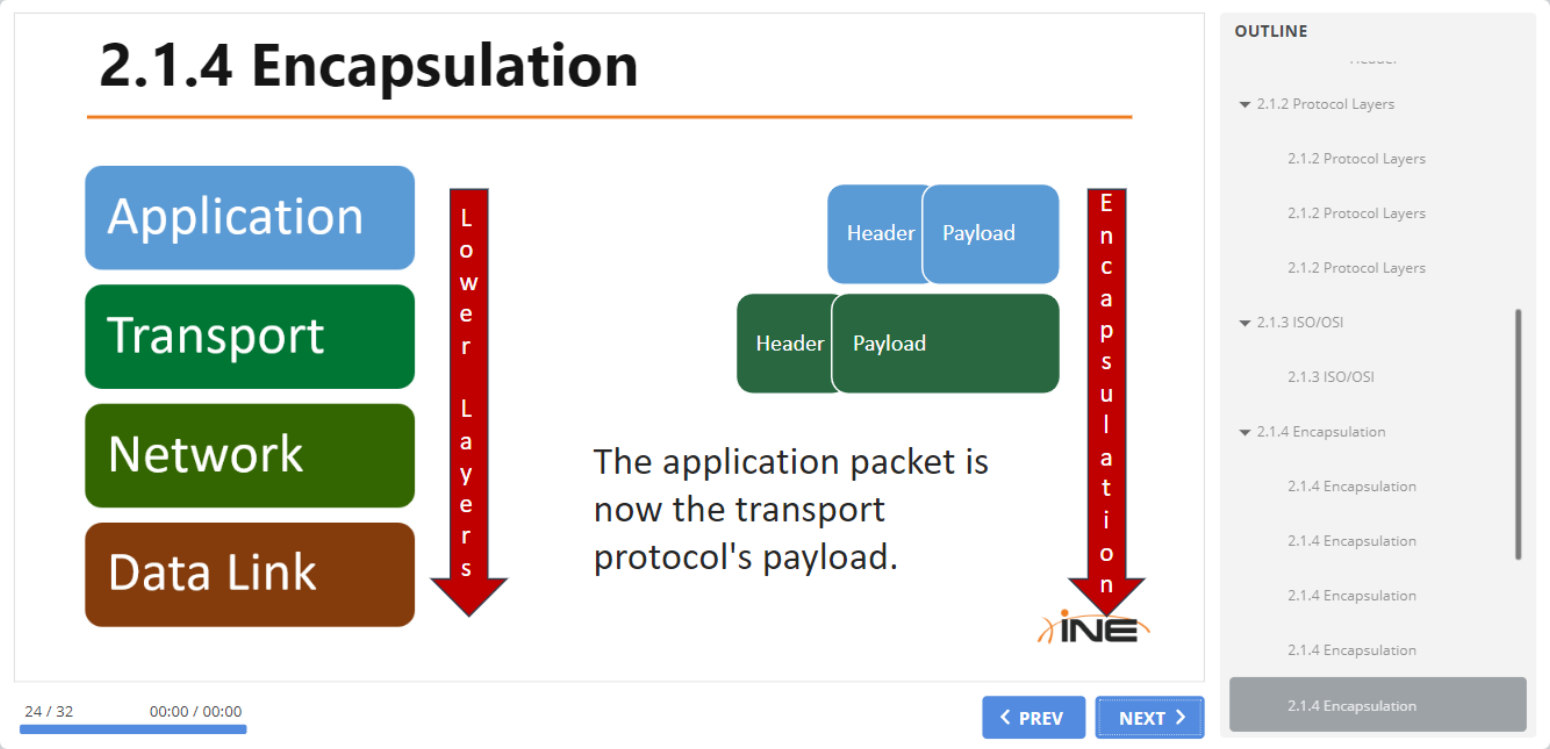
OUTLINE

Header
2.1.1.1 Example – The IP Header
2.1.1.1 Example – The IP Header
2.1.2 Protocol Layers
2.1.2 Protocol Layers
2.1.2 Protocol Layers
2.1.2 Protocol Layers
2.1.3 ISO/OSI
2.1.3 ISO/OSI
2.1.4 Encapsulation
2.1.4 Encapsulation
2.1.4 Encapsulation
2.1.4 Encapsulation



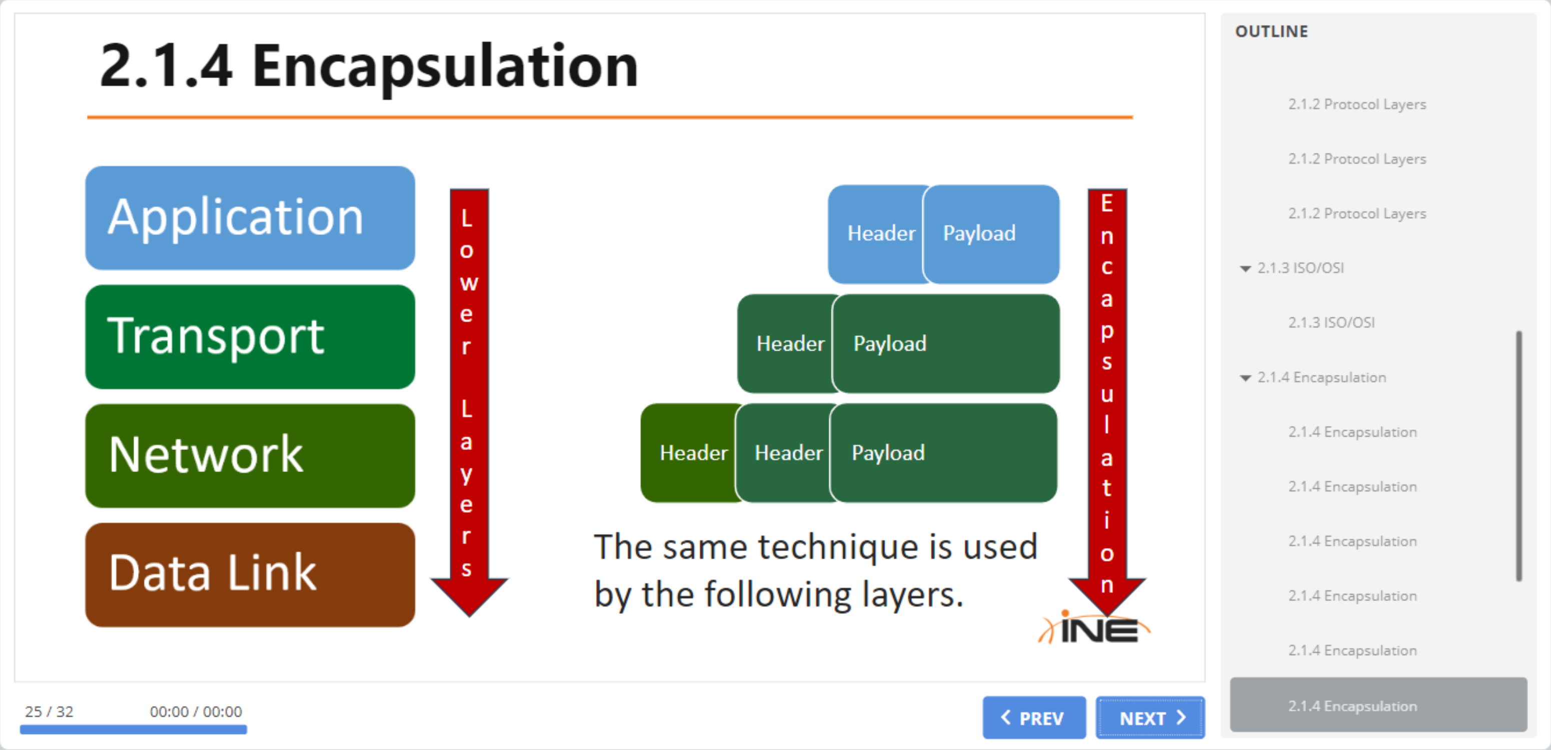
2.1.4 Encapsulation

The application layer gives its packet to the transport layer, which adds its own header.



2.1.4 Encapsulation

The application packet is now the transport protocol's payload.



2.1.4 Encapsulation

Application

Transport

Network

Data Link

Lower Layers

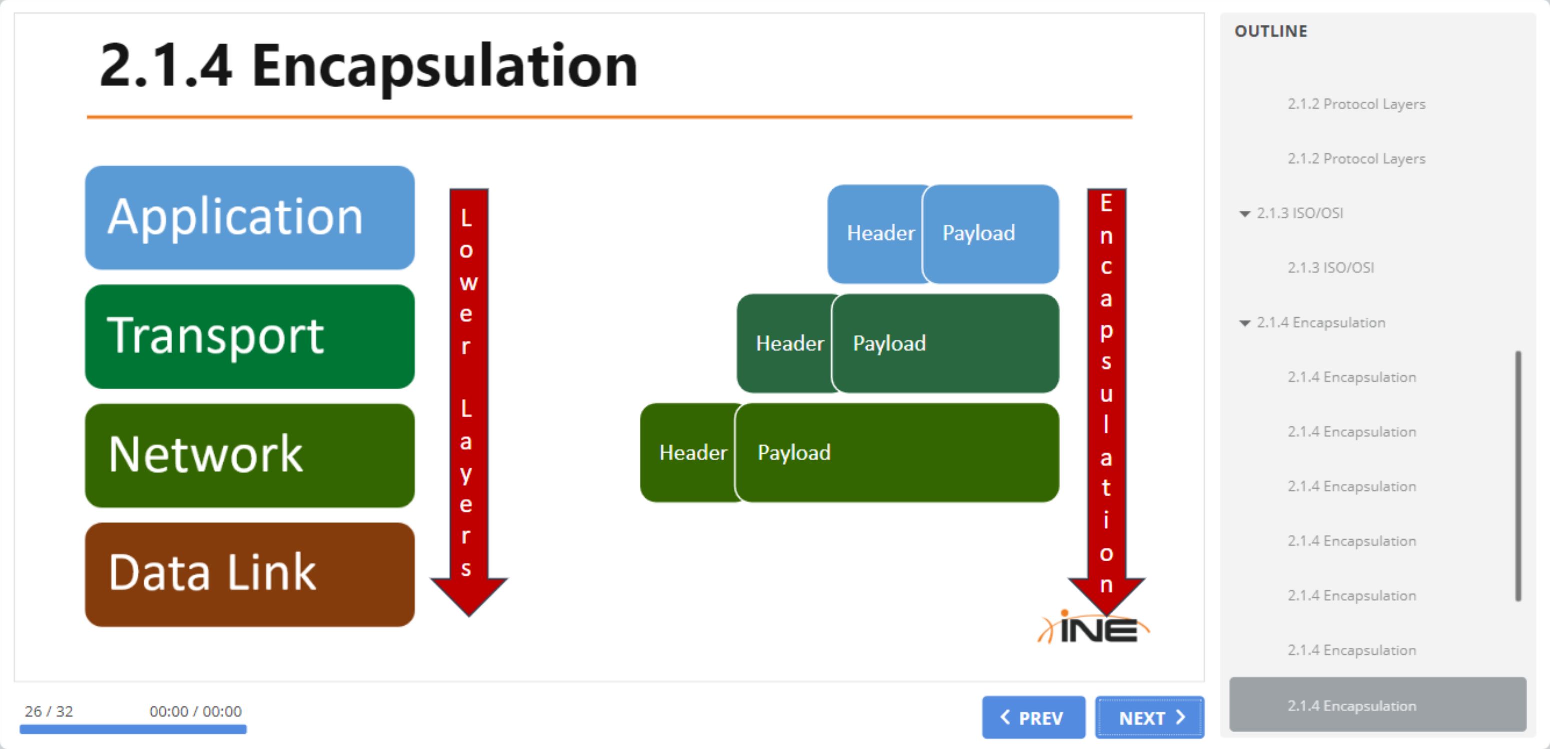
Header | Payload

Header | Payload

Header | Header | Payload

Encapsulation

The same technique is used by the following layers.



2.1.4 Encapsulation

Application

Transport

Network

Data Link

Lower Layers

Header Payload

Header Payload

Header Payload

Encapsulation

OUTLINE

2.1.2 Protocol Layers

2.1.2 Protocol Layers

2.1.3 ISO/OSI

2.1.3 ISO/OSI

2.1.4 Encapsulation

2.1.4 Encapsulation

2.1.4 Encapsulation

2.1.4 Encapsulation

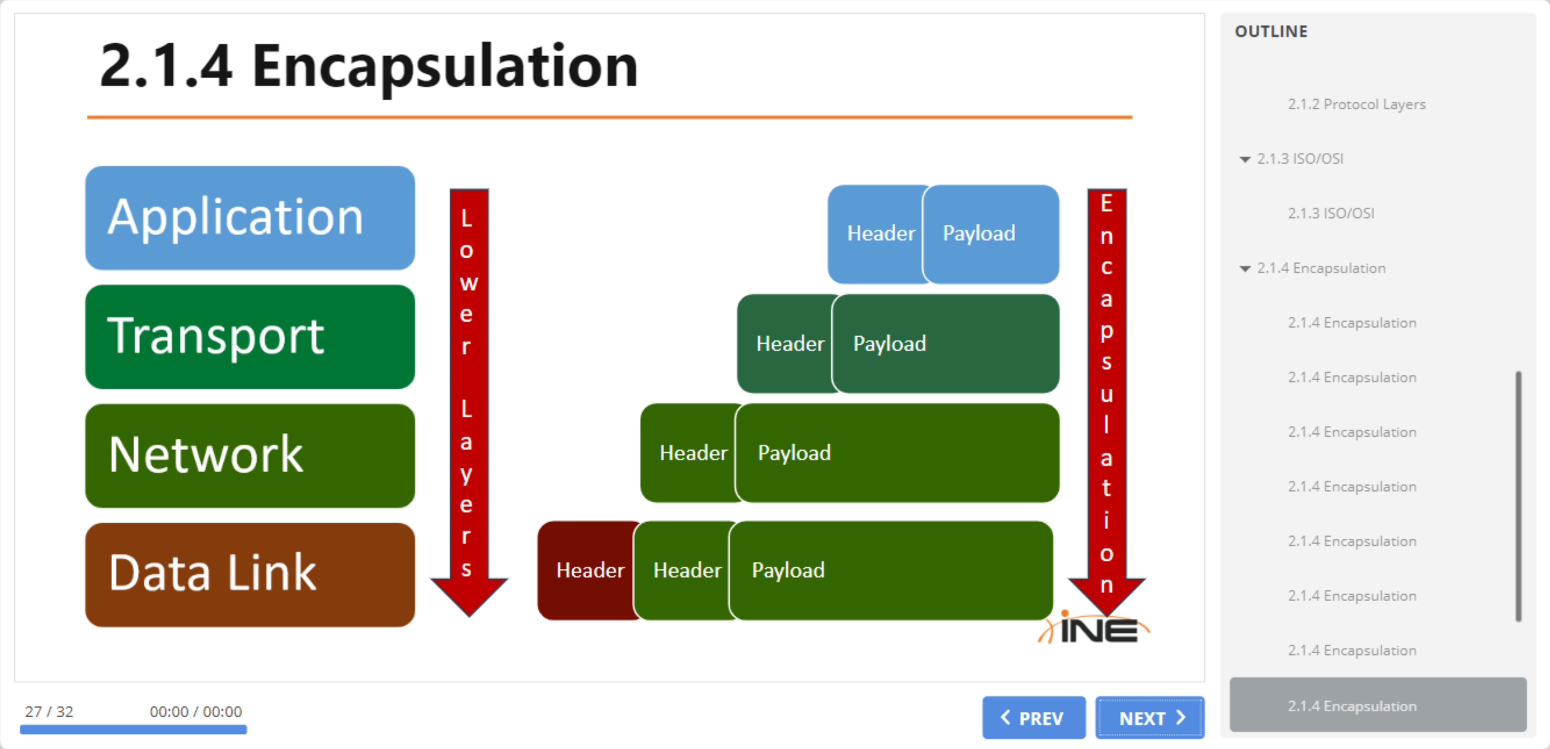
2.1.4 Encapsulation

2.1.4 Encapsulation

2.1.4 Encapsulation

2.1.4 Encapsulation

2.1.4 Encapsulation



Application

Transport

Network

Data Link

Lower Layers

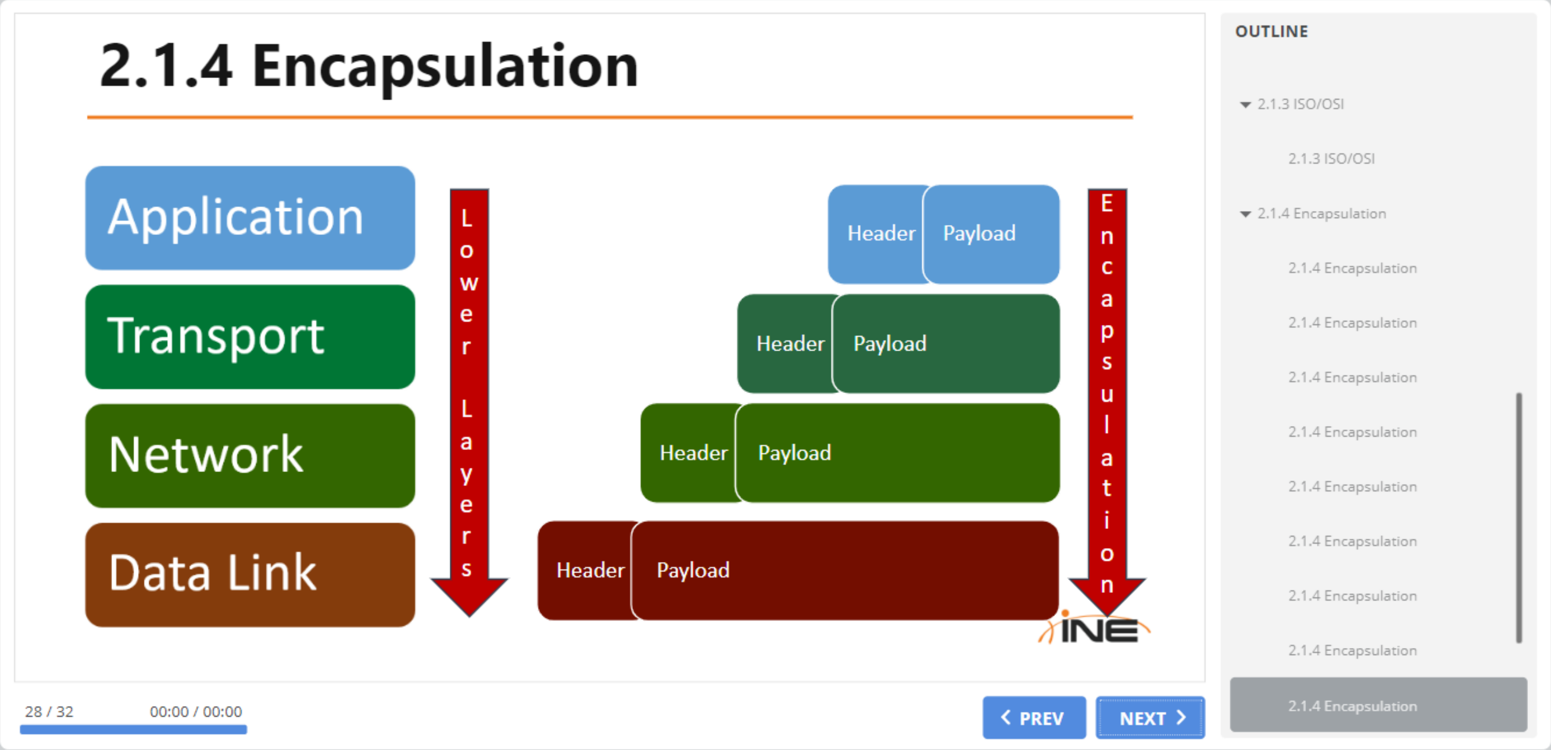
Header Payload

Header Payload

Header Payload

Header Header Payload

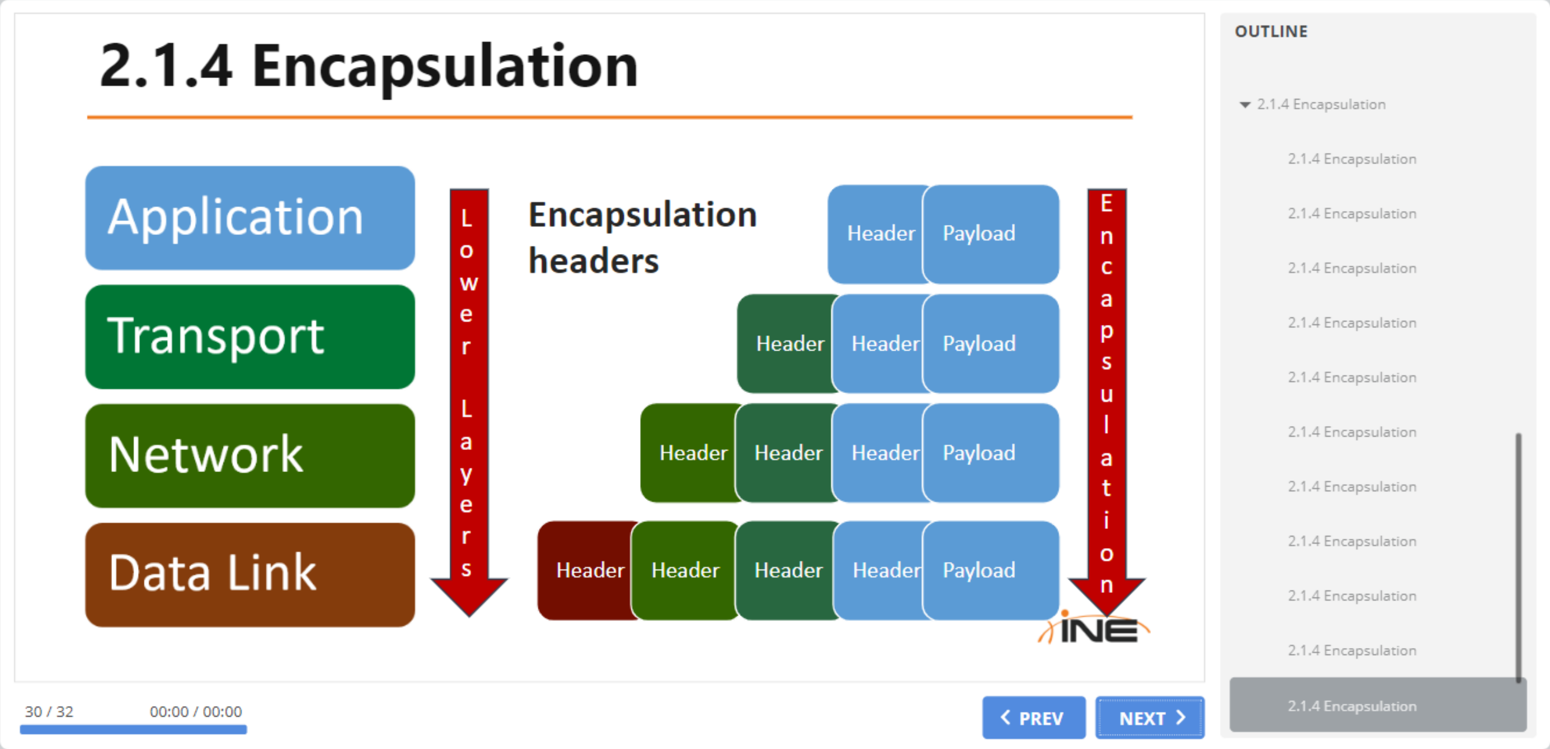
Encapsulation



2.1.4 Encapsulation

2.1.4 Encapsulation

- During encapsulation every protocol adds its own header to the packet, treating it as a payload.
- This happens to **every packet** sent by a host.



2.1.4 Encapsulation

Application

Transport

Network

Data Link

Lower Layers

Encapsulation headers

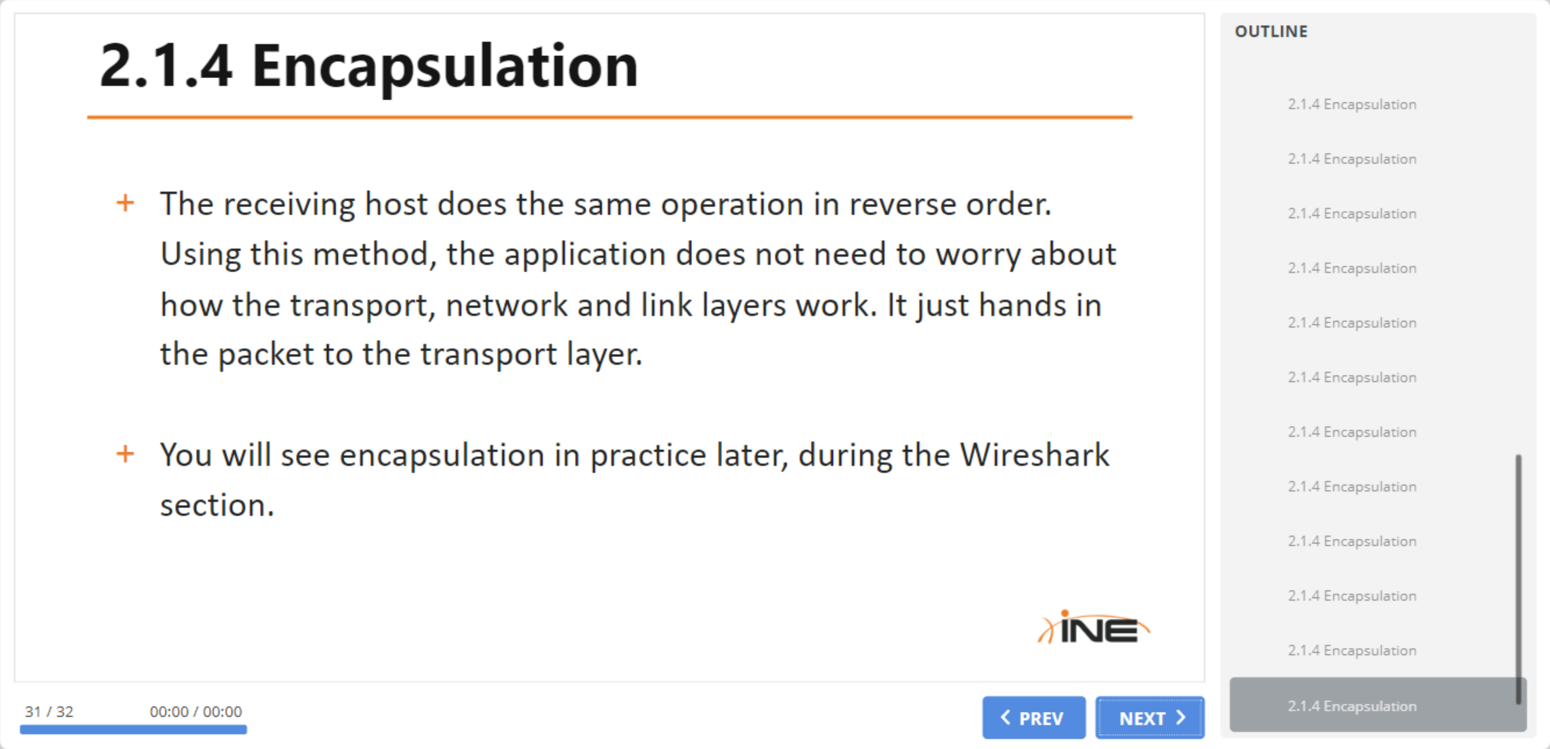
Header Payload

Header Header Payload

Header Header Header Payload

Header Header Header Header Payload

Encapsulation



2.1.4 Encapsulation

- The receiving host does the same operation in reverse order. Using this method, the application does not need to worry about how the transport, network and link layers work. It just hands in the packet to the transport layer.

- You will see encapsulation in practice later, during the Wireshark section.

References

- ISO/OSI Model: http://support.microsoft.com/kb/103884