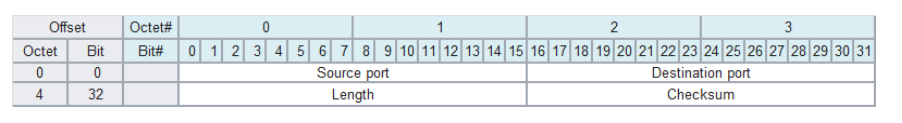
USER DATAGRAM PROTOCOL (UDP)

* Khi chúng ta vận chuyển một chuỗi byte tới tiến trình đích, chường trình UDP sẽ bổ dung them một chuỗi byte nhỏ (8 byte) vào đầu của chuỗi byte dữ liệu. chuỗi byte bổ sung của UDP được gọi là **header.** Phần dữ liệu UDP nhận từ ứng dụng được gòi là phần **payload.**
* UDP header cùng với payload tạo ra một đơn vị thông tin riêng gọi là datagram.
* Cấu trúc của một **UDP header**:



* **Mô tả chuỗi byte**:

Mảng này chia thành 4 phần:

* ***Source port***: chiếm từ 0 – 15 (byte 0 – 1)

+ Là số cổng của tiến trình nguồn

+ Giá trị nằm trong khoảng [0,65535]

* ***Destination port***: chiếm từ 16 đến 31 (byte 2 - 3)

+ Là số cổng tiến trình đích

+ Giá trị nằm trong khoảng [0,65535]

* ***Length*** : chiếm từ 32 đến 47 (byte 4 - 5)

+ Là dộ dài của Header + data

+ Là độ dài của Header + data [8,65 535] tức là (8 byte header + 65527 bye data)

+ Giúp duy trì ranh giới của dữ liệu.

* ***Checksum*** : chiếm từ 48 đến 63 (byte 6 - 7).

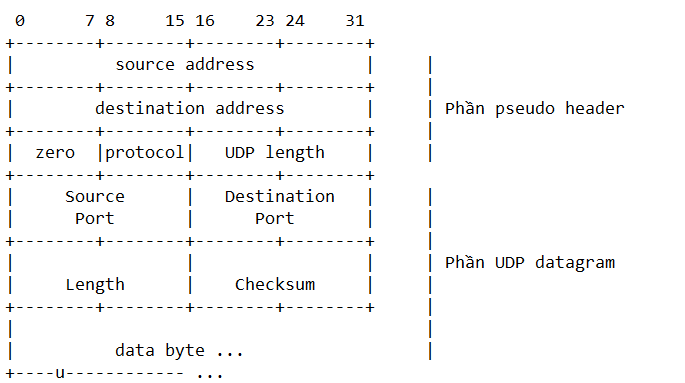
+ Dùng để kiểm tra lỗi header và dữ liệu nhằm đảm tính toàn vẹn của gói tin.

* **Tính toán checksum**:
* Chia original message thành k blocks với n bits trong mỗi block
* Cộng k blocks với nhau
* Cộng phần carry vào sum
* Đảo bit 0-1 1-0 của sum => checksum
* Cộng check sum với tất cả các blocks

+ Nếu kết qua là 0xffff thì kết quả được chấp nhận.

+ Còn lại thì không.

* **Cấu trúc tổng hợp pseudo header và UDP datagram:**



* **Pseudo header**: là một Header giả không được truyền cho người nhận, chỉ đơn giản tham gia vào việc tính toán Checksum.
* UDP length = Length.

VD: Ta có một cấu trúc pseudo Header + UDP datagram như sau:

+ Source address: 152.1.51.27 ( 0x9801 , 0x331b )

+ Destination address: 152.14.94.75 ( 0x980e , 0x5e4b)

+ Zero: 0x00

+ Protocol: 0x11

+UDP length: 0x000a

+ Source Port: 0xa08f

+Destination port: 0x2694

+Length: 0x000a

+ Data 0x6262

* Checksum = 0x14de.