CSE589 PA2 算法

Checksum:

检验和算法可以分成两步来实现。首先在发送端，有以下三步：

1. 把即将发送的IP头部中的检验和设置为0，然后以16位为一个间隔，将IP头部分成许多个16位的字段；
2. 将第1步获得的所有字段进行二进制相加求和；
3. 把最终结果取反，就得到检验和，再将该值填充到IP头部。

其次在接收端，也有相应的三步：

1. 把接收到的IP头部分成16位一个间隔的字段集合；
2. 所有字段进行二进制相加求和；
3. 将最终结果取反，判断该结果是否为0，若为0，则说明检验和正确，若不为0，则协议栈会丢掉这个包。（你没看错，这一步还是要取反）

The sending side of rdt simply accepts data from the upper layer via the rdt\_send(data) event, creates a packet containing the data (via the action make\_pkt(data)) and sends the packet into the channel. In practice, the rdt\_send(data) event would result from a procedure call (for example, to rdt\_send()) by the upper-layer application.

On the receiving side, rdt receives a packet from the underlying channel via the rdt\_rcv(packet) event, removes the data from the packet (via the action extract (packet, data)) and passes the data up to the upper layer (via the action deliver\_data(data)). In practice, the rdt\_rcv(packet) event would result from a procedure call (for example, to rdt\_rcv()) from the lower- layer protocol.

s