TCP Reset Attack on Video Streaming

DRAFT Design Report

Abdullah Al Ishtiaq
Student Id. : 1505080

July 22, 2019

Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology
(BUET)
Dhaka - 1000

Contents

1 Introduction

The Transmission Control Protocol (TCP) is a core protocol of the Internet Protocol Suite. It is one of the 2 main transport layer protocols which sit on top of the IP layer. TCP provides reliable, ordered and error-checked host-to-host communication services for applications. It is considered a stateless protocol suite because each client connection is newly made without considering whether a previous connection had been established or not.
Although TCP is widely used in major internet applications, it introduces a few vulnerabilities too. The most common of these vulnerabilities are: Denial of Service (DOS), Connection Hijacking, TCP Veto, TCP Reset attack etc.

2 TCP Reset Attack on Video Streaming

TCP reset attack on video streaming applications does not target the protocol's typical method of closing a connection which uses a 4-way handshake method. Rather it uses the protocol's method of immediately terminating an unwanted, unexpected or erroneous connection. From the specifications of Transmission Control Protocol (TCP) given in RFC-0793 (September 1981) we quote, "Reset (RST) must be sent whenever a segment arrives which apparently is not intended for the current connection". It is an important property of TCP for ensuring robustness, but at the same time it has opened up a scope of exploitation.
In TCP reset attack on video streaming, an attacker forges a spoofed RST packet that pretends to be one coming from the original video streaming server. As a result the victim immediately closes the TCP connection and goes to CLOSED state. In addition to that, upon receiving additional packets from the original server, the victim itself sends RST packet to the server terminating the connection at the remote end. In this way the attacker can successfully disrupt video streaming on its victim's machine.

3 Topology

In figure 1, the attacker and the victim are in the same subnet and the server is in a remote network. Without the loss of generality there can be more switches or routers in the network, and more hosts connected to the switch. This type of topology is carefully chosen as we need to sniff packets destined to the victim machine to gather information about source IP, TCP port numbers and sequence number in order to successfully perform a TCP reset attack.

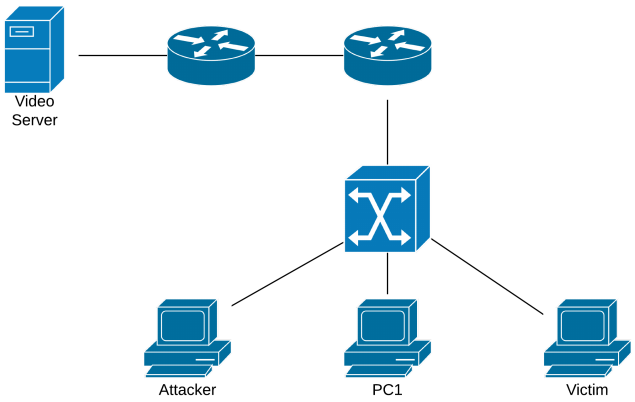

Figure 1: Topology of the Network

4 Timing Diagrams

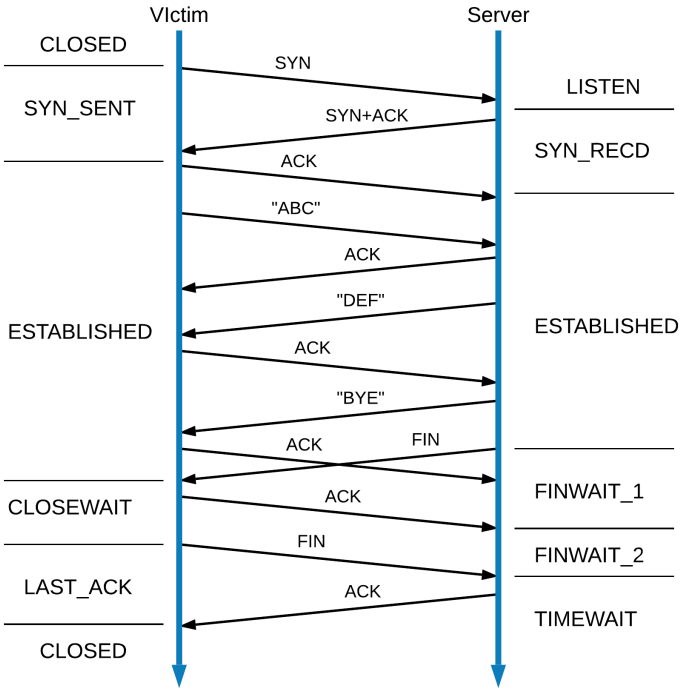

Figure 2: TCP Timing Diagram

In figure 2, the timing diagram of typical communication for video streaming between the streaming server and victim is shown. Here connection can be closed by either of the parties and in both cases a 4-way handshake will occur.
We will perform the TCP reset attack by sending forged RST packet from the attacker machine. The timing diagram of the attack is shown in figure 3. In this case the victim will assume the video server has unexpectedly terminated the TCP connection, and upon receiving additional messages it will send RST back to the server. Finally the server will also close the connection receiving the RST packet.
These timing diagrams are drawn according to the specifications provided in RFC-0793.

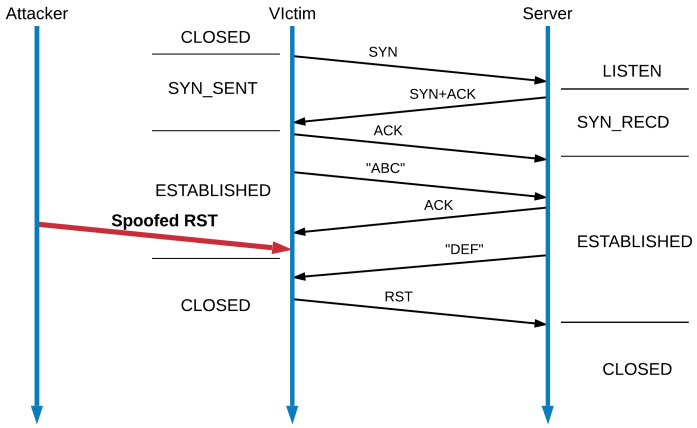

Figure 3: TCP Reset Attack Timing Diagram

5 Strategy

The strategy of our attack can be described in 3 steps. Those are:

1. First we need to find out the IP address of either the victim machine or the video server.

2. Then we have to sniff packets in the network to discover the other IP address, the TCP port numbers and the correct sequence number.

3. Finally we forge a TCP packet with correct IP addresses, TCP Port numbers and sequence number and with RST bit set and send it to the victim machine.

Here one important aspect needs to be discussed. We can also perform the attack by sending forged RST packets to the video streaming server, but

chances are that it will be self harmful as the server may block the attacker's IP address. So instead of doing so, we are going to send the forged packets to the victim machine. Another important note is that we must send the packet with haste as the sequence number in the forged RST packet must be within victim's window to be effective. Otherwise it will be discarded without any impact on the connection.

6 Packet Details

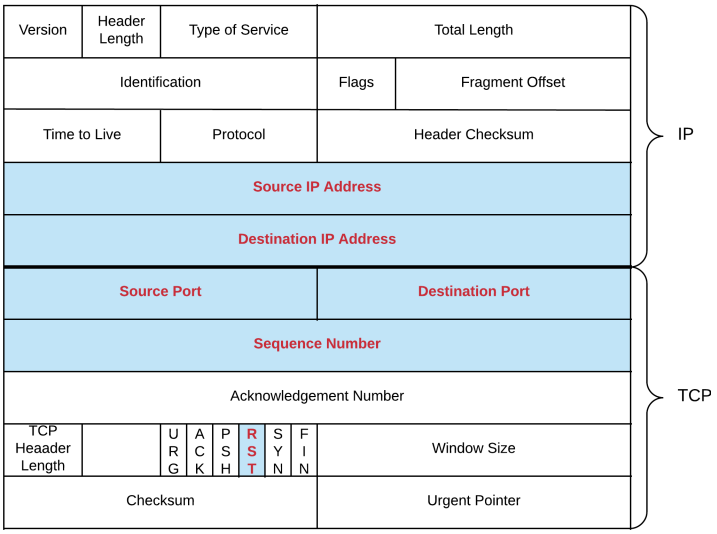

Figure 4: Attack Packet Header

In figure 4, we show the specific fields in the TCP/IP header that need to be changed in order to perform TCP reset attack. Here source IP will be the video streaming server's IP address, Destination IP will be victim's IP address, and the port numbers will be set correspondingly. Sequence number must be correctly discovered through sniffing. Finally RST bit must be set to 1. The other fields will be changed accordingly with the help of programming language libraries. Also payload to this header is not really in this case as it is merely an RST packet.

7 Justification

When the victim machine in our design receives the RST packet, it does not have any idea about the packet's actual origin. If we correctly specify IP addresses, port numbers and sequence number, it can perfectly mimic its identity as an

original. So the victim machine has no option other than to terminate its TCP connection. The main challenge in this approach is to forge a packet with correct sequence number and to send it quickly within the victim's window. If we can do that, it can be inferred from above discussion that our TCP reset attack will be successful.