Roll : 1605043
ICMP Blind Connection-Reset + Blind throughput reduction attack against TCP

Definition:

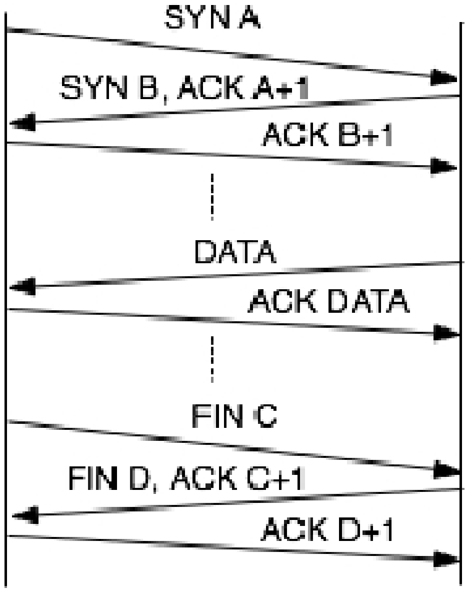
ICMP Blind connection Reset Attack:

TCP's policy of reaction to network errors depends on
the type of error being reported. If the network problem being reported is a "hard error",
TCP will immediately abort the corresponding connection. Therefore, an attack could be carried out via forging an ICMP error message that indicates a Hard Error which will reset an arbitrary TCP connection, even off path. If the target is BGP(Border Gateway Protocol), this attack can potentially DoS the whole network.

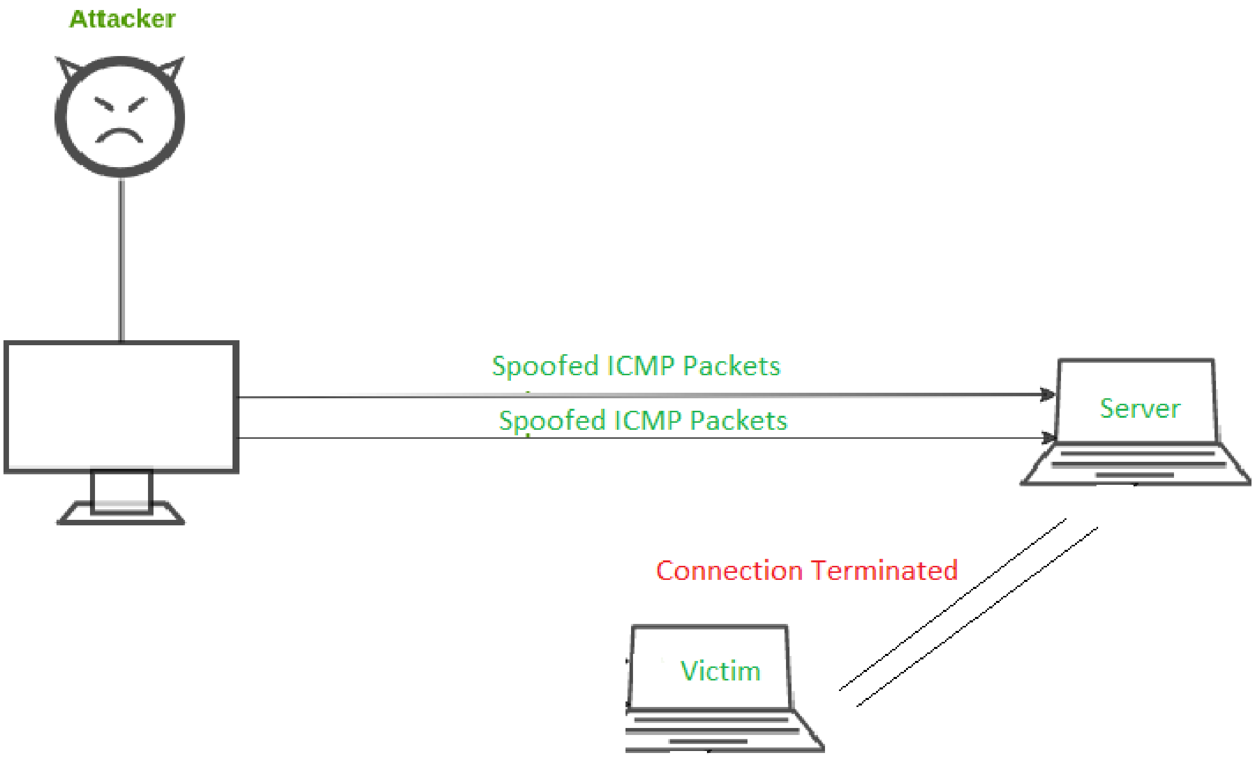
ICMP Blind throughput-reduction attack:

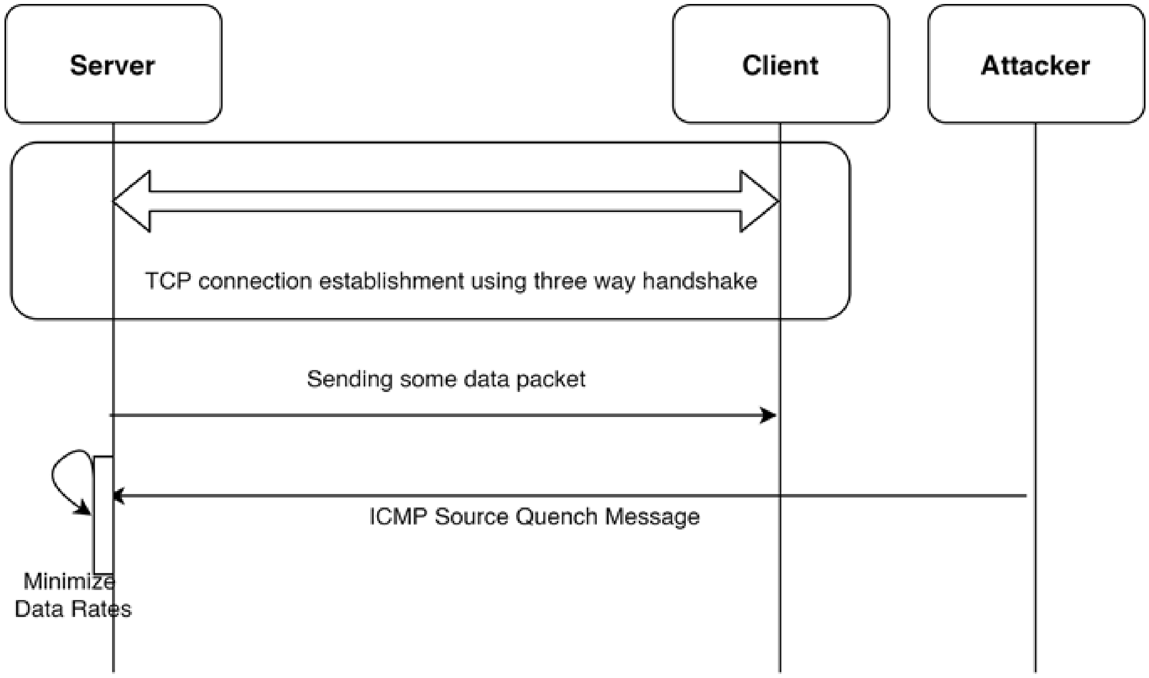
This attack can be performed by blindly reducing the throughput of a TCP connection. The Internet Architecture does not provide much support for congestion control at the Network layer. Only mechanism available is a "Choke packet" named "ICMP Source Quench". These messages are sent to the host advising it to slow down the data transmission rate. Therefore, an attack can be carried out by forging ICMP messages to fool the host into thinking that the network is congested and reducing the throughput of an arbitrary TCP connection even off-path.

TCP Timing Diagram:

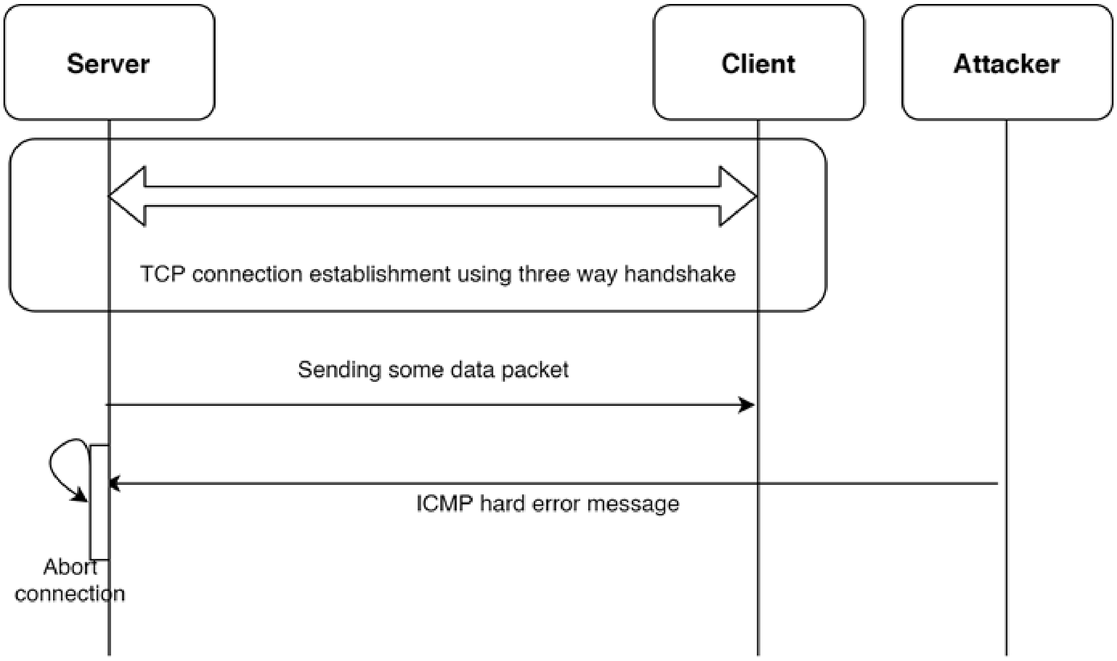


Attack Timing Diagram:




Server
Client
Attacker
TCP connection establishment using three way handshake
Sending some data packet
ICMP Source Quench Message
Minimize
Data Rates

ICMP Blind Connection reset attack

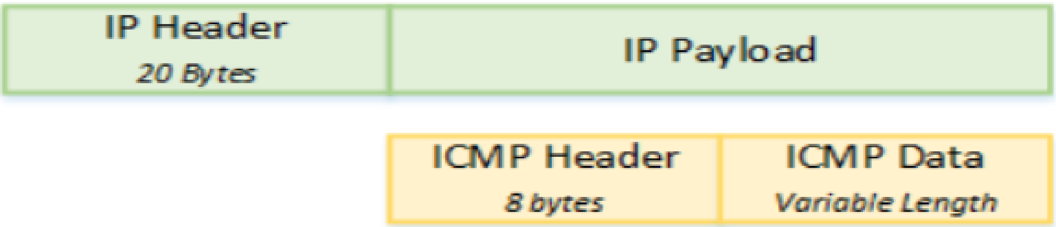


ICMP Source Quench Attack

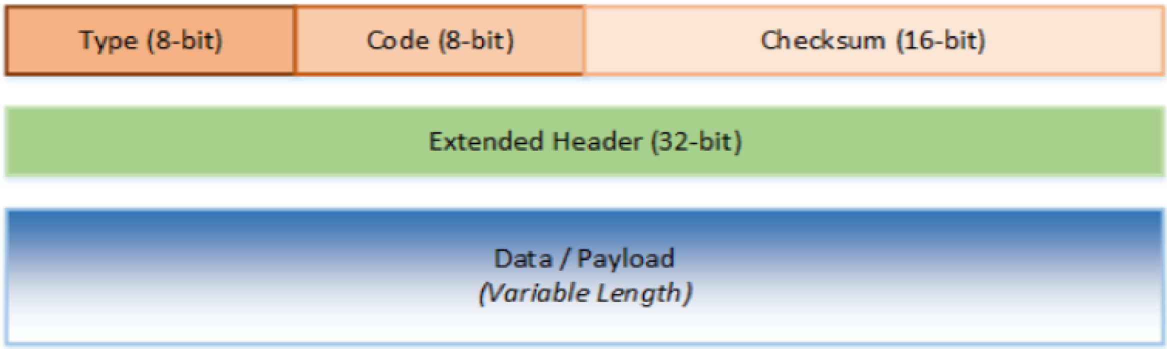
ICMP PACKET DETAILS:

It has 20 bytes of Header and Variable length ICMP message with 8 bytes of ICMP Header.Here, the last 4 bytes of the ICMP Header is generally not used.

Type	Code	Description
3	0	Destination unreachable
	1	Destination host unreachable
	2	Destination protocol unreachable
	3	Destination port unreachable
	4	Fragmentation required
4	0	Source Quench
8	0	Echo Request

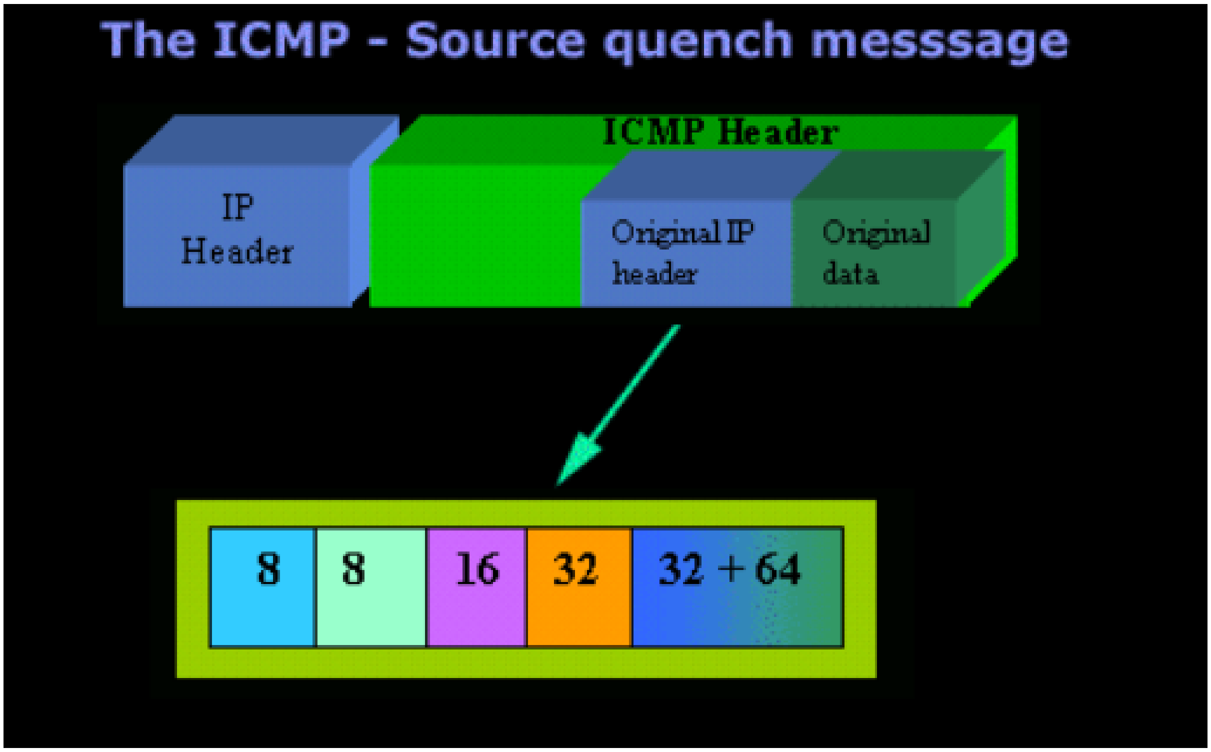


Below shows the format of an ICMP message. There are different values for the type field, which identify the ICMP message. So a type of ICMP message will use different values of the code field to specify the condition.

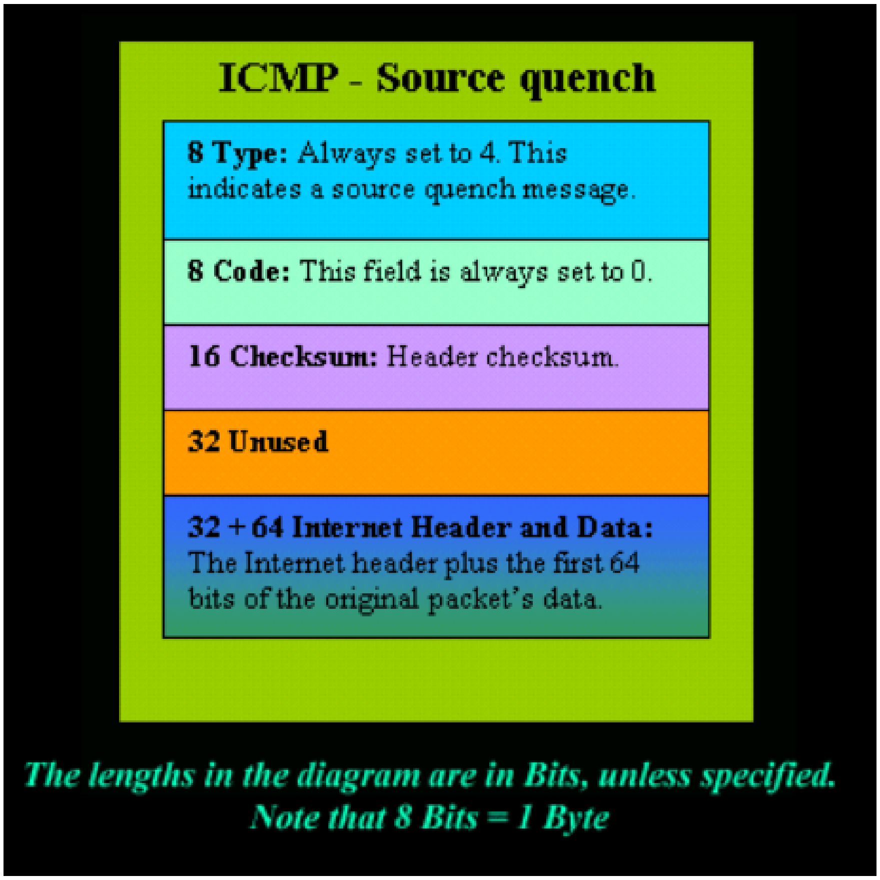


The last field talks about the Checksum. Before an ICMP message is transmitted, the checksum is computed and is inserted into the field. So at the receiving end the checksum is calculated again and verified against the checksum field. If any mismatch is found, then it confirms that an error or change has occurred.And the last 4 bytes are normally not used.

The “Choke Packet” ICMP Source Quench Message:



The ICMP - Source quench messsage



ICMP - Source quench

8 Type: Always set to 4. This indicates a source quench message.

8 Code: This field is always set to 0.

16 Checksum: Header checksum.

32 Unused

32 + 64 Internet Header and Data: The Internet header plus the first 64 bits of the original packet's data.

The lengths in the diagram are in Bits, unless specified.
Note that 8 Bits = 1 Byte

Design Strategy:

I plan to construct the tools for forging the ICMP packets using C++. The main goals:

1. For ICMP Blind reset attack, create ICMP Hard packets, which are ICMP packets of Type 3, Code 2,3 or 4 as shown above. For this a ICMP packet header has to be modified. Thousands of forged packets need to be sent to the host to increase the chances of success.
2. For Blind throughput reduction, create ICMP source quench message that fools the host into thinking that the connection is congested. These packets are ICMP messages of Type: 4, code: 0.

I plan to use two virtual machines to simulate the attackers and victims.

Justification:

The Reset vulnerability should make our Blind reset attack possible. We will generate many forged Hard Error packets with Sequence numbers of different values to increase the chance of having a packet with acceptable sequence number and thus increasing the chance of a successful attack.

The internet Architecture does not provide much support for congestion control other than a mechanism called “Choke packet” i.e. “ICMP Source Quench”. This should easily allow us to forge convincing ICMP source quench messages using our tool which can fool the host into thinking the connection is congested and resulting in reduced data transmission rate.