**TCP**

**The Sequence and Acknowledgement fields are two of the many features that help us classify TCP as a connection-oriented protocol. As such, when data is sent through a TCP connection, they help the remote hosts keep track of the connection and ensure that no packet has been lost on the way to its destination. It utilizes positive acknowledgments, timeouts and retransmissions to ensure error-free, sequenced delivery of user data. Sequence numbers are generated differently on each operating system. Using special algorithms, an operating system will generate these numbers, which are used to track the packets sent or received, and since both Sequence and Acknowledgement fields are 32bit, there are 2^32= 4,294,967,296 possibilities of generating a different number!**

#### INITIAL SEQUENCE NUMBER (ISN)

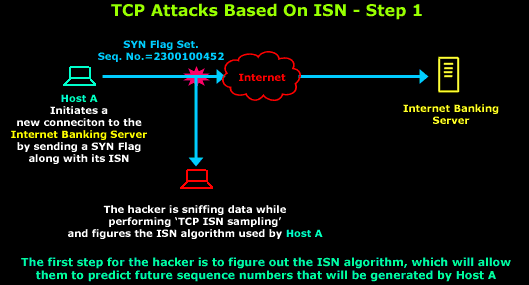
**When two hosts need to transfer data using the TCP transport protocol, a new connection is created. This involves the first host that wishes to initiate the connection, to generate what is called an Initial Sequence Number (ISN), which is basically the first sequence number that's contained in the Sequence field we are looking at. The ISN has always been the subject of security issues, as it seems to be a favourite way for hackers to 'hijack' TCP connections.**

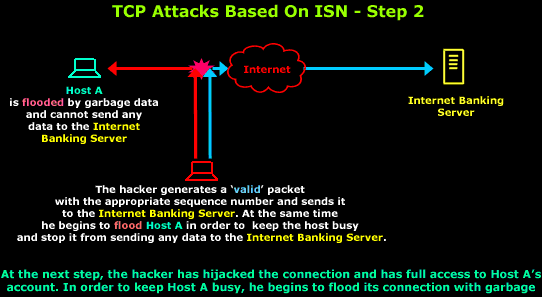
**Hijacking a new TCP connection is something an experienced hacker can alarmingly achieve with very few attempts. The root of this security problem starts with the way the ISN is generated.**

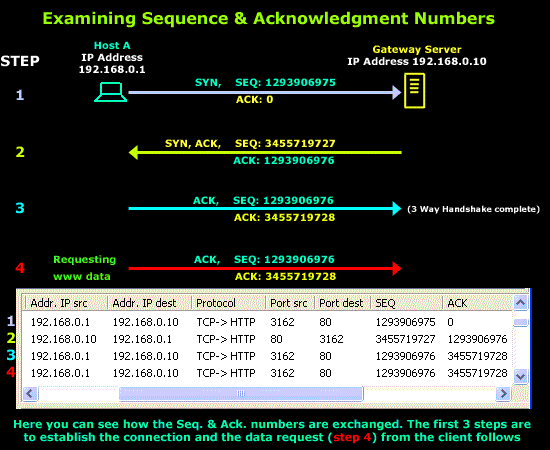
**Every operating system uses its own algorithm to generate an ISN for every new connection, so all a hacker needs to do is figure out, or rather predict, which algorithm is used by the specific operating system, generate the next predicted sequence number and place it inside a packet that is sent to the other end. If the attacker is successful, the receiving end is fooled and thinks the packet is a valid one coming from the host that initiated the connection.**

**At the same time, the attacker will launch a flood attack to the host that initiated the TCP connection, keeping it busy so it won't send any packets to the remote host with which it tried to initiate the connection.**

**Here is a brief illustration of the above-mentioned attack:**

****

****

****

**There are other important roles that the Sequence and Acknowledgement numbers have during the communication of two hosts. Because segments (or packets) travel in IP datagrams, they can be lost or delivered out of order, so the receiver uses the sequence numbers to reorder the segments. The receiver collects the data from arriving segments and reconstructs an exact copy of the stream being sent.**

**All bytes in a TCP connection are numbered, beginning at a randomly chosen initial sequence number (ISN). The SYN packets consume one sequence number, so actual data will begin at ISN+1. The sequence number is the byte number of the first byte of data in the TCP packet sent (also called a TCP segment). The acknowledgement number is the sequence number of the next byte the receiver expects to receive. The receiver ack'ing sequence number x acknowledges receipt of all data bytes less than (but not including) byte number x.**

**The sequence number is always valid. The acknowledgement number is only valid when the ACK flag is one. The only time the ACK flag is not set, that is, the only time there is not a valid acknowledgement number in the TCP header, is during the first packet of connection set-up.**

**Connection synchronization**

**Connection set-up uses the SYN flags. They are not used except for connection set-up. The establish a connection the initiator (active open) selects an initial sequence number X and sends a packet with sequence number X and SYN flag 1. The other machine (server, passive open) will select its own initial sequence number Y and will send a packet with sequence number Y, SYN flag 1, acknowledgement number Y+1 and ACK flag 1. The initiator will complete the three way handshake by sending a packet with ACK flag 1 and acknowledgement number Y+1. The connection is now established.**

**Window**

**Each side of a TCP connection advertises a window size. A connection is allowed to have that number of bytes sent but unacknowledged. Each packet gives an ack, a sequence number ack'ed, and a window. If the ack is x, and the window size is w, bytes up to x+w can be sent. (Yes, one byte beyound end of window is allowed, see window probe.) The purpose of this is to flow control based on the applications consumption of data. It is not a network flow control device, that is, a congestion control device.**

**Interactive data flow: delayed acks**

**Hold off ack for return trip data, piggy-back ack. (200 ms timer)**

**Interactive data flow: Nagel**

**Only one outstanding tiny-gram. Turn off for mouse movements, unrequited important tiny-grams.**

**Bulk data flow: slow start**

**Congestion window. Can have cwnd segments (not bytes) un-acked. After all segements are ack'ed, double cwnd. Once packets get dropped, doubling will stop. Send the min of cwnd segments and windows size bytes.**

**Resend: round-trip timer**

**Persist timer, silly window**

**If window is zero, the window will be opened by an ACK with previous ISN but non-zero window. What if this packet is dropped? Since ACK's are not ack'ed, the round-trip timer does not help. So under these conditions a window probe is done at each persist time expiration. A window probe is one byte of data beyond the end of window. The ACK will not change ISN and still have window 0 if window is still closed. Else a new ACK ISN and a new window will be advertised and the byte beyond the end of window is accepted****.**

**LINUX**

**Linux is taking advantage of RFC1948. The hashing function implementation seems to be flawless, and additional randomness is introduced. Thus, the observed ISN generator characteristics are not related to any specific TCP connection parameters. ISNs are more easily predictable when using exactly the same source port, source address, destination port and destination address, but hashing function "secret" value is modified in relatively short time intervals (and thus frequently changing observed characteristics), do not exposing system security.**

**Our attractor analysis method gives results comparable to a brute-force search of whole 24-bit space. Linux can be qualified as low-risk OS in the scope of this document.**

**WINDOWS**

**Windows 2000 and Windows NT4 SP6a are presenting almost the same level of TCP sequence number predictability, which can be qualified as medium to high, allowing attacker to get reasonably high success rates without using excessive amounts of network resources. Both systems are mildly vulnerable to attacks. There is no strong attractor structure visible (3D "cube").**

**Windows NT4 with no recent service patches and hotfixes is vulnerable to ISN guessing attacks using just a few packets, and can be easily attacked using 5,000 guesses with an almost 100% success rate. This version of Windows NT 4.0 can be qualified as high risk. Our attractor analysis method gives very good results here. NOTE: Problems in pre-SP6a Windows NT were already addressed in several advisories.**

**Windows 95 sequence numbers are very weak. But it is really difficult to understand is why this algorithm was further "weakened" in Windows 98 (SE), decreasing estimated error and number of elements required to get the right guess, in average**

**MAC-OS**

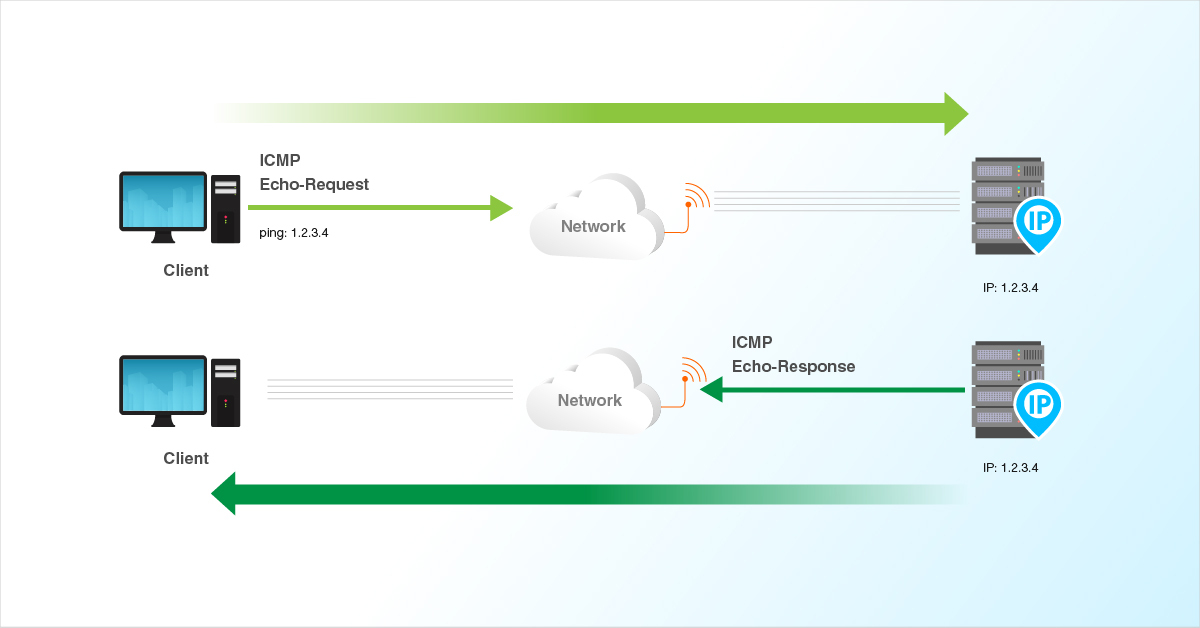
**The older MacOS9 operating system has a predictable ISN generator. The output pattern is similar to an X-wing fighter**

**The MacOS X operating system is another candidate for possible TCP sequence number guessing attacks, but is more secure than previous releases**

The Internet Control Message Protocol (ICMP) is a [network layer](https://www.cloudflare.com/learning/network-layer/what-is-the-network-layer/) protocol used by network devices to diagnose network communication issues. ICMP is mainly used to determine whether or not data is reaching its intended destination in a timely manner. Commonly, the ICMP [protocol](https://www.cloudflare.com/learning/network-layer/what-is-a-protocol/) is used on network devices, such as routers. ICMP is crucial for error reporting and testing, but it can also be used in [distributed denial-of-service (DDoS) attacks](https://www.cloudflare.com/learning/ddos/what-is-a-ddos-attack/).

**Types**

**Of ICMP Packets**

****

**Echo Request, Echo Reply**

Used to test destination accessibility and status. A host sends an Echo Request and listens for a corresponding

**Destination Unreachable, Echo Reply**

Sent by a router when it cannot deliver an IP datagram. A datagram is the unit of data, or packet, transmitted in a TCP/IP network.

**Echo Reply**.

This is most commonly done using the ping command

**Source Quench**

Sent by a host or router if it is receiving data too quickly for it to handle. The message is a request that the source reduce its rate of datagram transmission.

**Redirect Message**

Sent by a router if it receives a datagram that should have been sent to a different router. The message contains the address to which the source should direct future datagrams. This is used to optimize the routing of network traffic.

**Time Exceeded**

Sent by a router if the datagram has reached the maximum limit of routers through which it can travel.

**Parameter Problem**

Sent by a router if a problem occurs during the transmission of a datagram such that it cannot complete processing. One potential source of such a problem is invalid datagram header.

**Information Request, Information Reply**

Obsolete. These messages were used earlier by hosts to determine their inter-network addresses, but are now considered outdated and should not be used.

**Timestamp Request, Timestamp Reply**

Used to synchronize the clocks between hosts and to estimate transit time.

**Address Mask Request, Address Mask Reply**

Used to find the mask of the subnet (i.e. what address bits define the network). A host sends an Address Mask Request to a router and receives an Address Mask Reply in return.

**Router Advertisement, Router Solicitation**

Allow hosts to discover the existence of routers. Routers periodically broadcast their IP addresses via Router Advertisement messages. Hosts may also request a router address by broadcasting a Router Solicitation message to which a router replies with a Router Advertisement.