

Covert Channels

Comp 8505 Assignment 1

Mat Siwoski - A00758640

Introduction/Analysis

The purpose of this assignment was to become familiar with covert channels and to design a covert channel using the TCP/IP protocol suite. Further, Craig Rowland's application "covert_tcp.c" was to be analysed for any shortcomings within his code.

In the application, "covert_tcp.c", Craig Rowland uses three techniques for embedding covert messages within the header:

- 1.) Bounce Server: Set the ISN to a value of one less than the byte and bounce them off another server than to the original server.
- 2.) IP Header ID Field: Set the IP header ID field to the data intended for transfer and sending the data as normal. The IP Header ID Field is for reassembly of large fragments of packets.
- 3.) TCP Initial Sequence Number: Set the number to the data intended for transfer and repeatedly sending SYN packets to a remote listening host.

There are some weaknesses with Craig Rowland's code in "covert_tcp.c".

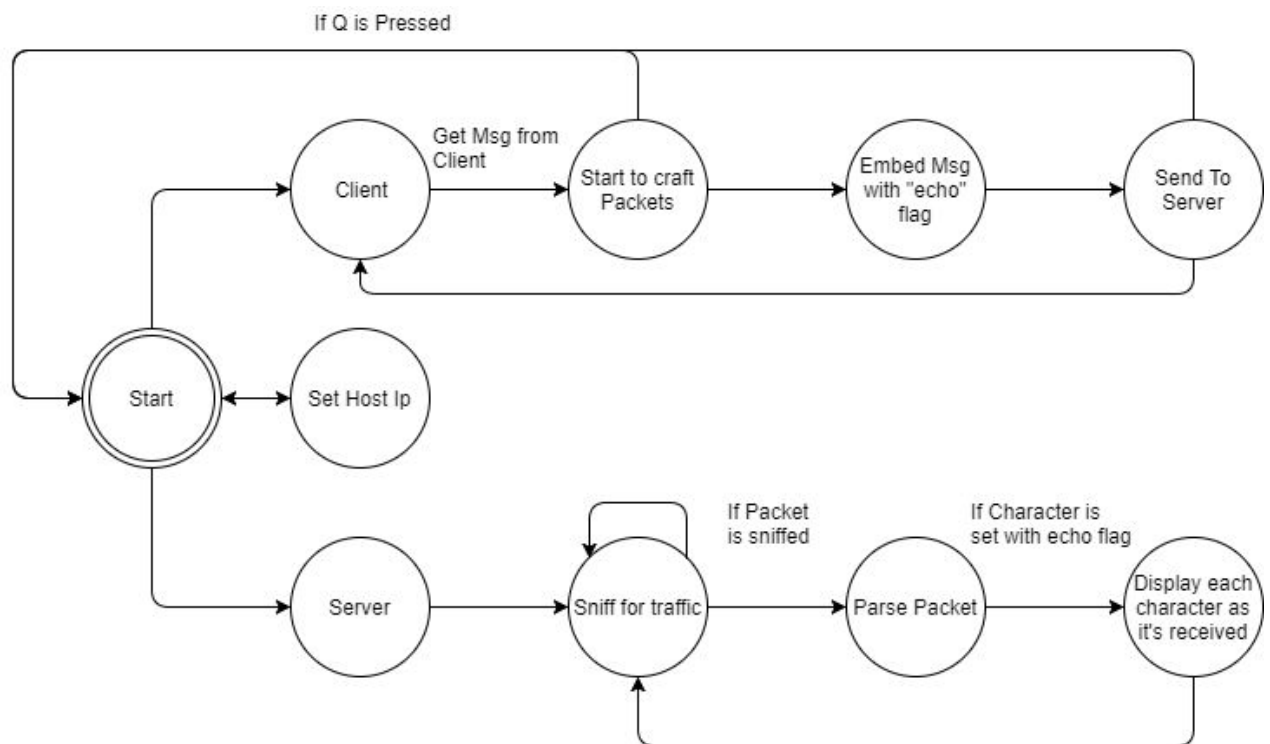
- 1.) Network infrastructure: Network infrastructure can manipulate and change the IP ID field. The reason this can happen is that this field is used for large amounts of traffic.
- 2.) Observant network administrators: By manipulating the TCP initial sequence number, if the network has an observant network administrator, they may be able to notice that the numbers are not necessarily falling into correct sequence. Although this is a possibility, it is still very difficult to notice and may still succeed.

For my assignment, I decided to use the control flags within the TCP Header, specifically to use the **ECHO** flag within the header. By using the Scapy packet manipulation program, I am able to manipulate and insert a character in place of the **ECHO** flag.

Application Requirements

- Your technique for embedding covert data into the headers must be one that is not covered by any of the techniques in the paper.
- You may only use the TCP, UDP, or IP headers for this exercise.
- You are required to show all the data supporting the success (or lack thereof) of your data embedding scheme.

Design



Tests

Here are the results of testing the application.

1.) IP Not Valid

```
root@Maciu: /home/maciu/Desktop/School/C8505/A1
Craft a covert message to send (Type Q to go to menu): Q

1. Server
2. Client
3. Set Host IP
4. Quit

Please choose an option: 3
Set the host IP: asdf

1. Server
2. Client
3. Set Host IP
4. Quit

Please choose an option: 2
IP is not valid. Please set the host IP.

1. Server
2. Client
3. Set Host IP
4. Quit

Please choose an option: █
```

2.) Setting Host IP

```
root@Maciu: /home/maciu/Desktop/School/C8505/A1
C8505 - Assignment 1 - Covert Channel

1. Server
2. Client
3. Set Host IP
4. Quit

Please choose an option: 3
Set the host IP: 192.168.0.1

1. Server
2. Client
3. Set Host IP
4. Quit

Please choose an option: 2
Craft a covert message to send (Type Q to go to menu): Q

1. Server
2. Client
3. Set Host IP
4. Quit

Please choose an option: █
```

3.) Client Sending a message

```
root@Maciu: /home/maciu/Desktop/School/C8505/A1
Craft a covert message to send (Type Q to go to menu): Hello
Sending Message: Hello

WARNING: Mac address to reach destination not found. Using broadcast.
.
Sent 1 packets.
WARNING: Mac address to reach destination not found. Using broadcast.
.
Sent 1 packets.
WARNING: more Mac address to reach destination not found. Using broadcast.
.
Sent 1 packets.
WARNING: Mac address to reach destination not found. Using broadcast.
.
Sent 1 packets.
WARNING: Mac address to reach destination not found. Using broadcast.
.
Sent 1 packets.
WARNING: more Mac address to reach destination not found. Using broadcast.
.
Sent 1 packets.
Craft a covert message to send (Type Q to go to menu): Q

    1. Server
    2. Client
    3. Set Host IP
    4. Quit

Please choose an option: 4
root@Maciu: /home/maciu/Desktop/School/C8505/A1#
```

4.) Server receiving the message

```
root@Maciu: /home/maciu/Desktop/School/C8505/A1
root@Maciu: /home/maciu/Desktop/School/C8505/A1# python3 a1.py
WARNING: No route found for IPv6 destination :: (no default route?). This affects only IPv6

C8505 - Assignment 1 - Covert Channel

    1. Server
    2. Client
    3. Set Host IP
    4. Quit

Please choose an option: 1
H
i

H
e
l
l
o

^Croot@Maciu: /home/maciu/Desktop/School/C8505/A1#
```

5.) Wireshark Result

The image shows a Wireshark network traffic capture. The top pane displays a list of packets with columns for No., Date, Time, Source, Destination, Protocol, Length, and Info. The filter bar at the top shows the expression `ip.src_host == 192.168.0.203 and ip.dst_host == 192.168.0.203`.

No.	Date	Time	Source	Destination	Protocol	Length	Info
28729	2017-09-17	23:24:41.1147678	125.639193282	192.168.0.203	TCP	54	[TCP Window Update] 72 → 61846 [ECN] Seq=1 Win=8192 Len=0
28772	2017-09-17	23:24:43.2165913	127.741016810	192.168.0.203	TCP	54	[TCP Window Update] 101 → 62568 [ECN] Seq=1 Win=8192 Len=0
28890	2017-09-17	23:24:45.3245762	129.848905702	192.168.0.203	TCP	54	[TCP Window Update] 108 → 64978 [ECN] Seq=1 Win=8192 Len=0
28913	2017-09-17	23:24:47.4326071	131.957032567	192.168.0.203	TCP	54	[TCP Window Update] 108 → 51777 [ECN] Seq=1 Win=8192 Len=0
28919	2017-09-17	23:24:49.5246331	134.049058582	192.168.0.203	TCP	54	[TCP Window Update] 111 → 36882 [ECN] Seq=1 Win=8192 Len=0
28925	2017-09-17	23:24:51.6105691	136.140994609	192.168.0.203	TCP	54	[TCP Window Update] 10 → 30817 [ECN] Seq=1 Win=8192 Len=0

The middle pane shows the details of the selected packet (No. 28890). It displays the following information:

- Frame 28890: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
- Ethernet II, Src: IntelCor_08:c2:61 (4c:08:93:09:c2:61), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
- Internet Protocol Version 4, Src: 192.168.0.203, Dst: 192.168.0.203
- Transmission Control Protocol, Src Port: 108 (108), Dst Port: 64978 (64978), Seq: 1, Len: 0
- Source Port: 108
- Destination Port: 64978
- [Stream index: 40]
- [TCP Segment Len: 0]
- Sequence number: 1 (relative sequence number)
- Acknowledgment number: 0
- Header Length: 20 bytes
- Flags: 0x00 (ECN)
- 0000 = Reserved: Not set
-0.... = Nonce: Not set
-0.... = Congestion Window Reduced (CWR): Not set
-1.... = ECN-Echo: Set
-0.... = Urgent: Not set
-0.... = Acknowledgment: Not set
-0.... = Push: Not set
-0.... = Reset: Not set
-0.... = Syn: Not set
-0.... = Fin: Not set
- [TCP Flags: *****]
- Window size value: 8192
- [Calculated window size: 8192]
- [Window size scaling factor: -1 (unknown)]

The bottom pane shows the packet bytes in hexadecimal and ASCII format. The first few bytes are `ff ff ff ff ff 4c 08 93 09 c2 61 08 00 45 00`, which correspond to the Ethernet II header.