Enhancing Protocol Steganography Using Firefly Algorithm

Asst. Prof. Dr. Ziyad Tariq Mustafa*, Asst. Prof. Dr. Bassim Abdul Baki Juma**, Authman Waleed Khalid**
* Head of Computer Science Department, University of Diyala
** Computer Eng. Department, University of Technology

Abstract — *Protocol steganography is a secret mechanism that can be used to leak significant information across a network in a manner that violates security policy and it can be difficult to detect. The huge amount of data and vast number of different protocols in the internet seems ideal as a cover for secret messages. In this paper, a proposed method is suggested for protocol steganography system enhanced by firefly algorithm. The proposed method uses the TCP/IP protocol header field of packets chosen by firefly algorithm to cover secret message. The secret characters are encoded to ASCII code before merged in IP ID field using specially designed embedding / extracting algorithms in order to make the system more complex to be defeated by attackers. Results are tested successfully with different sizes of secret messages, and different sizes cover raw packets.*

Keywrods — Covert Channel, Firefly Algorithm, IP ID, Network steganography, Protocol Steganography, Packet Header, Secret Communication.

---

1 Introduction

The number of articles related to information hiding, network steganography and their techniques has been increased. It has to be taken into account that packet steganography techniques can be involved in anything based on protocol, even out of the networking scope [1].

The worldwide network of the Internet is the perfect medium for steganography to occur. Data can be hidden in web pages that pass over the Internet, even more surreptitious and unique way to hide messages would be in the unused fields of the TCP/IP packet headers. The operation of the Internet runs on the Transmission Control Protocol and Internet Protocol (TCP/IP) [2].

Several synonyms of TCP/IP network steganography exist such as "Protocol covert channel", that defined by any communication channel as a process to transfer information in a manner that violates the system's security policy manipulating protocol header fields. Covert channels implementations are stealth, it poses a problem for highly secure environments such as government agencies and military ones [3].

One noteworthy feature of IP, for the purposes, is that it allows the fragmentation and reassembly of long datagram. TCP, on the other hand, does aim to provide a reliable channel to its clients. It is connection-oriented, and keeps its reliability properties even over networks that exhibit packet loss, reordering and duplication. Its features for implementing reliability and flow control give scope for steganographic coding [4].

A protocol header can serve as a carrier for a steganographic covert channel if a header field can take one of a set of values, each of which appears plausible to passive warden. The warden should not be able to distinguish whether the header was generated by an unmodified protocol stack or by a steganographic encoding mechanism [5].

This work is about "Enhancing Protocol Steganography using Firefly Algorithm" exploiting TCP/IP traffic to formulate a covert channel transporting secret message.

2 The Proposed System

The proposed system is composed of transmitter and receiver that are communicating through the Internet as shown in block diagram of figure (1).

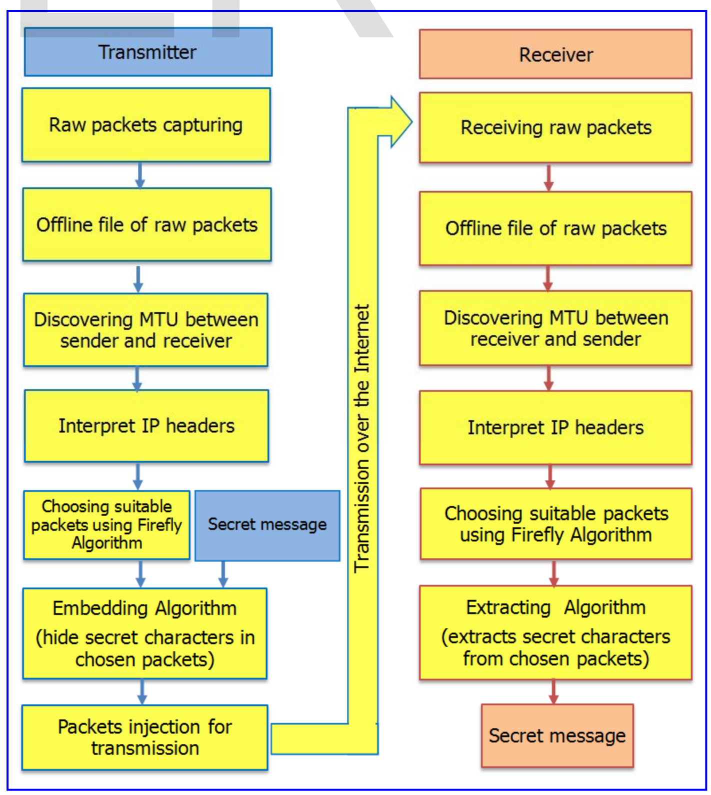

Figure (1) Block diagram of the proposed system

International Journal of Scientific & Engineering Research Volume k‰(ÚÚÜÌ¤Hû , ÈÙÊÏ¡l¥h‰K
ISSN 2229-5518

At sender side, packets are captured through the Internet using Wireshark application. Captured packets are stored in offline Pcap file in order to be interpreted and analyzed. Maximum Transfer Unit (MTU) is determined in path between transmitter and receiver while firefly algorithm is used to select suitable packets for steganography purpose.

Secret characters could be hidden in the header of selected packets using embedding algorithm. One character is embedded per one chosen packet until the entire secret file is embedded. Then, Stego packets are injected into the network.

At receiver side, packets are received, interpreted and analyzed. MTU is determined in path between receiver and transmitter while firefly algorithm is used to select suitable stego packets that used as a cover. Secret characters are extracted using extraction algorithm.

The proposed system is designed to overcome traffic mixture and easily extracts the exact secret message, as described in processes of figure (2).

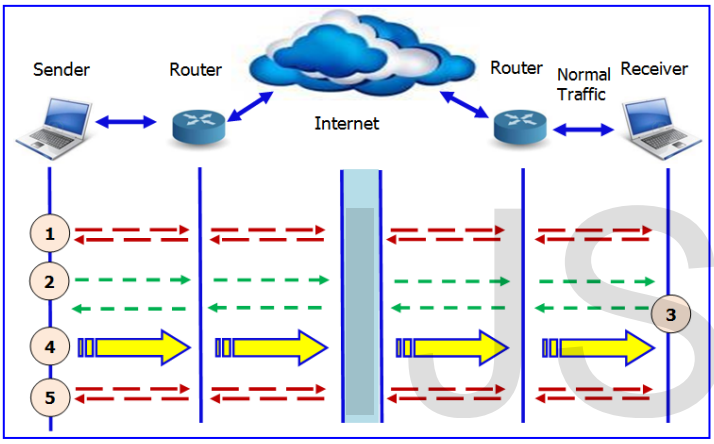

Figure (2) Description of Processes of the Proposed System

Step 1: The sender transmits ICMP packets (three starting pings) to inform the receiver about the next incoming secret message. The receiver is immediately cutout any communication, waiting and only listen to sender's message in order to avoid traffic mixture.

Step 2: Then the sender calculates MTU in path between sender and receiver.

Step 3: The receiver calculates MTU in path between receiver and sender.

Step 4: The sender starts packets injection where secret message is hidden, while the injected packets are received by the listening receiver.

Step 5: At last, the sender transmits ICMP packets (three ending pings) to inform the receiver about the end secret message.

3 The Proposed Algorithms

The main function of steganography system is to hide the secret data in an undetectable manner. This could be accomplished by embedding the data within header field of chosen packet, IP identification specifically.

3.1 The Firefly Algorithm

Swarm intelligence is employed in the proposed system at both sides, in order to enhance security of the system in an intelligent manner. The used swarm intelligence technique is firefly algorithm. Appling firefly algorithm for choosing packets is shown in algorithm (1). This algorithm is depended on the standard firefly algorithm discovered by Yang [6] in 2008.

Algorithm (1) is depending on features of packets such as the protocol type field in header of each packet as it should be TCP. Otherwise the packet should be excluded. The fitness value of the algorithm depends on the range of packet total length. The algorithm is applied on raw packets features to choose the suitable packets for hiding. MTU range for chosen packet as declared in the following formula

500 Byte ≤ chosen packet length ≤ 1452 Byte … (f)

Algorithm (1) Applied firefly algorithm

Input: (N) Raw packets
Output: (M) Chosen packets

```
Initialize population of fireflies 'N' raw packets and 'M' chosen packets
Determine (fitness function) of first raw packet N=1
(MTU value according to formula (f) and protocol type=TCP )
  While (! EOF N raw packets)
  For ( i = 1 to N )
    For ( j = 1 to M)
     IF (protocol type = TCP)
      IF (500 ≤ Total length 'L' of current packet ≤ MTU )
        Determine the Attractiveness between chosen packets
       Determine distance 'D' between sequenced packets.
       Calculate the Movement 'X' of current packet.
       Then choose jth packet.
      End IF
     End IF
    End For j
   End For i
  Rank the chosen "packets" ascending regarding total length
 End while
Return required chosen packets
```

The proposed system uses minimum limit for packet length range that is not less than (500 bytes) to avoid choosing small TCP packets that are not secure to be used for steganography system. The firefly algorithm compares each chosen packet with the others in order to rank them in ascending order according to total length.

The ranking operation is not real changes on the raw packets in Pcap file but it is a renumbering of chosen *packets orders* done by firefly algorithm.

International Journal of Scientific & Engineering Research Volume ƙǂƌ(ÚÚÜÌ‡Ꮀǂ , ÈÙÊÏıl ƔƕK
ISSN 2229-5518

The new order of packets is called *packets rank*, which is handled in the proposed system to be used by the embedding/extracting algorithm. Ranking is a special part in the design of firefly algorithm and work as precise comparison function to find the best mathematical rank for chosen packet among the other ones. In order to find the best suitable rank, the firefly algorithm is iterating and comparing packets.

3.2 The Embedding Algorithm

The embedding algorithm is designed to embed one secret character per chosen packet. The identification field is (16-bit) and the designed embedding algorithm uses the least significant (8-bit) only to make relatively small change to the value of original identification field as shown in Algorithm (2).

Algorithm (2) Embedding Algorithm

Input: Chosen packets and Secret Data
Output: Stego packets

Step 1: Read character 'char' of embedded file (secret.txt).
Step 2: Convert character 'char' to its ASCII equivalent value.
Step 3: Read value of 16-bit raw packet identification field.
Step 4: Zeroing least significant 8-bit (Raw IP ID AND 65280).
Step 6: Adding the secret 8-bit ASCII to have (New IP ID).
Step 7: Return packet for injection with (New IP ID) field.

3.3 The Extracting Algorithm

The extracting algorithm at the receiver side is designed to extract one secret character from each packet. The received packet identification field is a value of 16-bit. The extracting algorithm is shown in Algorithm (3).

Algorithm (3) Extracting Algorithm

Input: Stego packets
Output: Secret Data

Step 1: Read value of 16-bit received packet identification field
Step 2: Zeroing most significant 8-bit (Rec IP ID AND 255)
Step3: Convert the 8-bit ASCII value to equivalent character
Step 4: Save the extracted character to the received file

4. Results

Real Internet traffic was captured during this research. The study was conducted to analyze the internet traffic by capturing different type of packets at different situations. The numbers of captured sample packets are (1,000,000 packets). These packets are captured with the aid of (Dumpcap) which is part of the Wireshark application.

Statistics are calculated per session, and average results are presented. Many sessions are captured with different sizes of Pcap file such as (100, 500, 1000, 2000, 3000, 5000 and 10000 raw packets).

The analysis of the captured Internet traffic have been conducted and proven that the minimum length of TCP/IP packet is about 40 bytes (20 byte IP header + 20 byte TCP header + 0 bytes data) and the maximum IP packet length (header and data payload) founded is about 1452 byte as shown in figure (3).

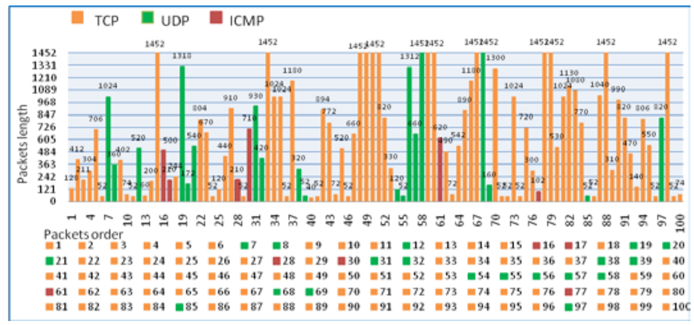

Figure (3) Histogram of Packets Diversity in Sample of (100) Packet

The proposed system is enhanced by employing swarm intelligence technique. Firefly algorithm is designed with criteria relating to packets features. Therefore, FFA is applied with the proposed system on the same above sample of (100 raw packets) as cover to embed same secret message (this is secret file from heart of univers) with previously calculated MTU value range (from 500 to 1452 byte) in the algorithm. FFA would choose packets that fit the criteria as shown in Figure (4 - A).

Before the system embeds the secret message, firefly algorithm implement sort operation as shown in Figure (4 - B). Sort operation changes the order of the chosen packets by ascending order depending on the packets total length.

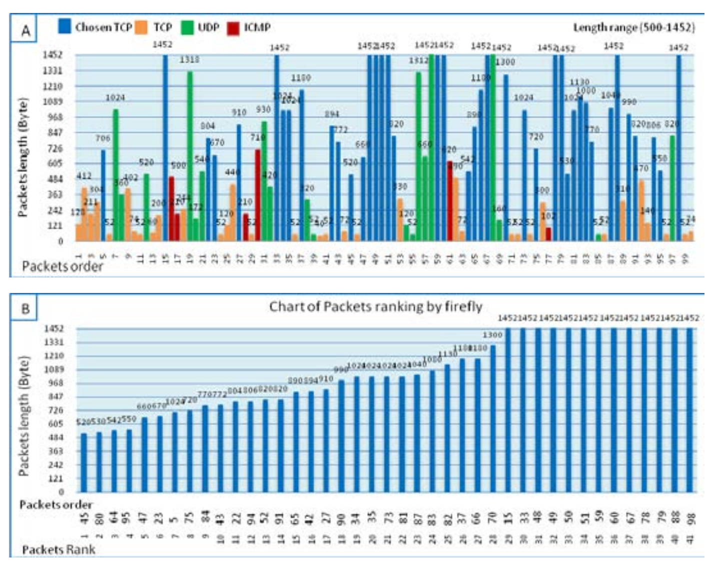

Figure (4) A – Histogram of Choosing Packets by FFA
B - Packets Ranking (Sort Operation) by FFA

International Journal of Scientific & Engineering Research Volume ĸǡ(ÚÚÜÌǡĦǡ ,ÈÙÊÏıĮγƫK
ISSN 2229-5518

The firefly algorithm iterates while performing sort operation on packets. The number of iterations is variable depending on how many secret characters are there.

Table (1) illustrates relative features of the (41) chosen packets in the sample. These packets are chosen and ranked by firefly algorithm. The first column represents the old order (D) of raw packets before firefly process and the (R) column is the new rank for packets. The table illustrates decimal value of raw IP ID for each packet and its equivalent (16) bit binary number.

The proposed system is designed with special embedding algorithm that replaces the least significant eight bits with binary bits of ASCII value belongs to the secret character (c). These operations resulted in a steganographic (16-bit) value of the new IP ID field, as illustrated in Table (1).

Table (1) Ranked chosen packets features and stego results

D	R	Total leng	Raw IP ID	Raw IP ID Binary		C	char ASCII	Stego IPID Binary		Stego IPID
45	1	520	44967	10101111	10100111	t	01110100	10101111	01110100	44916
80	2	530	60270	11101011	01101110	h	01101000	11101011	01101000	60264
64	3	542	59884	11101001	11101100	i	01101001	11101001	01101001	59753
95	4	550	18010	01000110	01011010	s	01110011	01000110	01110011	18035
47	5	660	11032	00101011	00011000	␣	00100000	00101011	00100000	11040
23	6	670	58216	11100011	01101000	I	01101001	11100011	01101001	58217
5	7	706	12559	00110001	00001111	s	01110011	00110001	01110011	12659
75	8	720	56638	11011101	00111110	␣	00100000	11011101	00100000	56608
84	9	770	60274	11101011	01110010	s	01110011	11101011	01110011	60275
43	10	772	44965	10101111	10100101	e	01100101	10101111	01100101	44915
22	11	804	58215	11100011	01100111	c	01100011	11100011	01100011	58211
94	12	806	18009	01000110	01011001	r	01110010	01000110	01110010	18034
52	13	820	9375	00100100	10011111	e	01100101	00100100	01100101	9317
91	14	820	18872	01001001	10111000	t	01110100	01001001	01110100	18804
65	15	890	46539	10110101	11001011	␣	00100000	10110101	00100000	46368
42	16	894	44964	10101111	10100100	f	01100110	10101111	01100110	44902
27	17	910	61591	11110000	10010111	i	01101001	11110000	01101001	61545
90	18	990	18871	01001001	10110111	l	01101100	01001001	01101100	18796
34	19	1024	64421	11111011	10100101	e	01100101	11111011	01100101	64357
35	20	1024	64422	11111011	10100110	␣	00100000	11111011	00100000	64288
73	21	1024	57738	11100001	10001010	f	01100110	11100001	01100110	57702
81	22	1024	60271	11101011	01101111	r	01110010	11101011	01110010	60274
87	23	1040	44659	11101011	01110000	o	01101111	11101011	01101111	44653
83	24	1080	60273	10101110	01110011	m	01101101	10101110	01101101	60271
82	25	1130	60272	11101011	01110001	␣	00100000	11101011	00100000	60192
37	26	1180	27034	01101001	10011010	h	01101000	01101001	01101000	26984
66	27	1180	46540	10110101	11001100	e	01100101	10110101	01100101	46437
70	28	1300	57120	11011111	00100000	a	01100001	11011111	01100001	57185
15	29	1452	2832	00001011	00010000	r	01110010	00001011	01110010	2930
33	30	1452	64420	11111011	10100100	t	01110100	11111011	01110100	64372
48	31	1452	9371	00100100	10011011	␣	00100000	00100100	00100000	9248
49	32	1452	9372	00100100	10011100	o	01101111	00100100	01101111	9327
50	33	1452	9373	00100100	10011101	f	01100110	00100100	01100110	9318
51	34	1452	9374	00100100	10011110	␣	00100000	00100100	00100000	9248
59	35	1452	60190	11101011	00011110	u	01110101	11101011	01110101	60277
60	36	1452	60191	11101011	00011111	n	01101110	11101011	01101110	60270
67	37	1452	46541	10110101	11001101	i	01101001	10110101	01101001	46441
78	38	1452	60268	11101011	01101100	v	01110110	11101011	01110110	60278
79	39	1452	60269	11101011	01101101	e	01100101	11101011	01100101	60261
88	40	1452	44660	10101110	01110100	r	01110010	10101110	01110010	44658
98	41	1452	58010	11100010	10011010	s	01110011	11100010	01110011	57971

Many different types of experiments have been performed, to measure imperceptibility, detectability; capacity, performance and consistency of the proposed steganography system.

5. Conclusions

From this work, several conclusions can be drawn as follows:

1) From real traffic analysis, it could be seen that TCP protocol is dominating the Internet.

2) The protocol analysis proved that the packets total lengths are vastly variant and depending on the fragmentations processes, path MTU and the protocol type.

3) The proposed steganography system enhanced by firefly algorithm which is a powerful technique that succeeds in search optimization.

4) The changes on IP ID, are very difficult to detect even when the traffic is analyzed, due to intelligent technique for chosen the packets and the special design of embedding algorithm, which are gave transparency to secret data.

5) Network steganography systems usually utilize unused field of protocol header. This work demonstrated that some of the used fields (such as IP ID) in protocol header also could be exploited as cover for steganography data.

6) The steganography system without firefly algorithm is not too secure, enhanced by firefly algorithm which is a powerful technique that succeeds in search optimization.

6. References

[1] T. Handel and M. Sandford, "Hiding Data in the OSI Network Model", Information Hiding Workshop (IH 1996), Springer Press, vol. (1174) of LNCS, pp. (23–38), Cambridge, UK, May/June, 1996.

[2] Stephen Lewis and Steven J. Murdoch, "Embedding Covert Channels into TCP/IP", Information Hiding Workshop 2005 proceeding, Cambridge, pp. (1-7), 2005.

[3] Bender, W., D. Gruhl, and N. Morimoto, "Techniques for Data Hiding", IBM Systems Journal , Vol. (35), No. (34), pp. (69-77), 1996.

[4] Eric Cole "Hiding in Plain Sight: Steganography and the Art of Covert Communication", John Wiley & Sons; Pap/Cdr Edition, 25 April, 2003.

[5] Behrouz A. Forouzan, "TCP/IP Protocol Suite", Book from McGraw-Hil Press, 4th Edition, 2010.