Error Probability Analysis of IP Time To Live Covert Channels

Sebastian Zander, Philip Branch, Grenville Armitage
Centre for Advanced Internet Architectures (CAIA)
Swinburne University of Technology
Melbourne Australia
{szander, pbranch, garmitage}@swin.edu.au

***Abstract*—Communication is not necessarily made secure by the use of encryption alone. The mere existence of communication is often enough to raise suspicion and trigger investigative actions. Covert channels aim to hide the very existence of the communication. The huge amount of data and vast number of different protocols in the Internet makes it ideal as a high-bandwidth vehicle for covert communications. A number of researchers have proposed different techniques to encode covert information into the IP Time To Live (TTL) field. This is a noisy covert channel since the TTL field is modified between covert sender and receiver. For computing the channel capacity it is necessary to know the probability of channel errors. In this paper we derive analytical solutions for the error probabilities of the different encoding schemes. We simulate the different encoding schemes and compare the simulation results with the analytical error probabilities. Finally, we compare the performance of the different encoding schemes for an idealised error distribution and an empirical TTL error distribution obtained from real Internet traffic.**

***Index Terms*—Security, Covert Channels, Network Protocols**

I. Introduction

Often it is thought that the use of encryption is sufficient to secure communication. However, encryption only prevents unauthorised parties from decoding the communication. In many cases the simple existence of communication or changes in communication patterns, such as an increased message frequency, are enough to raise suspicion and reveal the onset of events. Covert channels aim to hide the very existence of the communication. They hide within pre-existing (overt) communications channels by encoding additional semantics onto 'normal' behaviours of the overt channels.

Lampson introduced covert channels as a means to secretly leak information between different processes on monolithic systems [1]. In recent years the focus has shifted to covert channels in network protocols [2]. The huge amount of data and vast number of different protocols in the Internet makes it ideal as a high-bandwidth vehicle for covert communications. The capacity of covert channels in computer networks has greatly increased because of new high-speed network technologies, and this trend is likely to continue. Even if only one bit per packet can be covertly transmitted, a large Internet site could lose 26GB of data annually [3].

Covert channels are primarily used to circumvent existing information security policies, to ex-filtrate information from an organisation or country in a manner that does not raise suspicions of the network owners or operators. Although network covert channels may not be used frequently today, because of increased measures against 'open channels', such as the free transfer of memory sticks in and out of organisations, the use of covert channels in computer networks will increase in the near future [4].

The IP Time To Live (TTL) header field limits the lifetime of an IP packet, preventing packets from living forever during routing loops [5]. A packet's TTL is set by the sender and decremented by each network element along the path processing the packet's IP header (e.g. routers and firewalls). Packets are discarded if their TTL becomes zero while still in transit. A number of researchers have proposed different techniques to encode covert information into the TTL field [6]–[8]. Since routers and middleboxes modify the TTL fields of packets in flight and packets can take different paths through the network the TTL covert channel is a noisy channel [9].

In this paper we derive analytical solutions for the error probabilities of three known and one new TTL covert channel encoding scheme. The motivation behind this work is that the channel capacity can be computed if the error probability is known (e.g. using existing channel models such as the binary channel [10]). We simulate the different encoding schemes and compare the simulation results with the theoretical predictions demonstrating that our analytical error probabilities are valid. We also present the characteristics of realistic TTL error distributions based on real Internet traffic obtained from traffic traces. Finally, we compare the performance of the different encoding schemes for an idealised error distribution used in the simulation and a realistic error distribution obtained from the traffic traces.

The paper is structured as follows. In section II we briefly explain the basic concepts of covert channels. In Section III we define the channel error and present empirical TTL error distributions based on captured traffic traces. In Section IV we derive analytical solutions for the error probabilities of the different encoding schemes. In section V we compare the simulation results with the analytical results, and also compare the performance of the different encoding techniques for idealised and realistic error distributions. Section VI concludes and outlines future work.

1-4244-0977-2/07/$25.00

II. Covert Channels Overview

The de-facto standard covert channel communication model is the prisoner problem [11]. Two people, Alice and Bob, are thrown into prison and intend to escape. To agree on an escape plan they need to communicate, but all their messages are monitored by Wendy the warden. If Wendy finds any signs of suspicious messages she will place Alice and Bob into solitary confinement – making an escape impossible. Alice and Bob must exchange innocuous messages containing hidden information that (hopefully) Wendy will not notice.

Extending this scenario towards communication networks, Alice and Bob use two networked computers to communicate. They run some innocuous overt communication between their computers, with a hidden covert channel. Alice and Bob share a secret useful for determining covert channel encoding parameters and for encrypting/authenticating the hidden messages. For practical purposes Alice and Bob may well be the same person (e.g. a hacker ex-filtrating restricted information). Wendy manages the network and can monitor the passing traffic for covert channels or alter the passing traffic to disrupt or eliminate covert channels. Figure 1 depicts the communication model (Alice sending to Bob).

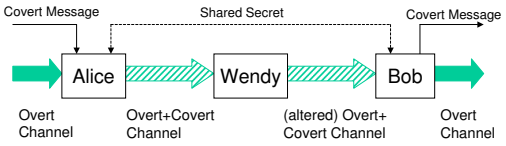

Figure 1. The prisoner problem – model for covert channel communication

In computer networks Alice and Bob do not have to be the sender and receiver of the overt communication. One or both of them may act as a middleman (see Figure 2). If Alice can observe and manipulate an existing overt communication from an innocent sender that reaches Bob, she can insert a covert channel into it. Bob does not need to be the receiver of the overt communication, but merely must be able to observe it to decode the hidden information. If Bob can also alter the overt communication, he can even remove the covert channel preventing the receiver of the overt communication from discovering it. A middleman could be located for example inside a network router or inside an end host's network stack.

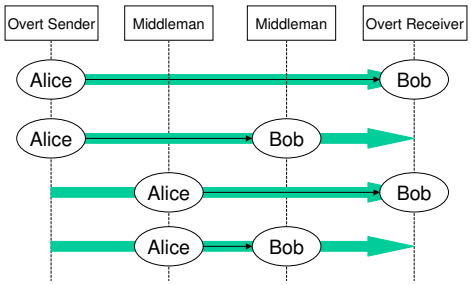

Figure 2. Communication scenarios depending on sender and receiver locations

III. TTL Covert Channel Errors

In this section we analyse the sources of error for covert channels implemented through modulation of the TTL value. A covert channel bit is mapped onto a TTL value, or a succession of TTL values in a number of different ways. Although reasonably stable between two end-points, the TTL value is nonetheless subject so some noise:

- Deletions of bits caused by loss of overt packets,
- Bit errors caused by reordering of overt packets and
- Bit errors caused by TTL modifications and path changes.

In this paper we only focus on bit errors caused by TTL modifications and path changes. In future work we will work on a combined model for all the different errors. Routers and middleboxes modify the TTL fields of packets in flight and packets can take different paths through the network between covert sender and receiver. The result is that TTL values within a packet flow change between consecutive packets and this causes bit errors on the TTL covert channel.

In the analysis that follows we define a 'TTL error' as a deviation in the TTL value from the most common (modal) value of the TTL during the life of a packet flow (identified by the 5-tuple of IP addresses, port numbers and protocol). Let the most common TTL value be TTL_{norm}. Then for a packet i of the flow the TTL error is:

$$X_i = TTL_i - TTL_{norm}. \quad (1)$$

We analysed TTL changes for seven packet traces of different size, origin and date containing a mix of traffic (including web, peer-to-peer, game, and email traffic). We only consider flows with at least four packets and at least one packet per second to limit the amount of data. The traffic traces are described in more detail in [9]. Figure 3 and 4 show the TTL error distributions for the Leipzig and Waikato trace respectively. Note that the y-axes are logarithmic and we only show error rates $\geq 1^{-7}$.

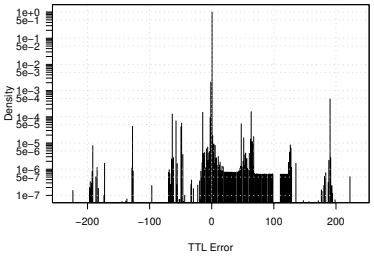

Figure 3. TTL error distribution for the Leipzig dataset

The empirical error probability is less than 0.5%. Error values are largely confined between -200 and 200, and the error probability does not monotonically decrease with increasing TTL error. For some datasets there are characteristic peaks around ± 64, ± 128 and ± 191 (see Figure 4). These peaks are caused by middleboxes manipulating the TTL field of packets

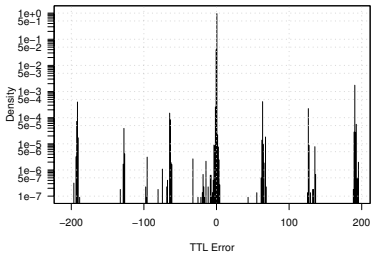

Figure 4. TTL error distribution for the Waikato dataset

of TCP flows [9]. For space reasons we cannot show the other distributions but their main characteristics are fairly similar.

IV. Error Probability Analysis

In order to determine the capacity of a channel, it is necessary to determine its error probability distribution. In this section we derive error probability distributions for a number of different covert channels that modulate the TTL in different ways.

A. Assumptions

Let the discrete random variable X_i be the TTL error of a packet *i*. We base our analysis on the following assumptions:

1) The covert data is uniform random distributed (the probability of a 0 or 1 being transmitted is equal to $\frac{1}{2}$). This is the case if the covert data is encrypted with a cipher producing a uniform random distribution.
2) We assume that only one bit of covert data is encoded per TTL. This simplifies the analysis and intuitively also maximises the stealth of the channel (future research will investigate encoding multiple bits).
3) We assume all X_i are independent identical distributed (i.i.d.) random variables and the probability distribution is stationary (reasonable assumption if the covert sender encodes covert data into multiple parallel flows).

B. Direct Encoding

Qu *et al*. proposed to encode covert bits directly into the TTL field [6]. The least significant bit in each TTL is replaced by the covert bit to be sent (see Figure 5). Since the TTL is decremented by one per hop between the covert sender and receiver, the receiver needs to know the hop count in order to decode the covert information.

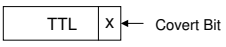

Figure 5. Direct Encoding of covert bits into the TTL field

For direct encoding techniques the error probability only depends on the error occurring for each packet independently of other packets. Errors occur if the absolute value of the TTL error is greater than zero and an odd number. Because even errors do not modify the lowest bit, they do not cause an error in the covert channel. Since the maximum TTL value is 255 the error probability is:

$$P_D = \sum_{k=-128}^{127} P(X = 2k+1). \quad (2)$$

C. Mapped Encoding

In mapped encoding schemes a 0-bit and a 1-bit are encoded as two different TTL values. Usually one of the TTL values is TTL_{norm} of the packet flow and the other value is a slight modification. Qu *et al*. proposed encoding a 0-bit as TTL_{norm} and a 1-bit as increase of TTL_{norm} by some integer Δ [6]. Zander *et al*. proposed to encode a 0-bit as TTL_{norm} and a 1-bit as decrease of TTL_{norm} by some integer Δ [8]. Effectively Δ is the absolute difference between the TTL value for a logical 0 and the TTL value for a logical 1. Figure 6 shows an example for both techniques.

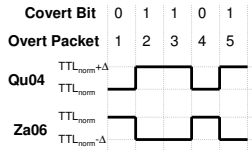

Figure 6. Mapped encoding by modulating the TTL value

We assume that the receiver either knows the mapping or learns the mapping by watching the TTL stream and assuming the two most common TTL values are the symbols for a logical 0 and a logical 1. Then the error probability only depends on the error occurring for each packet independently of other packets.

First we derive the error probability for Za06 encoding. The error probability for $0 \rightarrow 1$ and $1 \rightarrow 0$ errors is not identical. The probability for $0 \rightarrow 1$ errors is:

$$P_{0\rightarrow 1} = P(X \leq -\left\lceil \frac{\Delta}{2} \right\rceil) \quad (3)$$

where $\lceil . \rceil$ is the ceiling function. The probability for $1 \rightarrow 0$ errors is smaller for even Δ because we assume the receiver decodes a 1-bit in case the received symbol is exactly the threshold value (value in the middle between a 0-bit and 1-bit):

$$P_{1\rightarrow 0} = P(X \geq \left\lceil \frac{\Delta}{2} + \frac{1}{2} \right\rceil). \quad (4)$$

Given assumption 1 it does not affect the overall error probability how the receiver decides at the threshold. However, in general it is best to decode the threshold value as the bit occurring most frequently in the data. The overall error probability follows from assumption 1 and Equations 3 and 4:

$$\begin{aligned} P_{M_{Za06}} &= \frac{P_{0\rightarrow 1}(\Delta)}{2} + \frac{P_{1\rightarrow 0}(\Delta)}{2} \\ &= \frac{P(X \leq -\lceil \frac{\Delta}{2} \rceil)}{2} + \frac{P(X \geq \lceil \frac{\Delta}{2} + \frac{1}{2} \rceil)}{2}. \end{aligned} \quad (5)$$

Analogue the error probability for Qu04 is derived to:

$$P_{M_{Qu04}} = \frac{P(X \geq \lceil \frac{\Delta}{2} \rceil)}{2} + \frac{P(X \leq -\lceil \frac{\Delta}{2} + \frac{1}{2} \rceil)}{2}. \quad (6)$$

If the error distribution is symmetric $P_{M_{Za06}}$ and $P_{M_{Qu04}}$ are identical.

D. Differential Encoding

Differential encoding encodes the covert bits as change between the TTL values of subsequent packets. Lucena *et al*. described a technique to modulate the IPv6 Hop Limit field (the IP TTL equivalent in IPv6) [7]. They proposed to encode one bit per packet pair where a logical 1 is encoded as TTL increase by Δ and a logical 0 as TTL decrease by Δ (referred to as Lu05 encoding). This technique is problematic because long series of 0 or 1 bits lead to a large decrease or increase of the TTL. Since the TTL is an 8-bit field it can actually happen that the TTL value 'wraps-around' in the number space. Therefore, we also analyse an improved novel differential encoding technique here.

The sender encodes a logical 0 by repeating the last TTL value. A logical 1 is encoded by a TTL change, alternating between the two possible values (see Table I). The receiver decodes a constant TTL as logical 0 and a TTL change as logical 1. The scheme is similar to the Alternate Mark Inversion (AMI) coding (therefore referred to as AMI encoding).

Table I
MODIFIED TTL BASED ON COVERT BIT AND PREVIOUS TTL

Encode	Previous TTL	Current TTL
0	TTL	TTL
0	TTL - Δ	TTL - Δ
1	TTL	TTL - Δ
1	TTL - Δ	TTL

Figure 7 shows an example for Lu05 and AMI encoding schemes (for the same sequence of covert bits as in Figure 6).

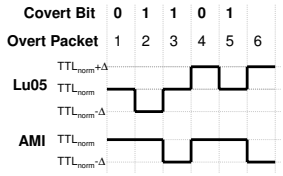

Figure 7. Differential encoding of covert bits as TTL changes

Differential schemes encode covert bits as change between two TTL values and therefore the error probability depends on the difference of the two errors. Let $Z = Y - X$ be the difference of the two TTL error distributions of two consecutive packets *x* and *y*. Then the probability that *Z* is larger than some integer *z* can be computed using the discrete convolution [12]:

$$P(Z \geq z) = \sum_{m=z}^{\infty} \sum_{n=-\infty}^{\infty} P(X = n) \cdot P(Y = m + n). \quad (7)$$

For AMI encoding a $0 \rightarrow 1$ error occurs when the absolute value of *Z* is larger then $\frac{\Delta}{2}$ (assuming at the threshold the receiver always decodes a 0-bit). A $1 \rightarrow 0$ error occurs when *Z* is in the interval $[\frac{\Delta}{2}, \frac{3\Delta}{2} + \frac{1}{2})$ and the bit is encoded as TTL decrease or when *Z* is in the interval $(-\frac{3\Delta}{2} - \frac{1}{2}, -\frac{\Delta}{2}]$ and the bit is encoded as TTL increase. This is because any TTL change larger than $\frac{\Delta}{2}$ is decoded as logical 1. The probability that a logical 1 is encoded as increase/decrease is $\frac{1}{2}$ given assumption 1. Then the overall error probability is:

$$\begin{aligned} P_{AMI} &= \frac{P_{0\rightarrow1}(\Delta)}{2} + \frac{P_{1\rightarrow0}(\Delta)}{2} \\ &= \frac{P(|Z| > \frac{\Delta}{2})}{2} + \frac{\frac{1}{2}P(\lceil \frac{\Delta}{2} \rceil \leq Z < \left\lceil \frac{3\Delta}{2} + \frac{1}{2} \right\rceil)}{2} \\ &+ \frac{\frac{1}{2}P(-\left\lceil \frac{3\Delta}{2} + \frac{1}{2} \right\rceil < Z \leq -\lceil \frac{\Delta}{2} \rceil)}{2}. \end{aligned} \quad (8)$$

For Lu05 a $0 \rightarrow 1$ error occurs when *Z* is larger than Δ (assuming at the threshold the receiver always decodes a 0-bit) and a $1 \rightarrow 0$ error occurs when *Z* is smaller equal than $-\Delta$. The overall error probability is:

$$\begin{aligned} P_{Lu05} &= \frac{P_{0\rightarrow1}(\Delta)}{2} + \frac{P_{1\rightarrow0}(\Delta)}{2} \\ &= \frac{P(Z > \Delta)}{2} + \frac{P(Z \leq -\Delta)}{2}. \end{aligned} \quad (9)$$

Note that the peak-to-peak amplitude is 2Δ for Lu05 in comparison with all the other schemes.

E. Error Probability Distribution

The actual error probabilities can be computed based on the empirical error distribution (see Section III). Alternatively, if a theoretical model for the error exists, it can be used to compute the error probabilities. Furthermore, Chebyshev's inequality provides a very loose upper bound on the probability independent of the actual distribution [12]. Assuming σ^2 is the variance and *k* is an integer ($k \geq 1$) the upper bound on the error probability is given by:

$$P(|X| \geq k\sigma) \leq \frac{1}{k^2}. \quad (10)$$

V. SIMULATION

We simulate all encoding schemes with artificial overt traffic and measure the error rates. We then compare the error rates obtained in the simulation with the theoretical error probabilities. Since there exist no models for realistic TTL error distributions and the sole purpose of our simulation is to verify our analytical error probabilities we use a simple idealised TTL error model.

A. Methodology

A custom-build tool simulating a communication channel between covert sender and receiver performs the simulation. All encoding techniques described in Section IV have been implemented. The sender-part of the simulator encodes covert

bits into the TTL fields of a stream of artificial overt IP packets (synthetic packet trace). Afterwards, the packets TTL values are modified according to a specified error distribution to simulate the channel error. Finally, the receiver-part of the simulator decodes the covert bits from the stream of overt packets. The error rate is the number of wrongly decoded bits divided by the total number of bits.

In all simulations we use uniform random covert data to avoid any bias towards specific input data. The overt data is a synthetic packet trace with approximately 42 million packets. The error is simulated using a Normal distributions with mean zero and different standard deviations $\sigma = \{0.75, 1, 1.5\}$. The values of σ have been chosen such that the resulting error rates are in a similar range as the error rates for empirical TTL error distributions (see Section V-C). Every simulated experiment is repeated 20 times.

We define A as the peak-to-peak signal amplitude of the encoding schemes (difference between the signal level of 1-bit and 0-bit). Then for direct schemes $A = 1$, for Lu05 $A = 2\Delta$ and for all other techniques $A = \Delta$. We vary the amplitude within a limited range to investigate its influence on the error rate, but avoid large changes that would compromise stealth.

Since the Normal distribution is symmetric the mapped error probabilities (Equations 5 and 6) give identical results. Therefore we only simulate Za06 as representative for both mapped encoding techniques. The most common TTL value TTL_{norm} is always set to 128. For direct encoding schemes we assume perfect knowledge of the true hop count at the receiver. For Lu05 the receiver detects wrap-wounds meaning no additional errors are introduced because of wrap-arounds (see Section IV-D).

B. Comparison of Theoretical Error Probabilities and Simulation Results

For comparing the simulation results with the theoretical probabilities we use the relative root mean square error (RMSE), which is the RMSE of the simulated error rates x_i compared to the theoretical error probability $\hat{x}$ divided by $\hat{x}$:

$$\delta RMSE = \frac{RMSE}{\hat{x}} = \frac{\sqrt{\frac{1}{N}\sum_i (\hat{x} - x_i)^2}}{\hat{x}}. \qquad (11)$$

Table II shows the relative RMSE for direct encoding. The difference between theoretical results and simulation results is very small for all σ.

Table II
THEORETICAL ERROR PROBABILITIES VS. SIMULATION RESULTS FOR DIRECT ENCODING

Sigma	Relative RMSE [%]
0.75	0.01
1.0	0.02
1.5	0.01

Figure 8, 9 and 10 show the relative RMSEs for mapped, AMI and Lu05 encoding respectively depending on the standard deviation of the error σ and the signal amplitude A.

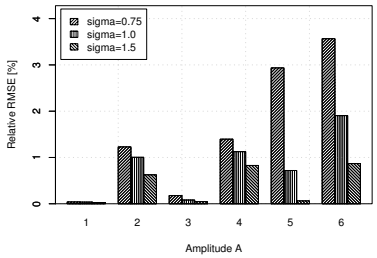

Figure 8. Theoretical error probabilities vs. simulation results for mapped encoding

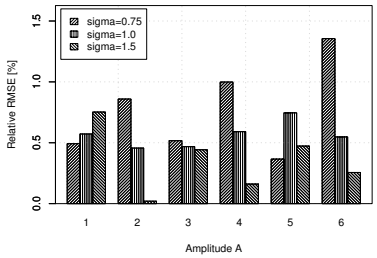

Figure 9. Theoretical error probabilities vs. simulation results for AMI encoding

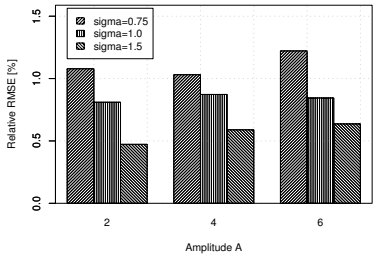

Figure 10. Theoretical error probabilities vs. simulation results for Lu05 encoding

The relative RMSE is generally $\leq$ 1.5% indicating a good match between the theoretical error probabilities and the simulation results. However, there are few larger relative RMSEs in Figure 8 for *A*=5,6. These are not caused by errors in the theoretical error probabilities or the simulation, but simply by the fact when errors are rare because of small σ and large A the statistical variance of the simulation results is high. For example, the relative standard deviation (standard deviation divided by theoretical error probability) increases from $\leq 0.3\%$ for $A \leq 4$ to $\geq 2.7\%$ for $A > 4$. A larger number of overt packets in the synthetic trace or a much larger number of repetitions would lead to smaller relative RMSEs.

C. Error Rates for Idealised and Empirical Error Distribution

Figure 11 compares the error rates of the different encoding schemes for the idealised (Normal distributed) random noise depending on the amplitude for standard deviations $\sigma = 0.75$

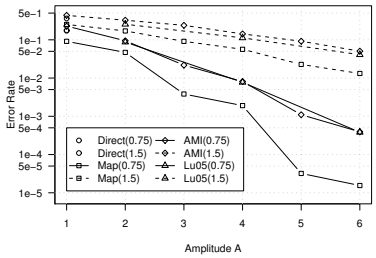

Figure 11. Error probabilities for idealised noise (Normal distribution)

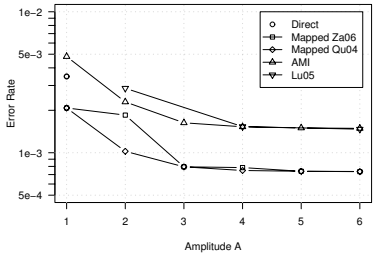

Figure 12. Error probabilities for realistic error distribution (Leipzig dataset)

and $\sigma = 1.5$ (the smallest and highest values). Note that the y-axis is logarithmic.

The figure shows that for smaller σ error rates are much lower for the same amplitude. The error rates of the AMI and Lu5 schemes are similar. Mapped encoding provides noticeable smaller error rates than the differential schemes. Direct encoding performs better than the differential schemes, but worse than mapped techniques.

As shown in Section IV in reality the TTL error does not follow a Normal distribution. Figure 12 shows the error rates for the different encoding techniques and different amplitudes for the empirical TTL error distribution obtained from the Leipzig dataset (with a logarithmic y-axis). Since the empirical distribution is not exactly symmetric we consider both mapped techniques separately. For space reasons we cannot show results for other datasets.

While the absolute error rates are different for the empirical error distribution when compared to Figure 11, qualitatively the performance of the techniques relative to each other is similar. Mapped techniques have the lowest error rate and the difference between both mapped schemes is small. Differential techniques have the highest error rates with AMI and Lu05 showing similar performance. Direct encoding performs better than differential encoding but worse then mapped encoding. Since the empirical error distribution has long tails, the error rate does not decrease as quickly with increasing amplitude as in the case of idealised noise (see Figure 11).

Note that the empirical TTL error distributions exclude very small flows (see Section IV). While Figure 12 provides a broad indication of the real error rates, the actual error deviates if the covert channel uses any flows regardless of their size.

VI. Conclusions and Future Work

A number of researchers proposed encoding covert information into the IP TTL field. This covert channel is not error-free because TTL fields are modified between covert sender and receiver, and packets can take different paths through the network. In this paper we derived analytical solutions for the error probabilities of three known and one novel TTL covert channel encoding techniques. We implemented all encoding schemes and simulated their use with artificial overt traffic and idealised noise. Our results show that the simulation error rates are very similar to the theoretical error probabilities for different standard deviations of the error and different amplitudes of the encoded signal. We also presented empirical TTL error distributions of real Internet traffic obtained from different traffic traces. Finally, we compared the error rates of the different encoding schemes for idealised and empirical error distributions.

There are a number of issues left for further research. We are developing a software framework for testing covert channels across real networks and for emulating covert channels with overt traffic taken from traffic traces. This software would make it possible to measure error rates in real networks and for emulated overt traffic, allowing us to compare the theoretical error probabilities with realistic error rates. We also plan extending our theoretical error probabilities to include bit errors from lost or reordered overt packets.

References

[1] B. Lampson, "A Note on the Confinement Problem," *Communication of the ACM*, vol. 16, pp. 613–615, October 1973.

[2] S. Zander, G. Armitage, P. Branch, "A Survey of Covert Channels and Countermeasures in Computer Network Protocols," *(accepted for publication in) IEEE Communications Surveys and Tutorials*, 2007.

[3] G. Fisk, M. Fisk, C. Papadopoulos, J. Neil, "Eliminating Steganography in Internet Traffic with Active Wardens," in *Proceedings of 5th International Workshop on Information Hiding*, October 2002.

[4] M. Van Horenbeeck, "Deception on the Network: Thinking Differently About Covert Channels," in *Proceedings of 7th Australian Information Warfare and Security Conference*, December 2006.

[5] J. Postel, "Internet Protocol," RFC 0791, IETF, Sept. 1981. http://www.ietf.org/rfc/rfc0791.txt.

[6] H. Qu, P. Su, D. Feng, "A Typical Noisy Covert Channel in the IP Protocol," in *Proceedings of 38th Annual International Carnahan Conference on Security Technology*, pp. 189–192, October 2004.

[7] N. B. Lucena, G. Lewandowski, S. J. Chapin, "Covert Channels in IPv6," in *Proceedings of Privacy Enhancing Technologies (PET)*, pp. 147–166, May 2005.

[8] S. Zander, G. Armitage, P. Branch, "Covert Channels in the IP Time To Live Field," in *Proceedings of Australian Telecommunication Networks and Applications Conference (ATNAC)*, December 2006.

[9] S. Zander, G. Armitage, P. Branch, "Dynamics of the IP Time To Live Field in Internet Traffic Flows," Tech. Rep. 070529A, CAIA Technical Report, May 2007. http://caia.swin.edu.au/reports/070529A/CAIA-TR-070529A.pdf.

[10] T. Cover, J. Thomas, *Elements of Information Theory*. John Wiley & Sons, Inc., 1991.

[11] G. J. Simmons, "The Prisoners' Problem and the Subliminal Channel," in *Proceedings of Advances in Cryptology (CRYPTO)*, pp. 51–67, 1983.

[12] C. M. Grinstead, J. L. Snell, *Introduction to Probability: Second Revised Edition*. American Mathematical Society, 1997.