Detection of Side-Channel Communication in a MANET Environment

B. Moore, R. Liscano, M. Vargas Martin
University of Ontario Institute of Technology
Oshawa, Canada
Brent.Moore, Ramiro.Liscano, Miguel.VargasMartin@uoit.ca

***Abstract*— Side-Channel communication is a form of traffic in which malicious parties communicate secretly over a wireless network. This is often established through the modification of Ethernet frame header fields, such as the Frame Check Sequence (FCS). The FCS is responsible for determining whether or not a frame has been corrupted in transmission, and contains a value calculated through the use of a predetermined polynomial. A malicious party may send messages that appear as nothing more than naturally corrupted noise on a network to those who are not the intended recipient. A metric known as Hamming Distance (HD) has been proposed in an attempt to differentiate purposely corrupted frames from naturally corrupted ones. In theory, it should be possible to recognize purposely corrupted frames based on how high this HD value is, as it signifies how many bits are different between the expected and the received FCS values. It is hypothesized that a range of threshold values based off of this metric exist, which may allow for the detection of Side-Channel communication across all scenarios.**

Keywords—Side-Channel Communication; MANET; F-Score; Hamming Distance; ROC curves

I. INTRODUCTION

Mobile Ad-Hoc Networks (MANETS) provide an easily configurable mobile platform where nodes can communicate without requiring the use of additional hardware to provide routing. While convenient, these networks are not without their share of drawbacks in terms of security, management, or packet loss. As with all wireless networks, there is some level of information loss or corruption due to signal fading, frame collisions, or environmental interference. Due to this, packets can become corrupt and will be disregarded by wireless receivers as noise. Side-Channel communication takes advantage of this as a method for discretely transmitting messages between two or more nodes. This process can be incredibly difficult to detect since it is hard to tell these intentionally corrupted messages from naturally corrupted ones; unfortunately for other nodes in a network, both appear as noise. Through the use of frame manipulation, it is possible for a malicious party to modify a frame in order for it to appear corrupted. Upon receiving a corrupted frame, most nodes will simply discard the frame, disregarding its existence. This behaviour is expected for all nodes on a network, unless the recipient has been configured in order to correctly receive and decode these secret frames. The type of communication described herein is known as "Side-Channel Communication", and while the process is conceptually simple to achieve, communication through this covert channel is difficult to establish.

A network frame has several fields that are used for a variety of functions in transit, such as the source and destination fields. The field responsible for determining whether or not a frame has been corrupted in transmission is known as the Frame Check Sequence (FCS), and is a four-octet length field containing a value that is calculated prior to transmission [1]. Upon arrival, the receiver node utilizes an agreed upon algorithm in an attempt to re-calculate the value of the frame's FCS field via a Cyclic Redundancy Check (CRC). If the calculated value matches the FCS field's value, then the frame has arrived safely; however, if the calculated FCS value does not match the one present in the frame, the frame is considered corrupted and immediately discarded by the node.

When determining a metric for use in detecting anomalous behaviour, such as Side-Channel, one of the requirements is that it must provide consistent results. In dealing with a MANET environment, this becomes increasingly more challenging as there is no guarantee of the surroundings or reliability of the network. There is also very little control over factors such as interference, and as such, the metric must be unaffected by FER. A metric known as Hamming Distance (HD) has been discovered to fulfill these requirements. An F-Score calculation will be used with this metric in order to provide a range of thresholds to detect Side-Channel communication.

II. BACKGROUND & RELATED WORK

A. Steganography

Steganography is a form of covert communication that dates back several millennia that is used to hide a covert message, but does not hide the fact that two parties are in communications [2]. In the internet age, steganography is most commonly linked with techniques involving graphical images or audio files as a carrier medium; however, this is not the only digital steganography currently in practice. Side-Channel communication is actually a form of steganography referred to as "Network Steganography".

The research presented Szczypiorski [3] may be accredited as one of the pioneering articles in the topic of Network Steganography. While most implementations of steganography systems are typically dedicated to multimedia, the research

978-1-4673-9795-7/15 $31.00 © 2015 IEEE
DOI 10.1109/CSCI.2015.179

presented in HICCUPS: Hidden Communication System for Corrupted Networks [3] develops a steganographic system from a network perspective. Even though research in network steganography is not uncommon, many techniques rely on optional packet header fields belonging to very specific network protocols [4]. HICCUPS [3] was developed with the idea that if messages are modified at the Data Link Layer of the OSI model, it is possible to take advantage of naturally occurring imperfections in network transmission, such as noise. The results of the study by Szczypiorski [3] demonstrated that while steganographic communication may be possible, there are very specific and challenging criteria that must be met. This includes the distinct lack of network interface cards that allow for the modification of frame header fields, such as the FCS field, in order to develop this form of Side-Channel.

In a standard wireless network, all devices receive a copy of each message sent but often disregard those messages when they are not the intended recipients. This functionality stems from the use of CSMA (Carrier Sense Multiple Access) and CSMA/CD (CSMA with Collision Detection) protocols on a network. These protocols operate at the Data Link Layer, and measure a network for an absence of traffic prior to transmission. The intended usage for these protocols is to ensure that data collisions do not occur, or occur less frequently, on a shared medium such as a wireless frequency. Szczypiorski [3] identifies these protocols as one of the properties required for a working implementation of HICCUPS to exist.

The suggested Side-Channel that is most relevant to this work is the "Corrupted Frame Mode". Szczypiorski [3] proposed that information could be exchanged through frames which feature intentionally created corrupt FCS fields. The benefit of a channel of this design is that it provides the ability to utilize nearly 100% of the bandwidth for a certain period, and relying on the functionality of CSMA, nodes that are not the intended recipients will simply discard these frames as noise. Szczypiorski [3] felt that this method was out of the scope of his research, as he was unable to acquire a network interface card allowing for the manual modification of CRC checksums. While initial research on his proposed channel was left largely untested in his work, it has become the focus of many works involving the Defence Research & Development Canada (DRDC) and UOIT [5].

B. Establishing a Side-Channel

Due to the obscure nature of the Side-Channel problem, it was unlikely that network simulators would offer the necessary functionality for modification of source code off the shelf. As such, experiments were opted to be conducted in a real network using real data, using a method for emulating the behavior of Side-Channel.

In his work, Najafizadeh [6] aimed to develop a system in which a Side-Channel link could be established. Previous works in the topic [3] had shown that a form of Side-Channel communication may be possible through modification of the FCS field using the Sinalgo [7] network simulator, an open source Java-based network simulator that provides an easier platform for modification than other simulators such as NS-2 or QualNet [6]. Najafizadeh [6] was able to establish a simulated Side-Channel communication in what was otherwise a rudimentary application [7]. Using collected data from his Side-Channel simulations, he examined the ratio of corrupt to non-corrupted traffic during periods where there existed Side-Channel, comparing them to those that had no Side-Channel. He was able to show that a system based off of the network's historical data would showcase a high degree of variance in the amount of Frame Error Rate (FER) when Side-Channel communication was occurring [6]. Using this, Najafizadeh [6] proposed an agent-based detection system that would trigger an alert depending on whether or not the variance of FER fell outside of an upper bound.

Another approach to the detection of Side-Channel communication was presented through the use of the RTS/CTS network mechanic by Madtha et al. [8] The Request to Send / Clear to Send (RTS/CTS) mechanic is one employed on many wireless networks as an optional feature used to prevent information loss due to packet collisions. In a network using this feature, a sender node will transmit an RTS frame to check the availability of a channel prior to sending out a data packet. If the channel is available, the destination node will reply with a CTS frame, informing other nodes to refrain from transmitting any data for a period of time. As soon as the sender node receives the CTS message, it will begin transmitting data packets. In the work of Madtha et al. [8], it was hypothesized that for every Side-Channel frame, there should be a corresponding RTS/CTS message pair. This means that the ratio of received application data and RTS frames should be 1:1 in a network with no data loss. In the presence of Side-Channel, this ratio will increase, and the amount of RTS messages will be significantly higher [8].

Several experiments were run using the QualNet [9] simulator, including an increase in the number of nodes, a varied number of Side-Channel links, and a range of inter-nodal distances. In the results presented by Madtha et al. [8] a distinct increase in RTS messages was shown to be disproportionate to the amount of known data packets when a Side-Channel is present. Unfortunately while this research provided promising results, several weaknesses were identified. In order for their method to work, a network is required to be running the RTS/CTS mechanic, which may not necessarily hold true for all networks. Additionally, this method is incredibly sensitive to Frame Error Rate, and in networks with a high degree of FER, there will be substantially more RTS messages as packets become naturally corrupted and require re-transmission, effectively skewing the results. The research presented by Madtha et al. [8] demonstrated a need for an FER insensitive metric.

With little to no available hardware for Side-Channel testing, the most likely option for experimentation was to use a simulation environment. Simulators provide an inexpensive, scalable solution for testing, and typically allow for easily modifiable parameters. Network simulators in particular also often provide channel models, routing, and full TCP stacks. With support for the OLSR protocol, the ability to instantiate controlled mobility, and featuring a full TCP/IP protocol stack, QualNet [9] appears to be the ideal Simulator for Side-Channel experiments. Unfortunately, there are several issues when

attempting to post-process or analyze data from QualNet scenarios, as frames generated within experiments are actually devoid of any useable information, such as FCS values. With no suitable simulator alternatives on the horizon, the need for a real-world Side-Channel implementation is strengthened.

C. Cyclic Redundancy Check

In a wireless network, corrupted packets are typically detected through the use of a Frame Check Sequence (FCS). Frame Check Sequences work by appending a fixed-length binary sequence to the FCS field in the frame. This sequence is calculated by the source node based on the data within the frame. Upon receiving a frame, the destination node recalculates the FCS sequence and compares it with the one included with the frame. If these values do not match, the frame is considered corrupt, and the node may request retransmission or drop the frame. The most common type of Frame Check Sequence is Cyclic Redundancy Check (CRC). This CRC value is calculated by considering the result of the remainder when dividing the polynomial for the data payload by the CRC polynomial.

After a CRC value has been calculated, it is appended to the FCS field and the frame may then be transmitted. There is not just one standard CRC polynomial, but rather a set of standardized polynomials defined by a variety of organizations. While the CRC calculation does assist with the detection of transmission errors, it is entirely possible that certain bits may corrupt in such a way that a receiver may still calculate a CRC identical to the one that was sent [10]. Furthermore, the work conducted by Koopman and Chakravarty [10] states that a given CRC polynomial may operate more or less effectively on any given application. Further work by Koopman [11] demonstrates that a 32-bit CRC polynomial, commonly known as the "Koopman Polynomial", provided the best error detection out of any of the standardised CRC calculations. As such, this is the CRC calculation that has been used for the experiments described herein.

D. Hamming Distance & F-Score

Throughout this work, the term "Hamming Distance" or "HD" refers to the number of bits that differ between an expected calculated CRC value and the actual value calculated by the recipient node. HD is a mathematical concept that was introduced by Richard Hamming in 1950 [12], and is commonly used today in coding theory when comparing the difference between bit strings of equal length [13]. Consider the following examples demonstrating the HD between two similar, but different strings: "CAT" & "CAR" have a difference (HD) of 1 character. When dealing with binary strings, the HD is equal to the number of ones in an XOR between two strings of length n. An example of this has been provided using the two CRC polynomials that will be utilized in this research in Fig. 1. First, the Default and Koopman 32-bit CRC polynomials must be converted to their binary notation, and then an XOR may be performed to get their HD. In Table 2.1 below, it is shown that there are 13 positions with an XOR binary value of 1, meaning that the expected HD between the Koopman and Default CRCs is 13. This, of course, does not mean that there will always be an HD value of 13, as there is a chance that natural corruption may occur. This corruption may cause certain bits to flip, creating CRCs with a variety of HDs.

Default (0x04C11DB7):	1	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	1	0	0	0	1	1	1	0	1	1	0	1	1	1	1
Koopman (0x741B8CD7):	1	0	1	1	1	0	1	0	0	0	0	0	1	1	0	1	1	1	0	0	0	1	1	0	0	1	1	0	1	1	1
XOR:	0	0	1	1	1	0	0	0	0	1	1	0	1	1	0	1	0	1	0	0	1	0	0	0	1	0	0	1	0	0	0

Fig. 1. An XOR of the Default & Koopman CRC polynomials.

The F-Score method (also known as F_1 score or F-measure) is commonly used in a variety of works for classification [14]. This approach is often used when testing the effectiveness of a feature, such as the HD between two CRC values.

III. Identifying a Side-Channel

A. Calculating a Threshold

An HD threshold is a whole number at which the maximum number of Side-Channel messages are detected, while avoiding false alarms when naturally corrupted Normal messages have a non-zero HD. The difficulty when selecting a value is that if your threshold is too high you will easily miss large volumes of Side-Channel communication; alternatively, if the threshold is too low, you will end up with a large amount of false positives. The primary measurement chosen for this threshold calculation is a concept known as F-Score.

When classifying data, the most common approach to verification is to assess the data against a trusted set of correctly identified results. In doing so, you are able to determine whether or not data has been flagged as True/False Positive, or True/False Negative. Two calculations exist, known as Precision (1) and Recall (2), which take these classifications into consideration when determining the relevance of the classification. Precision focuses largely on what fraction of the results were relevant to the classifier, by taking the number of True Positives and dividing it by the total number of data points identified as Positive.

$$Precision = \frac{True\ Positives}{(True\ Positives + False\ Positives)} \quad (1)$$

$$Recall = \frac{True\ Positives}{(True\ Positives + False\ Negatives)} \quad (2)$$

Both Precision and Recall are compounded in order to calculate a composite value known as F-Score (3). F-Score utilizes the harmonic mean of Precision and Recall in order to find the best possible combination, or in this case, the Optimal HD Threshold. F-Score is displayed as a value which falls between 0 and 1, where 0 is considered highly inaccurate and 1 is considered perfectly accurate.

$$F = 2\times\frac{(Precision\times Recall)}{(Precision + Recall)} \quad (3)$$

This work aims to present a precise range of thresholds calculated using F-Score that could be used to detect Side-Channel in a variety of situations. In order to calculate a threshold using this method, a data set with known Positive and Negative samples must exist, along with some metric to test against a threshold value (in this case, Hamming Distance).

IV. Establishing Side-Channel Experiments

A. Experimental Design

In order to establish an actual Side-Channel using hardware, one would need the ability to modify the FCS field, which, as with all MAC layer operations, is a functionality that is locked into the firmware of most current 802.11 wireless network cards. A chipset known as the Atheros AR5212 developed by Qualcomm [15] supports a flexible MAC layer allowing for modification of the device's CRC algorithm. Unfortunately, devices with this chipset are no longer in production. Without the ability to generate an alternative CRC using hardware, a substitute method for emulating Side-Channel communication is necessary. The emulated Side-Channel must also be easily recognizable during the analysis phase in order to allow for modification in post-processing. The simplest way to execute this is to establish a Constant Bitrate (CBR) communication on a port that differs from the rest of the network traffic; for this, Nping was used to send UDP messages at a fixed rate.

The goal of the experiments is to simulate a military foot soldier platoon's ad-hoc communication; as such, the selected MANET protocol, the positioning of the nodes, and the communication methods have been kept in line with potential real-world scenarios. For the experiments, a stable version of the OLSR protocol (0.6.8) was installed and run on the wlan0 interface using Ubuntu 14.04. Each of the nodes were provided with a static IP Address belonging to the 10.10.10.0/24 network, which allowed for a clear overview of the network activities in the live feed displayed in Wireshark.

The "Fire Team Wedge" formation [16] is the most basic formation a fire team can select. This formation provides the unit with visibility of the Team Leaders, while covering a large patrol area. The interval between soldiers in this formation is suggested to be 10 meters, however this inter-operative distance is variable depending on a variety of factors including visibility and terrain conditions. The flexibility of the inter-operative spacing allowed for some modification to the platoon positioning in the experiments. Due to spatial limitations, and given that volunteers did not have radios for communication, inter-operative positioning was reduced to a distance of 5 feet.

The original method of emulating Side-Channel, was to transmit traffic over a different application than the Normal traffic (for example, sending FTP data over port 21); however, during the initial testing process it was discovered that generating traffic through this level of the TCP stack did not allow for enough control over the rate of transmission. Nping is an open source network packet generation tool that is commonly used in networks for measuring response times, detecting active hosts, and can even be used to generate raw packets for stress testing, ARP poisoning, or Denial of Service attacks. The Nping tool is versatile, and provides the ability to control the rate of transmission, number of packets sent, and destination port. For the experiments, Side-Channel traffic was transmitted across port 1337, while "Normal" traffic was sent through port 80. In the experiment scenarios that were run, the rate of Side-Channel transmission was varied in order to test the HD detection technique against different ratios of Side-Channel to normal traffic communication. The agent node was not included in the OLSR network, but instead simply captured nearby traffic in a promiscuous state using Wireshark and the AirPCap TX network capture card.

Fig. 2 shows the communication links and the direction of transmission between each of the nodes in the platoon, where node 10.10.10.13 is transmitting Side-Channel, and the rest of the links consist of Normal traffic.

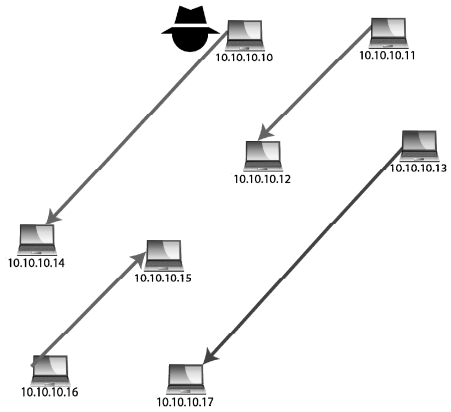

Fig. 2. Communication links between the Side-Channel and normal traffic.

Two parameters that were varied for the experiments were: the ratio of Side-Channel to normal communication, and the percentage of Frame Error Rate. While the ratio of Side-Channel to Normal traffic required running an additional experiment with each increase in volume, the modification to the percentage of FER was achievable offline in post-processing. During each of the five-minute experiments, Side-Channel communication would begin at the one minute and thirty second mark, and continue for ninety seconds. The experiments were run a total of 12 times, with the number of Side-Channel messages per second increased by an additional message each time in order to provide varying ratios of Side-Channel to normal traffic. The scenarios consisted of the following percentages of Side-Channel traffic: 9%, 12%, 14%, 19%, 22%, 25%, 28%, 30%, 35%, 37%, and 40%. These percentages correlated directly to an additional 1 Side-Channel frame per second. Through post-processing, 22 different levels of FER were introduced to each of the 12 experiments, creating 264 unique data sets for analysis.

The Wireshark capture files were parsed using TShark in order to obtain the relevant frames, and then run through MATLAB & Sinalgo. Frames corresponding to port "1337" were provided with a generated Koopman 32-bit CRC, while all other traffic was provided with a standard 32-bit CRC. Frame Error Rate was artificially introduced by corrupting a predetermined percentage of traffic using an AWGN channel.

V. Analysis of Results Using F-Score

A. Performing the F-Score Calculation

Using F-Score for calculation, several threshold values were tested for each of the experiment scenarios, with the value presenting the highest F-Score value ultimately being chosen as

the threshold. Fig. 3 shows the results of the F-Score calculations for proposed threshold values from 1 – 30. With an F-Score value of 0.9973 (on a scale from 0 – 1), the threshold of 9 was selected in this scenario. Upon further examination of the F-Score results, it is evident that while a threshold value of 9 did not have the highest number of True Positives, it did in fact have a lower number of False Negatives than some of the higher thresholds. A threshold of 10 or higher had even fewer False Positives, but began to present a larger number of False Negatives, a trend that continued as the threshold increased. This suggests that the F-Score Calculation places a higher level of significance to the best harmonic combination of False Positives and False Negatives.

Threshold	True(+)	False(+)	True(-)	False(-)	Accuracy	Sensitivity	Specificity	Precision	F-Score (FER=0.83%)
1.00	364	14	2302	0	0.9948	1	0.994	0.963	0.9811
2.00	364	14	2302	0	0.9948	1	0.994	0.963	0.9811
3.00	364	14	2302	0	0.9948	1	0.994	0.963	0.9811
4.00	364	12	2304	0	0.9955	1	0.9948	0.9681	0.9838
5.00	364	11	2305	0	0.9959	1	0.9953	0.9707	0.9851
6.00	364	8	2308	0	0.997	1	0.9965	0.9785	0.9891
7.00	364	6	2310	0	0.9978	1	0.9974	0.9838	0.9918
8.00	364	5	2311	0	0.9981	1	0.9978	0.9864	0.9932
9.00	363	1	2315	1	0.9993	0.9973	0.9996	0.9973	0.9973
10.00	361	0	2316	3	0.9989	0.9918	1	1	0.9959
11.00	352	0	2316	12	0.9955	0.967	1	1	0.9832
12.00	331	0	2316	33	0.9877	0.9093	1	1	0.9525
13.00	310	0	2316	54	0.9799	0.8516	1	1	0.9199
14.00	279	0	2316	85	0.9683	0.7665	1	1	0.8678
15.00	223	0	2316	141	0.9474	0.6126	1	1	0.7598
16.00	159	0	2316	205	0.9235	0.4368	1	1	0.608
17.00	107	0	2316	257	0.9041	0.294	1	1	0.4544
18.00	63	0	2316	301	0.8877	0.1731	1	1	0.2951
19.00	35	0	2316	329	0.8772	0.0962	1	1	0.1754
20.00	15	0	2316	349	0.8698	0.0412	1	1	0.0792
21.00	5	0	2316	359	0.866	0.0137	1	1	0.0271
22.00	2	0	2316	362	0.8649	0.0055	1	1	0.0109
23.00	0	0	2316	364	0.8642	0.00	1.00	NaN	NaN
24.00	0	0	2316	364	0.8642	0.00	1.00	NaN	NaN
25.00	0	0	2316	364	0.8642	0.00	1.00	NaN	NaN
26.00	0	0	2316	364	0.8642	0.00	1.00	NaN	NaN
27.00	0	0	2316	364	0.8642	0.00	1.00	NaN	NaN
28.00	0	0	2316	364	0.8642	0.00	1.00	NaN	NaN
29.00	0	0	2316	364	0.8642	0.00	1.00	NaN	NaN
30.00	0	0	2316	364	0.8642	0.00	1.00	NaN	NaN

Fig. 3. Demonstration of F-Score results used to select a threshold of 9.

The F-Score calculation was tested against the data from 252 out of the total 264 experiments in order to determine the possibility of a potentially universal threshold or range of thresholds that would allow for the detection of a Side-Channel in any network The analysis in Fig. 4 of the calculated Optimal HD Thresholds for each Side-Channel demonstrates that the mean, median and mode of these thresholds fall between 10 and 13, and begin to normalize once the percentage of Side-Channel increases beyond 25%. By calculating the Standard Deviation of the thresholds for each amount of Side-Channel, it is possible to say with 99% confidence that the range of Optimal Thresholds falls between the 11 – 12 range.

B. Using the Threshold to Find a Side-Channel

Now that F-Score has been used to determine a threshold value, the results of the experiments must be tested against these thresholds. Figure 5 illustrates the HD of all captured frames, where Normal Traffic is shown to have an HD between 0 and 12 depending on the level of corruption, and Side-Channel frames feature an HD value between 11 and 24. As shown here, even with a higher volume of naturally corrupted frames, there remains a distinct difference between the HD of purposely corrupted Side-Channel frames and any naturally corrupted Normal Traffic. Using the suggested threshold range described in the previous section (11 – 12), one can see that this range fits nearly centered between the two HD mean trend lines.

	9%	12%	14%	19%	22%	25%	28%	30%	33%	35%	37%	40%
0% FER	1	1	1	1	1	1	1	1	1	1	1	1
Actual FER	9	7	9	8	6	8	7	7	7	6	6	5
5% FER	11	11	10	10	10	9	9	10	9	9	9	9
10% FER	11	11	11	10	10	10	10	10	9	10	10	10
15% FER	12	11	11	11	10	10	10	10	10	10	10	10
20% FER	11	12	12	11	11	11	11	11	10	10	10	10
25% FER	12	12	12	12	11	11	11	11	10	10	11	11
30% FER	13	11	11	11	11	11	11	11	10	11	11	11
35% FER	13	12	12	11	11	11	11	11	11	11	11	11
40% FER	13	12	13	12	12	11	11	11	11	11	11	11
45% FER	12	12	12	12	11	11	11	11	11	11	11	11
50% FER	13	13	13	12	12	12	12	11	11	11	11	11
55% FER	13	12	12	12	12	11	12	11	11	11	11	11
60% FER	13	12	13	12	12	12	11	11	11	11	11	11
65% FER	13	13	13	12	12	12	12	11	11	11	11	11
70% FER	13	13	13	12	12	12	12	12	11	11	11	11
75% FER	13	12	12	12	12	12	12	11	11	11	11	11
80% FER	13	13	13	12	12	12	12	12	12	11	12	12
85% FER	14	13	13	12	12	12	12	12	12	12	12	11
90% FER	13	13	13	12	12	12	12	12	12	12	12	12
95% FER	13	13	13	12	12	12	12	12	12	12	12	12
100% FER	13	13	13	12	12	12	12	12	12	12	12	11
Mean	12.4	12.0	12.1	11.4	11.2	11.1	11.1	11.0	10.7	10.7	10.8	10.6

Fig. 4. Optimal HD Thresholds for all 264 unique experiment combinations.

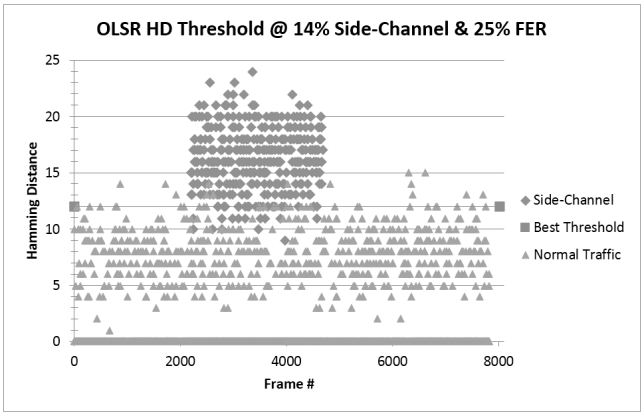

Fig. 5. A visual representation of the HD values for all frames in a scenario with 14% Side-Channel traffic & 25% Frame Error Rate.

By calculating the confidence level results at 95% and 99% confidence for each of the Normal and Side-Channel HDs, it can be stated that each of the HD means of Normal frames fall within the range of 8.0 +/-0.32 with 95% confidence, and 8.0 +/-0.42 with 99% confidence. These results suggest that the appropriate threshold should exist somewhere between the calculated population means of the two values. With F-Score shown to be capable in its ability to define a threshold based on HD, it is important to ensure the validity of the HD metric itself. One way to achieve this is through use of ROC curves.

C. ROC Curves

The Receiver Operating Characteristic (ROC) curve is a plot of the True Positive Rate (TPR) against the False Positive Rate (FPR) (1 – Specificity) for all possible thresholds of a diagnostic experiment. Assessing a threshold technique using an ROC curve is relatively simple: the more a curve represents a 90-degree angle, the more optimal the test is for that particular dataset [17]. A prediction method with the best

possible outcome would yield a point at the upper leftmost corner of the ROC space, which represents 100% Sensitivity (no False Negatives) and 100% Specificity (no False Positives). An ideal F-Score threshold should have a high TPR, whilst maintaining a low FPR. Performing an analysis using ROC curves will demonstrate how effective the calculated HD thresholds are for a given Side-Channel experiment. Fig. 6 presents the ROC curves calculated for the HD thresholds on scenarios ranging from 9% to 25% Side-Channel, and from 0.8% (Actual FER) to 95% FER.

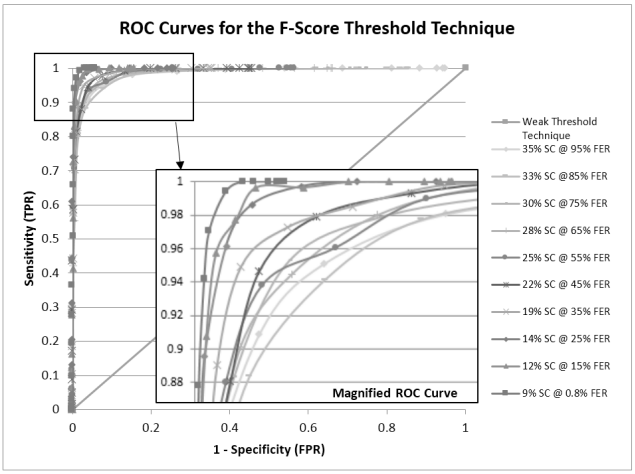

Fig. 6. Fig. 6. The ROC curve demonstrating the effectiveness of F-Score thresholds.

This technique suggests that the F-Score approach to defining thresholds is very effective when tested against the HD metric. It should be noted that the ROC curve is strongly influenced by False Negatives, and as such the scenarios with lower FER (less naturally corrupted Normal frames) depict a much higher area under the curve.

VI. Conclusion

This paper has presented the concept of Side-Channel communication through the modification of the CRC polynomials in a military oriented MANET environment and attempted to detect it, providing a range of threshold values that could be utilized across a variety of situations. The proposed range of 11 – 12 should provide an excellent starting point towards solving the issue of Side-Channel communication, as many of the techniques explored within this work have resulted in thresholds of similar values belonging to this range.

There are several areas identified within this paper that could benefit from additional research. First, the experiments presented use the same CRC polynomials for Normal and Side-Channel frames, although there is no guarantee that a malicious party would utilize one of these standard polynomials. Future work should be conducted through thorough experimentation on different standard and non-standard CRC polynomials. Another mechanism employed by many detection methods that could be further explored is Windowing, where a small subset is taken out of a larger dataset for processing. Finally, further work into the validation of the F-Score mechanic could be performed by comparing the calculated threshold range against those calculated through the use of several machine learning algorithms including Markov Chains, Bayesian modelling, and K-means. Further investigation into these techniques could provide an increased confidence for the threshold range.

VII. References

[1] "IEEE SA - 802.3-2012 - IEEE Standard for Ethernet," IEEE, 2015. [Online]. Available: https://standards.ieee.org/findstds/standard/802.3-2012.html

[2] G. C. Kessler, "An Overview of Steganography for the Computer Forensics Examiner," Gary Kessler Associates, June 2014. [Online]. Available: https://www.fbi.gov/about-us/lab/forensic-science-communications/fsc/july2004/research/2004_03_research01.htm

[3] K. Szczypiorski, "HICCUPS: Hidden communication system for corrupted networks," *The Tenth International Multi-Conference on Advanced Computer Systems ACS'2003,* pp. 31-40, 2003.

[4] B. Jankowski, W. Mazurczyk and K. Szczypiorski, "Information Hiding Using Improper frame padding," *Telecommunications Network Strategy and Planning Symposium (NETWORKS),,* pp. 1-6, 2010.

[5] V. Chea, "Hamming Distance as a Metric for the Detection of Side Channel in 802.11 Wireless Communications," MSc thesis (Business and Information Technology), University of Ontario Intitute of Technology, Oshawa, Canada, 2015.

[6] A. Najafizadeh, "Detection of Covert Communications based on Intentionally Corrupted Frame Check Sequences," MASc thesis (Electrical and Computer Engineering), University of Ontario Institute of Technology, Oshawa, Canada, 2011.

[7] D. C. Group, "Sinalgo – Simulator for Network Algorithms," [Online]. Available: http://www.disco.ethz.ch/projects/sinalgo/

[8] N. Madtha, M. Vargas Martin, R. Liscano, B. Moore, M. Salmanian, M. Li and P. Mason, "Detection of side-channel communication in ad hoc networks using request to send (RTS) messages," Toronto: IEEE 27th Canadian Conference on, 2014, pp. 1-6.

[9] "QualNet," SCALABLE Network Technologies, Inc., [Online]. Available: http://web.scalable-networks.com/content/qualnet

[10] P. Koopman and T. Chakravarty, "Cyclic redundancy checks via table look-up," *Communications of the ACM,* vol. 8, no. 31, pp. 1008-1013, 1988.

[11] P. Koopman, "32-bit cyclic redundancy codes for Internet applications," *International Conference on Dependable Systems and Networks,* pp. 459-468, 2002.

[12] R. Hamming, "Error detecting and error correcting codes," *Bell System Technical Journal,* vol. 29, pp. 147-160, 1950.

[13] K. Rosen, "Coding Theory," in *Applications of Discrete Mathematics*, New York, McGraw-Hill Higher Education, 2007, pp. 73-95.

[14] S. Beitzel, "On Understanding and Classifying Web Queries," Ph.D. thesis (Computer Science), Illinois Institute of Technology, Chicago, USA, 2006.

[15] A. Communications, "AR5002 Series Spec Sheet," [Online]. Available: http://mgvs.org/public/midge/datasheet/AR5002+spec+sheet.pdf

[16] H. D. O. T. A. Army, "FM 7-8 Infantry Rifle and Platoon Squad Field Manual," Washington DC, 1992. [Online]. Available: http://armypubs.army.mil/doctrine/DR_pubs/DR_a/pdf/fm3_21x8.pdf

[17] "Plotting and Intrepretating an ROC Curve," University of Nebraska Medical Centre, [Online]. Available: http://gim.unmc.edu/dxtests/roc2.htm