PROVIDING EFFECTIVE SECURITY IN MOBILE AD HOC NETWORKS WITHOUT AFFECTING BANDWIDTH OR INTEROPERABILITY

Lt. (N) Dan Lynch and Scott Knight
Royal Military College of Canada
Kingston, ON, Canada

Maria A. Gorlatova
Columbia University,
New York, NY, 10027

Yannick Lacharité et Louise Lamont
Communications Research Centre
Ottawa, ON, Canada

Ramiro Liscano
University of Ontario Institute of Technology
Oshawa, ON, Canada

Peter C. Mason*
Defence Research & Development Canada
Ottawa, ON, Canada

ABSTRACT

Finding security solutions for Mobile Ad hoc Networks (MANETs) that do not detrimentally affect their utility is a challenging research problem. We present mechanisms that can be used for detecting sophisticated attacks against MANETs as well as for providing methods of authentication and information leakage prevention. We implement our methods in a laboratory testbed and provided experimental evidence of their efficacy.

1. INTRODUCTION

Mobile Ad hoc Networks (MANETs) are recognised as being a disruptive technology that could have tremendous impact on military communications. Self-organising networks using simple, standardised protocols can accommodate a heterogeneity of devices (nodes) and maintain reliable communications in environments where the mobility of nodes creates a dynamic network topology. Among the important advantages of MANET technology is that the ease of set-up and network configuration should facilitate interoperability among nodes of varying capabilities as well as nodes from different forces and/or different nations. In a coalition environment where nodes of a single nation are too dispersed to form their own network, the ability to route traffic through intermediary nodes of partner nations allows spontaneous creation of allied communication networks – that is, the "organic" growth of networks takes advantage of the node density of the entire coalition. Unfortunately, the advantages of MANETs, including their ease of formation, their dynamic, distributed nature, and open wireless medium, inherently bring along with them a myriad of new and significant security vulnerabilities (Mason et. al, 2007).

Previously, we reported methods that provide solutions to attack detection using techniques that do not modify the communication protocols nor use any portion of the available communication bandwidth (Gorlatova et al., 2006, Gorlatova et al., 2007).[1] Here we further develop and refine those techniques to improve their speed and sensitivity and discuss how these improvements can be used for authentication purposes, Blue Force Tracking, and denying an attacker access to a side-channel.

1.1 Wormhole Attacks

Among the most difficult attacks to defend against in a MANET is the wormhole attack (Hu and Perrig, 2001). This attack is both a manifestation of a man-in-the-middle attack and a serious routing attack on the MANET. As

[1] In the interest of brevity, we have limited references to our own work and a few recent and salient examples in the open literature which contain full reference lists.

shown in Fig. 1, a wormhole attack consists of two colluding attackers acting in tandem to distort the perceived network topology, giving them complete control over a link in the network. One eavesdropping attacker simply forwards incoming traffic (unmodified) across an off-channel link to the second attacker who rebroadcasts it locally. The attackers are transparent to the network and its protocols, the result being that potentially distant nodes believe themselves to be one-hop neighbours-- information transmitted between these apparent neighbours, and much of the traffic in their local areas, is now sent through a wormhole created by the attackers. Once implemented, the attackers can elevate the importance of their link by providing high-bandwidth, giving them a wide range of potential attacks. Standard encryption techniques do not prevent this attack.

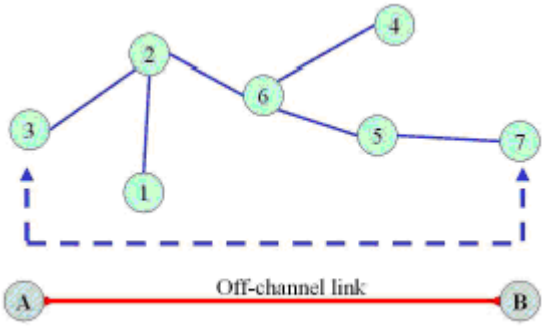

Figure 1. A Wormhole Attack launched by attackers A and B using an off-channel link to distort the network topology. Nodes 3 and 7 now believe themselves to be one-hop neighbours.

Significant research effort has been put towards the problem of mitigating the wormhole attack, much of it categorised and summarised in (Gorlatova, 2006). More recent examples include (Nguyen and Lamont, 2008, Nait-Abdesselam et al., 2008, and Sterne et al., 2007). Many published defenses involve modifications to the existing protocols, the introduction of new, bandwidth-consuming messages, or the overlay of externally-obtained location and timing information. While generally effective, none of these techniques are a panacea from a cost (both bandwidth and complexity) and interoperability perspective. Moreover, several proposed solutions, including some in (Gorlatova et al., 2006) only detect the existence of the wormhole *after* it begins to disrupt the network by selectively dropping traffic. Such solutions are unacceptable in military applications, particularly if topological information is relied upon for network situational awareness.

2. ATTACK DETECTION

In proactive MANET routing protocols such as the Optimised Link State Routing Protocol (OLSR), routing messages are broadcast periodically to maintain current routing tables. We develop in (Gorlatova et al., 2006, Gorlatova et al., 2007) an attack detection technique called Frequency-based Wormhole Attack Detection (FWAD) that uses Fourier analysis of the timing of these periodic messages. This technique can be used to detect wormholes, even if they are dormant (i.e. not dropping any traffic), and it does not require new messages, changes to the protocol, additional information overlays, nor tight network synchronisation.

FWAD works by locally constructing, at each node, a time series from the arrival times of HELLO messages and performing Fourier Analysis on this series to obtain a power spectral density (PSD). This PSD can be thought of as a fingerprint of the underlying protocol message behaviour. Comparison of the received PSD to the node's own broadcast HELLO message PSD allows for even tiny timing distortions caused by the wormhole to be detected (Gorlatova et al., 2006). The sensitivity of the technique is, however, limited by the presence of OLSR HELLO message jitter which is itself the type of statistical delay that FWAD is meant to detect. We further refine FWAD herein and demonstrate how improvements naturally enable additional security features, namely broadcast authentication and information-leakage protection.

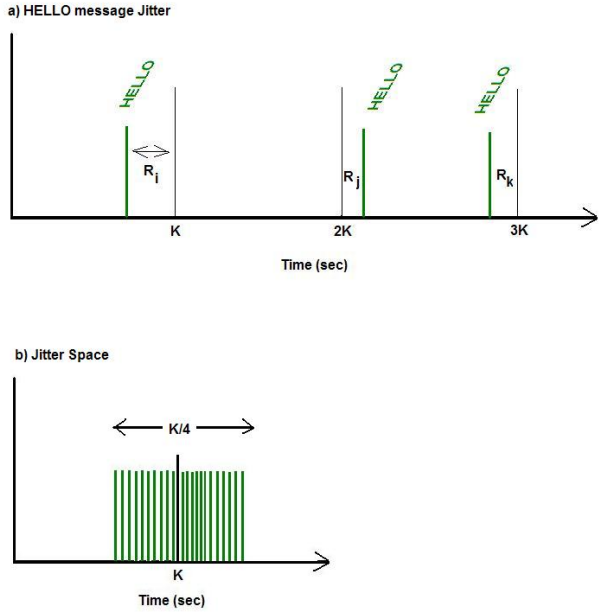

Figure 2. a) HELLO packets sent out every K seconds by OLSR, but with a random jitter (R_i,R_j, R_k…) attached to the timing of each message. b) The range of allowed jitter values.

2.1 Jitter Waveforms

The key idea first put forward in (Gorlatova et al., 2007) is to take advantage of the property of jitter in

proactive routing protocols. We do so by treating jitter as a mathematical function under our control instead of a random property of the system. As depicted in Fig. 2a), proactive protocols such as OLSR use random timing delay (jitter) applied to each periodic routing message to minimise repeated media access contention. According to the protocol, this jitter should be chosen from a uniform random distribution bounded within a range of K/4 of the base periodicity, as shown in Fig. 2b). Since there is no other constraint other than the K/4 bound, the message sender is provided with a degree of freedom within the bounds of the protocol. An implication of this is that the sending node can deliberately manipulate the timing of the protocol messages with no affect on interoperability (since the protocol remains intact) and at no cost in terms of usable bandwidth— properties that are highly desirable in security services. We will expand upon variations of this theme as this paper progresses.

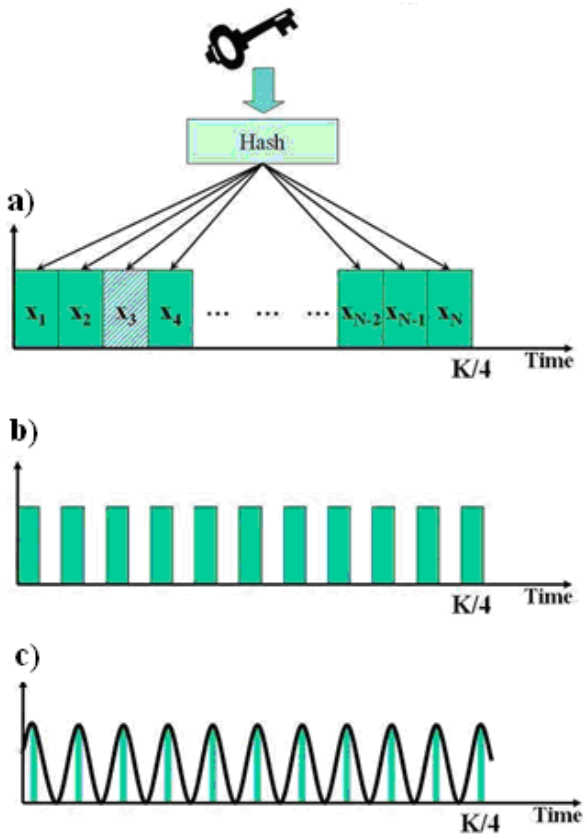

Figure 3 a) Keyed Jitter. Both sender and receiver know when jitter value x_3, for example, is to be sent. b) Partitioned Jitter. Here the jitter values are limited to discrete values (keyed or not) that could be viewed as an allowed alphabet of values. c) Partitioned Jitter viewed as the result of a uniform probability distribution overlaid with a sinusoidal probability function.

We proposed in (Gorlatova et al., 2007) that keying the jitter, as shown in Fig. 3a), would allow us to remove the greatest timing variations in the HELLO message time series used in FWAD and greatly enhance its sensitivity. Furthermore, we posited that jitter values could be restricted to discrete values, effectively partitioning "jitter space" into an alphabet of discrete values as depicted in Fig. 3b). This partitioning of jitter space is analogous to taking the uniform jitter probability function and suppressing a periodic subset of values (Fig. 3c).

2.2 Implementation

We have tested these ideas both in a simulation environment and in a laboratory testbed similar to that described in (Gorlatova et al., 2006) In both implementations, the OLSR HELLO message jitter is manipulated by a module that follows the procedure outlined in Fig. 4. Each sending node creates a unique sequence of jitter values by feeding a shared secret, its own address, and a timestamp into a sequence creation engine (we used a hash function)[2]. The values can then be further massaged using the modulation function, which in our case was either a sine wave probability suppression function, a simple discretisation (rounding or binning) of values, or both. The result is a sequence of jitter values which are indexed and appended to the HELLO message base periodicity. The receiving node can recreate the sending node's sequence and compare, by means of an analysis engine, the received jitter sequence to the expected sequence. There will, inevitably, be some noise in the received jitter values which can be dealt with either by reducing the size of the alphabet or by using statistical (and/or error correction) methods to minimise its impact. We tested our methods on a MANET scenario undergoing a wormhole attack, both in simulaton (using NS-2) and in an 802.11 testbed.

The first step was to create an OLSR jitter probability distribution similar to that shown in Fig. 5. The jitter was limited to discrete alphabet (partitioned) by simple rounding (binning) and further manipulated by suppressing the frequency of specific values according to a slowly varying (50Hz) sinusoidal probability function. The partitioning itself adds another frequency component of 200Hz to the distribution. Note that for demonstrative purposes, the figure (Fig. 5.) actually shows frequencies of 200Hz and 20Hz respectively.

2.3 Results

In the experiments, time series of HELLO message creation and reception times were constructed and Fourier transformed following the description in (Gorlatova et al.,

[2] The method of sequence creation given here is just one example of countless ways this could be done.

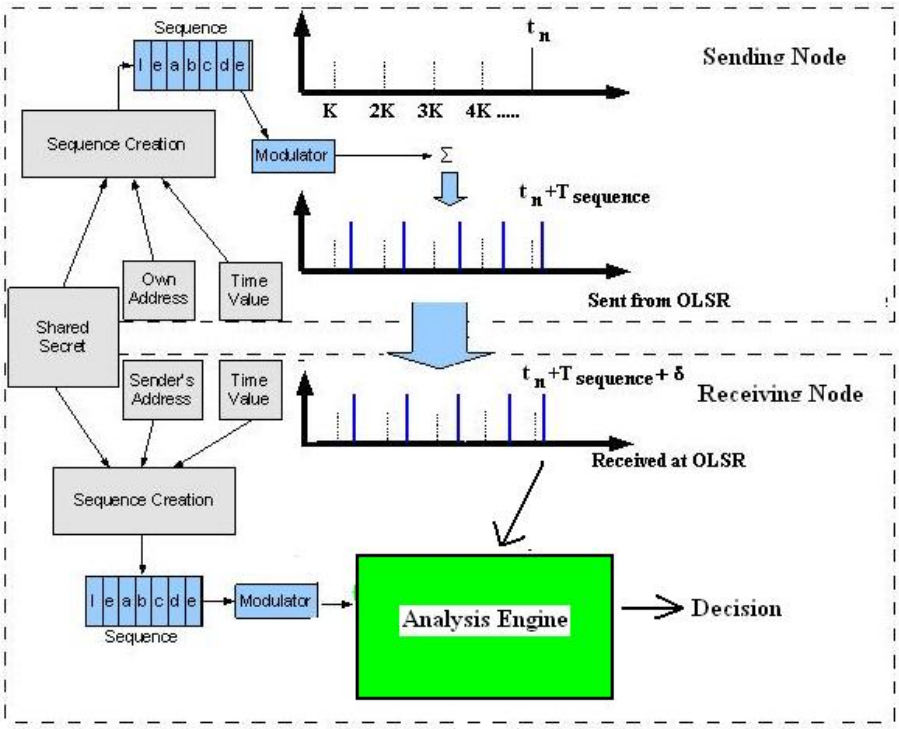

Figure 4. Schematic of the implementation of deliberate jitter into our NS-2 and Testbed environments. Both the sending and receiving nodes calculate the jitter stream which can be sent through a probability function modulator if desired. The resulting stream is transmitted by the sending nodes as jitter values associated with HELLO messages. The receiving node compares these values, which have acquired some noise δ in the process, with the expected jitter stream and feeds the data into an analysis engine for the purposes of attack detection and authentication.

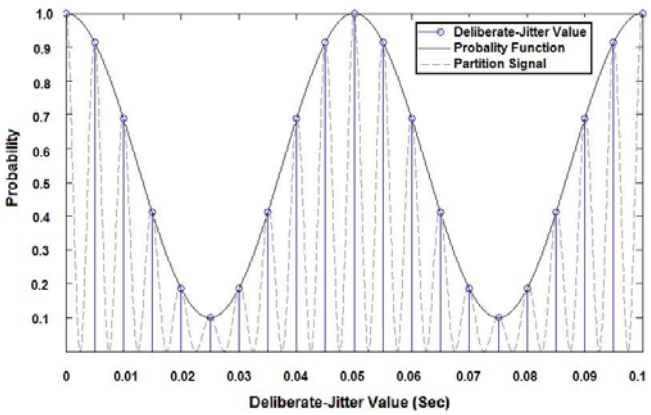

Figure 5. Combining simple partitioning with a slowly varying function that periodically suppresses the probability of selecting certain jitter values. The partitioning effectively introduces a rapidly varying alternations in the probability function (dotted lines) so there are two "signals" in play.

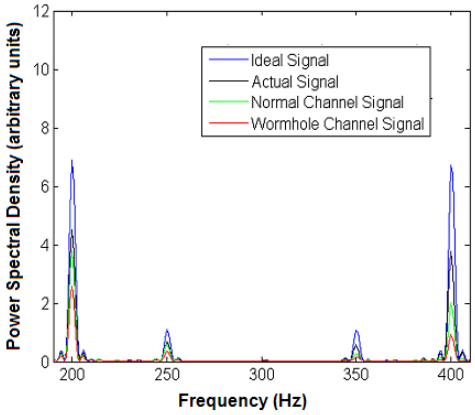

Figure 6. Power Spectral Density (PSD) of HELLO message time series with a partition frequency of 200Hz and overlaid sinusoidal probability function of frequency 50Hz. The ideal signal is calculated from theory, the actual signal is from the time series generated at the sending node. The normal channel refers to the PSD calculated at the receiving node via a valid link while the wormhole channel shows the PSD calculated at the receiving node after the HELLO messages have been tunneled through an off-channel link.

2007). The results are presented as power spectral densities (PSDs) in Fig. 6 which compares the PSDs of four different cases:

i. the series created by the OLSR jitter module at the sending node
ii. the series of actual HELLO message send times recorded at the MAC layer.
iii. the reception time series via a normal MANET link
iv. the reception time series for HELLO messages that have arrived through a wormhole.

The difference between the creation and sending series (cases i. and ii.) is caused by delays in internal processing and medium access contention at the sending station (Lynch, 2007). Fourier peaks are evident in Fig. 6 at the partitioning frequency, its harmonics, and at lobes offset from them by the frequency of the overlaid sinusoidal probability function. These peaks effectively measure the distortion in the HELLO message time series and can be used to discriminate between valid and wormhole links. To better demonstrate the differences seen in the analysis, Fig. 7 compares the Fourier peak intensity for a valid and wormhole link as a function of frequency, showing a clear difference between the links. To improve the statistics of this discriminator, we can sum the peak intensities over all harmonics to give a single value, which we refer to as the accumulated peak intensity, for each link.

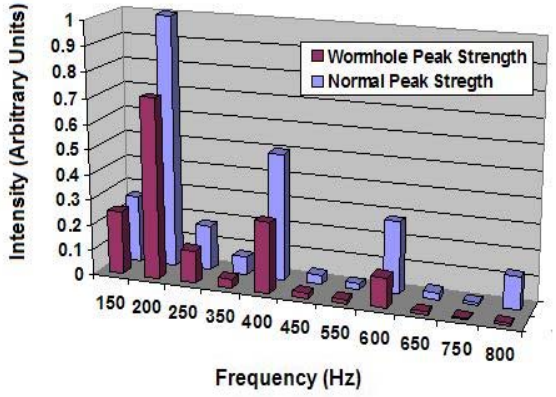

Figure 7. Testbed results of attack detection using the modified Frequency-based Wormhole Attack Detection method. Peaks are centred at the partion function frequency of 200Hz with lobes at +/- the overlaid probability frequency of 50Hz. Harmonics appear every 200 Hz.

In Fig. 8a) we show the accumulated peak intensities for the valid and wormhole links as a function of the number of HELLO packets in the time series in order to display the rapidity with which a conclusion can be reached. The error bars represent the maximum and minimum bounds seen in 15 variations of the experiment. The results show that we can conclusively discriminate between the valid link and the wormhole link in approximately two and a half minutes (150 seconds, assuming HELLO packet periodicity of 1 second). The NS-2 results (not shown) suggest that this detection time could be reduced to one minute if the OLSR code were optimised to reduce internal delays (Lynch, 2007). We emphasize again that these results are for a *dormant* wormhole, one that has not yet begun to actively disrupt network traffic. Figure 8b) shows the relative accumulated peak intensities associated with a valid link and a wormhole link under no data load and with a data transfer rate of 150 kbps between the two communicating nodes.[3] As indicated in the figure, while the peak intensities are diminished under load, the relative difference between them remains steady.

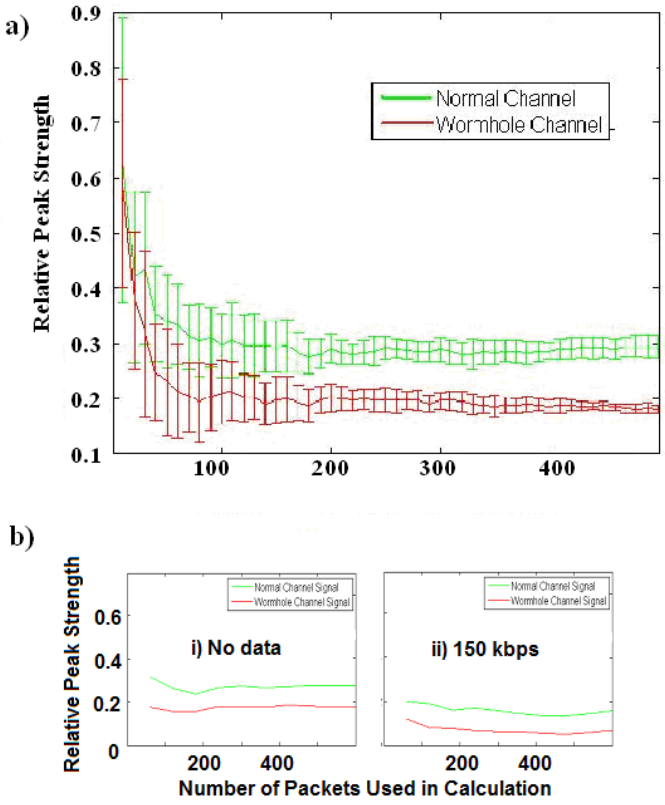

Figure 8. Testbed Results. a) Comparison of the peak strength of the signal of jitter values in normal conditions and when passed through a wormhole. The peak strength is the summation of all Fourier peaks shown in Figure 6. Panel b) shows the effect of traffic.

[3] The nodes used in the testbed are located in the same room so even in a no-load situation there is measurable contention for the medium.

3. AUTHENTICATION AND BLUE FORCE TRACKING

Our discussion has shown that it is possible to embed signals into the OLSR message protocol and use them to advantage in wormhole attack detection. Clearly, the applicability is not limited to this specific case. Similar techniques can be used for authentication purposes. To wit: a group of users could have an *a priori* agreement to embed a probability suppression signal of 200Hz overtop of their HELLO message distribution function. As shown in the previous section, this signal can be used for detecting the presence of a wormhole, but it could equally well be used as a beacon or signature validating group membership. We are certainly not suggesting that this is necessarily the most efficient way of providing a continuous authentication mechanism, merely pointing out that the possibility exists. There may be cases where an authentication alert is better sent discretely and this method facilitates a mechanism for doing so (Tang et al., 2008).

An obvious drawback of manipulating the probability function of jitter values is that an outside observer who is closely monitoring and statistically analysing the traffic of the MANET is bound to take note. This observer is less likely to notice, however, if the MANET is simply partitioning the jitter to a finite sized alphabet without the probability function overlay. In one set of our experiments, we partitioned the jitter into 200 discrete values. We chose this number because it was the maximum number of values that could be decoded with certainty on the receiving end. That is, the separation between jitter values was large enough that two different values would not be mistaken for one another given the noise present in the system. If each of these values is thought of as a character in an alphabet, this provides us with the ability to send messages, character by character, encoded as jitter values.

A simple demonstration of this would to take a message and obfuscate it by performing an XOR operation with a stream of deliberate jitter values, then using the resulting values as the new jitter keystream. The receiving node calculates the received jitter values and performs the same XOR operation with the expected jitter stream to reveal the message:

Received stream: f l y i n g s p a g h
⊕
Expected stream: e t t i m o n s t e r

Message: i a m c a n a d i a n

We used such techniques in our lab set up to do the equivalent of Blue Force Tracking— nodes that sent the expected stream were considered group members while those that did not were deemed outsiders. These techniques could be used by a nation to passively identify its own nodes in a coalition environment. It is worthwhile to note that a force could be employing this method without the knowledge of other partners since the distribution of jitter values is statistically random (within the size of the alphabet in use), so the protocol is not being compromised. Finally, if one wanted to guarantee that an eavesdropper was not able to distinguish between partitioned jitter and the protocol-specified uniform jitter distribution, a uniform continuous range could be used. The message passing techniques discussed above would still be possible but statistical and coding methods would then need to be employed to guarantee accurate resolution of the alphabet.

Given that message passing has been demonstrated using partitioned jitter, and that by moving to a continuous range of deliberate jitter values which, statistically speaking, would be indistinguishable from a random uniform distribution, an important issue arises. If a rogue node wished to leak information from within the network, it could do so by similar techniques. Now assume, however, we have keyed our jitter to produce a predictable stream. We may not be using this stream for authentication or wormhole attack detection, but as a proactive defense. If the stream of jitter values does not match our expected stream, we would have reason to believe that this node itself had taken control of the jitter distribution for illegitimate use as a side-channel. That is, we could be dealing with a rogue node leaking information from within the network.

4. CONCLUSIONS

In this paper, we have shown how we can take advantage of a previously untapped side-channel in MANETs and use it for providing security services. Our experiments show that a strategic, mathematical manipulation of the OLSR protocol's HELLO message jitter distribution function can be effective against defending against one of the most serious MANET vulnerabilities— a wormhole attack. Though we have not yet had the opportunity to investigate, other attacks, such as any variation of a replay attack, should also prove detectable with these methods.

We suggested in (Gorlatova et al., 2007) that such control could be used to greatly enhance our attack detection techniques by embedding signal information into this channel. We have now demonstrated these techniques in a lab environment and reported on their successful application. Wormhole attack detection can now be done locally in as little 150 seconds with a clear path to improving this number (Lynch, 2007). Further to

this, simple pattern-matching techniques can be employed so that the jitter keystream can be used as a continuous beacon of node identity/validity (continuous, broadcast authentication) or identifying to which nation a node belongs.

Node identification techniques similar to radio transmitter fingerprinting (Rasmussen et al., 2007, Hall et al., 2006) could instead be done by "watermarking" the message protocol, providing a flexible software solution instead of one tied to hardware. In addition, using a keyed jitter stream means that we now have an additional security function – the ability to detect the malicious use of a side channel. If a node were using a non-random, or differently keyed, stream of jitter values to leak information from within the network, we could quickly detect that this node is not using a valid jitter keystream and investigate. We stress that keying the jitter does not affect interoperability, since the protocol appears unchanged to an observer, nor is any new message overhead introduced.

REFERENCES

Gorlatova, M.A. 2006. *Review of Existing Wormhole Attack Discovery Techniques*. Contractor Report. CR-2006-165. Defence Research & Development Canada. August 2006.

Gorlatova, M.A., Kelly, M., Liscano, R, and Mason, P. C. 2007. Enhancing Frequency-based Wormhole Attack Detection with Novel Jitter Waveforms. *Proceedings of IEEE SecureCom 2007*. Nice, France. Sept. 2007. 304-309.

Gorlatova, M.A. Mason, P.C., Wang, M, Lamont, L., and Liscano, R. 2006. Detecting Wormhole Attacks in Mobile Ad Hoc Networks through Protocol Breaking and Packet Timing Analysis. *Proc. Of IEEE MilCom 2006*. October 2006. Washington D.C.

Hall, J., Barbeau, M. and Kranakis, E. 2006, Detection of Rogue Devices in Bluetooth Networks using Radio Frequency Fingerprinting, *IASTED International Conference on Communications and Computer Networks (CCN 2006)*, October 2006. Lima, Peru.

Hu, Y.C. and Perrig, A. 2001, *Packet Leashes: A Defense Against Wormhole Attacks in Wireless Networks*. Technical Report TR01-384 Rice University.

Lynch, D. 2007. *Wormhole Detection Through the Manipulation of Periodic Messages in the OLSR Protocol.* M.App. Sc. Thesis, Royal Military College of Canada.

Mason, P.C., Salmanian, M., Tang, H., and Seguin, D. 2007. *Security Issues in Dynamic Reconfigurable Ad hoc Networks*. The Technical Cooperation Program C3I Group Panel 11, Technical Report TR-C3I-TP11-1-2007.

Nait-Abdesselam, F., Bensaou, B., and Taleb, T. 2008. Detecting and Avoiding Wormhole Attack in Wireless Ad Hoc Networks. IEEE Communications Magazine. 127-133.

Nguyen, D-Q. and Lamont, L. 2008. A Simple and Efficient Detection of Wormhole Attacks. *2nd IEEE International Conference on New Technologies, Mobility and Security.* Tangier, Morocco, Nov. 2008.

Rasmussen, K.B., and Capkun, S. 2007. Implications of Radio Fingerprinting on the Security of Sensor Networks. *Proc. of IEEE SecureCom 2007*. Sept. 2007. Nice, France. 331-340.

Stern, D., Lawler, G., Gopaul, R., Marcus, K., and Kruus, P. 2007. Countering False Accusations and Collusion in the Detection of In-Band Wormholes. *Twenty-third Annual Computer Security Applications Conference, 2007 (ACSAC 2007)*. Dec. 2007. 243-256.

Tang, H., and Salmanian, M. 2008. Lightweight Integrated Authentication Protocol for Tactical MANETs, *Proc. IEEE TrustCom '08*. Zhangjiajie, China.