Steganographic Internet of Things: Graph Topology Timing Channels

Ira S. Moskowitz
Information Technology Division, Code 5580
Naval Research Laboratory
Washington, DC 20375
ira.moskowitz@nrl.navy.mil

Stephen Russell, Brian Jalaian
Battlefield Information Processing Branch
Army Research Laboratory
Adelphi, MD 20783-1197
stephen.m.russell8.civ@mail.mil
brian.jalaian.ctr@mail.mil

Abstract
Given the self-aware, artificially intelligent, and complex system-of-systems nature of the Internet of Things (IoT), unintended behavior will manifest itself in many forms. In this paper, we illustrate a method for steganographic messaging that can exploit IoT side channels and be resilient to the heterogeneous communications and application protocols that exist in the IoT. We show that IoT side channels are susceptible to network steganography. Moreover, it is possible to create a data-in-motion steganographic method without network protocol modifications and mathematically bound the channel capacity.

Introduction
The Internet of Things (IoT) is the realization of interconnected and ubiquitous computing, pervasive sensing, and autonomic systems that can affect the physical world. Some argue the promise of IoT technology represents the beginning of a transcendental shift in humans’ interaction with technology (Gubbi et al. 2013). The “things” that exist in the IoT can be generally thought of as physical or computational objects that label, sense, communicate, process, or actuate thereby bridging the physical and virtual worlds (Oriwoh and Conrad 2015; Pande and Padwalkar 2014). While there is no universally accepted definition of the IoT, the International Telecommunication Union Telecommunication Standardization Sector (ITU-T) defines the IoT as “a global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things... through the exploitation of identification, data capture, processing and communications capabilities, the IoT makes full use of things to offer services to all kinds of applications, whilst maintaining the required privacy (ITU-T ). “ The inclusion of privacy concerns in the ITU-T’s standard definition points to an implication that suggests significant security challenges in the IoT. Like any technology that impacts the physical world through extensions in cyber space, the scope and quantity of potential threat vectors increases significantly over those technologies that keep the domains separate. Given its pervasive nature, security in the IoT is an immensely challenging, but well known and documented, problem (Zhao and Ge 2013; Hwang 2015; Li and Da Xu 2017).

Security, and by implication deception, is a more significant concern when artificially intelligent autonomous and autonomic system behaviors are considered. Both of these characteristics are intrinsic “features” of the IoT and likely to increase in sophistication and complexity as technology matures, leading to greater system self-awareness. The notion of self-awareness in the IoT has led to a new term for characterizing the IoT: self-star (self-*.). Self-* encompasses a wide range of behaviors that extends the concept of autonomic computing to the ubiquity and invasiveness of the IoT. The term self-* covers several aspects of system-self, such as self-configuring, self-organizing, self-healing, self-adapting, and self-protecting (Berns and Ghosh 2009). As the devices in the IoT manifest greater self-* properties, it becomes increasingly unrealistic to expect human operators to maintain control over IoT system dynamics. Threats from malignant human IoT objectives or errant intelligent autonomic behaviors will progressively be the norm and present challenging latent security risks that are susceptible to manipulation.

Researchers and practitioners cannot ignore security problems related to technology and are actively addressing IoT security headlong, through defensive mechanisms such as differential-encryption, soft-biometrics, and decentralized-ledger approaches, amongst others (Her-ermann, Muhl, and Geihs 2005; Dorri et al. 2017; Dinca and Hancke 2017). However, passive intrinsic security threats are often overlooked. IoT devices typically operate on a primary communications channel and require a side channel for inter-process communications, control, and/or configuration. These side channels create another vulnerability, due to their usage for direct configuration or operations or self-* behaviors. In this manner, the scope and scale of the IoT will offer a tremendous platform for network steganography. Steganographic methods hide information, thereby making the information difficult to notice by embedding it in an information carrier. The distinction between steganography, network steganography in particular, and “covert channels” (communication paths that were neither designed nor intended for information transfer) is not well grounded (Lubacz, Mazurczyk, and Szczypiorski 2014). We apply the term network steganography because
IoT side channels natively operate as a secondary channel for telemetry or other control data rather than the primary channel for sensing or actuation information.

The threat of covert communication is a very real threat, and is well documented in the literature (for example (Moskowitz and Kang 1994)). Because of its scale, heterogeneity, and ubiquity the IoT will be highly susceptible to exploitations of covert communication channels. Moreover, the IoT (as a system) will have a complexity beyond human comprehension, possess an ability to affect the physical world, and will have sense of “self” manifesting autonomic behaviors imbued with artificially intelligent (AI) reasoning. Embedded in IoT devices themselves, the same AI capabilities that acquire, aggregate, and analyze other IoT actuators and data sources to assure provenance and veracity of suspect IoT cyber data, will enable self-protecting (adversarial or otherwise), obfuscating, and deceptive “thing-behaviors” that exploit intrinsic side channels and employ network steganography to avoid detection.

In this paper, we illustrate a method for steganographic messaging that can exploit IoT side channels and be resilient to the heterogeneous communications and application protocols that exist in the IoT network. We show that IoT side channels are susceptible to network steganography. Moreover, it is possible to create a data-in-motion steganographic method without network protocol modifications and mathematically bound the channel capacity. The paper is organized as follows. The next section provides a background on IoT side channels, IoT self-* properties, and network steganography. This section is followed by the description of the side channel timing covert channel and ring topology discussion. We conclude with a summary and a brief discussion of countermeasures.

**Background**

The modern IoT is, and will continue to be, comprised of sensing and actuation devices that operate as a single system (though distributed and compose-able), have embedded artificial intelligence with continuous learning capabilities, and have “self awareness.” A hallmark characteristic of IoT composite systems is self-star (self-*) behaviors, such as self-organizing, self-configuring, self-healing, self-protecting, etc., often referred to self-* properties or characteristics (Berns and Ghosh 2009; Kishore Ramakrishnan, Preuveneers, and Berbers 2014). The underlying notion behind self-* characteristics mandate an understanding of state, context, and environment, so that a device’s behavioral adaptation lead to recovery from arbitrary transient conditions or changes to an initial state. The basis for self-* relate to traditional topics of control theory and because IoT devices do not exist purely in isolation they typically utilize, or require, communication channels in which to transmit and receive messages to/from users and other devices/systems (Athreya, DeBruhl, and Tague 2013). Moreover, having awareness of context, IoT devices must assess the surrounding environment and use the assessment to best accomplish their current goals (Sicari et al. 2015). Assessing a device’s state and the environment in which it is operating mandates some manner of learning. Traditional incremental learning techniques can be used to address exposure to new observations for assessment. However incremental learning generally does not capture relationships between context and operational requirements. Therefore, continuous learning, a key enabler of self-* behaviors in IoT, will only advance in sophistication as artificial intelligence technologies are improved. Further, IoT devices are components in complex system-of-systems that are imbued with artificial intelligence to collectively achieve user goals.

To accomplish continuous learning and self-* behaviors many IoT devices and systems commonly utilize control channels or control messaging (Kim and Kim 2005), (Heath et al. 2008), (Kim and He 2015). Control or side channels are those communication pathways used for system functions, whether they be heartbeats, internal monitoring/signaling, self-awareness mechanisms, or other system-centric operations. The existence and nature of side channels creates an additional vector that can be exploited for both benign and malicious purposes (Ronen et al. 2017). From the perspective of security, passive observation of side channels, like primary channels, can lead to the leakage of information that may classify or even uniquely identify IoT devices, i.e. watermarking (Chhetri, Faezi, and Al Furouque 2017). There are methods for detecting the potential leakage from side channel messaging. For example, Coron et al. (Coron, Naccache, and Kocher 2004) developed statistical tests that can detect the presence of side channel leakage from cryptographic computations. Nonetheless, detection of leakage does not necessarily prevent it. Further, unlike physical information leakage, such as those resulting from electro-magnetic emissions, leakage that is embedded in the operational information itself can represent a particularly difficult security concern. In the context of this research, the exploitation of a side channel should not be confused with a side channel attack that, while related to information leakage, requires an encryption device or mechanism. Our focus is on the exploitation of side channels for covert communication, regardless of whether that communication is desirable or undesirable.

Much like primary communication channels, side channels can be used for covert communication. Originally coined by Szczypiorski (Szczypiorski 2003), the term network steganography refers to the use of telecommunications medium or protocols to conceal messages between a sender and a receiver. Contrary to typical steganographic methods that use digital media (e.g. pictures, audio, video files, etc.) as the means in which to embed hidden data, network steganography encode content in communication protocols control elements or intrinsic functionality. There are three generally accepted classifications of network steganographic methods: 1) protocol storage, 2) protocol timing, and 3) application protocol header modifications (Lubacz, Mazurczyk, and Szczypiorski 2014). Each of these methods take advantage of network communications protocols to exploit covert data channels and some methods are more efficient and robust than others (Collins and Agaia 2016). In the following sections we present a method for network steganography using IoT side channels and illustrate the method through mathematical examples.
A Timing Network Stego Side Channel

Given the self-aware, artificially intelligent, and complex system-of-systems nature of the IoT, threats will manifest themselves in many forms. The existence of side channels presents an exploit that can be utilized to hide information. In the context of network steganography it is not necessarily important to highlight the purpose of the covert communications, as it could be used for watermarking, benign inter-device self-* enabling behaviors, malicious network subversion, or other objectives. As a simple example, consider how a spy might utilize “normal” communications to embed a message to signal the start of a broader attack, or trigger the activation of a malicious activity. Now consider how the same spy might use the ubiquitous network of smart IoT devices to do the same thing and avoid detection. Extending this example, note that IoT devices are capable of actuating the real world and, unlike modern computers, will exist in the network at unprecedented scale and availability. From just this simple pragmatic example, it becomes clear that network steganography takes on a new cyber priority in the IoT. This priority becomes more significant when AI-enabled self-* behaviors create situations that could lead to the devices themselves employing covert messages (obscured from human users) for their own means.

The problem of covert messaging, independent of the transmission protocols, in IoT side channels boils down to: how can a sender in a network, described by a graph, covertly send information to a receiver, via the time that the message gets there? When framed in this manner, a steganographic threat can be considered as an IoT adaption of covert timing channels (Moskowitz, Greenwald, and Kang 1998; Martin and Moskowitz 2006; Moskowitz and Miller 1992). Consider a network of devices that are communicating with each other. Further, it is the desire for one transmitter node \( t_1 \) to covertly communicate with another receiver node \( n_R \) by manipulating the cover/legitimate traffic that it sends to the receiver node.

We make some basic assumptions that generalize to Internet communications. Moreover these assumptions span most commonly implemented IoT network protocols including Bluetooth and many IoT specific protocols, such as Z-wave, ZigBee, WiMo, and Insteon (Raglin et al. 2017).

1. ASSUMPTION: \( n_R \) addresses the message to \( n_R \), but it only has control over the first link it leaves \( n_{TR} \) on.

2. ASSUMPTION: The path that the message travels is a simple path; that is, the message never goes through the same node twice. Thus, when we use the term path it is understood to be a simple path.

To illustrate, we make another reasonable assumption: the sender and the receiver have synchronized clocks. The sender starts at \( t = 0 \) if it sends a time symbol \( t_1 \) it waits until it has been received at time \( t_2 \) and then sends the next symbol, and waits the amount of time of that symbol, and then sends the next. It is the inter-symbol gap time that is the information. The method we propose does not require any time stamping, or sequence numbering of the messages. The inter-symbol gap time is received noisely by the receiver. The symbols are \( t_1 < t_2 < \cdots < t_K, K \geq 2 \).

![Figure 1: Transmission of \( t_1, t_2, t_1, t_2 \), where \( t_1 = 1, t_2 = 2 \).](image)

![Figure 2: 3-clique, also 3-ring](image)

**Simple example: 3-ring \( R_3 \)**

Imagine we have a very simple 3-node graph given by the following adjacency matrix \( A_3 \)

\[
A_3 = \begin{pmatrix}
A & B & C \\
0 & 1 & 1 \\
1 & 1 & 0 \\
\end{pmatrix}
\]  

There are two paths (see Fig. 1) from \( A \) to \( C \), they are \( A \rightarrow C \), and \( A \rightarrow B \rightarrow C \). To simplify notation we write this as \( \langle A(C) \rangle \), and \( \langle A[B(C) \rangle \). Let us assume that the time to travel each link is \( t \), where \( t \) is the unit of time (a tick). We often suppress the \( t \) (normalize it to 1) and write \( n \) instead of \( n \cdot t \) for the sake of simplicity. Let \( T \) denote the time for a cover message to travel from \( A \) to \( C \), we have that \( T(A(C) = 1 \), and \( T(A[B(C)] = 2 \). By using the inter-symbol gap time \( C \) knows whether the trip took 1tick or 2ticks. Also, by our assumptions, \( A \) is allowed to pick the path that the cover message travels on. Keep in mind the real information we are interested in is \( T \), not the content of the cover message. The time that it takes for the cover message to arrive at \( C \) is the **covert** steganographic communication.

That is in terms of the cover channel \( \langle A|C \rangle \), and \( \langle A|B|C \rangle \) are the same, but the fact that this message takes either 1 or 2 ticks \( t \) is what forms the stego side channel.

What we presently have is a timing channel with input binary \( X \) and output binary \( T \). Using Verdù's result (Verdú 1990, Eq. 1), (Moskowitz, Greenwald, and Kang 1998) we have the following formula for capacity in bits per unit time \( C_t \) where \( X \) is the input to the channel, and the random variable \( T \) is the output random variable which is the cost in time as a function of the random variable \( X \). First we define the mutual information in unit of bits per unit time \( I_t \), and then \( C_t \)

\[
I_t = \frac{I(X,T)}{E(T)}, \text{ where } I(X,T) = H(X) - H(X|T) = I(T,X). 
\]  

\[
C_t = \sup_{X} I_t 
\]  

\( T \) takes on the values 1 or 2, however we generalize this to \( t_1, t_2, t_1 < t_2 \) for the sake of generality. This is a **noiseless** binary input, binary output, timing channel (Moskowitz...
and Miller 1994). That is the channel matrix is the $2 \times 2$ identity matrix. Hence $p(x_1) = p(t_1)$ and thus the conditional entropies $H(T|X)$ and $H(X|T)$ are zero (the conditional probabilities are 0 or 1, and by definition, $0 \log 0 = 1 \log 1 = 0$). We abuse notation and let $p = p(x_1)$ and $1 - p = p(x_2)$. Eq. (2) reduces to

$$C_t = \max_p \left( \frac{H(X)}{p_{t_1} + (1 - p)_{t_2}} \right)$$

(4)

$$C_t = \max_p \left( \frac{-p \log(p) - (1 - p) \log(1 - p)}{p_{t_1} + (1 - p)_{t_2}} \right)$$

(5)

**Theorem 1.** (Shannon 1948, Appendix 4), (Krause 1962), (Moskowitz and Miller 1994, Thm. 2), (Khan- 
dekar, McEliece, and Rodemich 2000) The capacity, in units of bits per $t$, of the channel in Eq. (5) is $C_t = \log \omega$, where $\omega$ is the unique positive root of the characteristic polynomial

$$\chi(x) := 1 - \sum_{i=t_1,t_2} x^{-i}, x \geq 0.$$  

Furthermore, the distribution on $X$ that achieves capacity is $p_c = \omega^{-t_1}$, where $p_c = P(X = t_1)$, which is equivalent to $1 - p_c = \omega^{-t_2}$.

Proof: We only sketch the proof since it is well referenced in the literature. We note that it generalizes to more than two symbols (see the later theorem in this paper). In short there are two equivalent ways of defining the capacity. The algebraic way is via saying that capacity is the limit superior of the number of different symbols that can be passed per unit time. The other way is as the maximum of the ratio of the mutual information, which in the noiseless case is the identity matrix. Hence $\omega^1 = 0$.

Furthermore, the distribution on $X$ that achieves capacity is $p_c = \omega^{-t_1}$, where $p_c = P(X = t_1)$, which is equivalent to $1 - p_c = \omega^{-t_2}$. Eq. (2) reduces to

$$C_t = \max_p \left( \frac{-\omega^{-t_1} \log(\omega^{-t_1}) - \omega^{-t_2} \log(\omega^{-t_2})}{\omega^{-t_1} \cdot t_1 + \omega^{-t_2} \cdot t_2} \right)$$

(14)

$$C_t = \left( \frac{t_1 \omega^{-t_1} \log(\omega) + t_2 \omega^{-t_2} \log(\omega)}{\omega^{-t_1} \cdot t_1 + \omega^{-t_2} \cdot t_2} \right) = \log \omega$$  

(15)

Note, it is often computationally easier to express the characteristic polynomial with positive coefficients and find its root. For our example from Fig. 1 we find that the positive solution of

$$x^2 - (x + 1) = 0$$

is $\frac{1 + \sqrt{5}}{2}$. Thus, we have that $C_t = \log \frac{1 + \sqrt{5}}{2} \approx .694$ bits/t and the critical probability $p_c = \frac{1 - \sqrt{5}}{2} \approx .618$.

Note that in the 3-ring in Fig. 1 there are always two paths between any two nodes, the shortest of length 1, and the longest of length 2. Therefore, $A$ can communicate (via the method described) to $C$ with a capacity of .694 bits/t. Note there is nothing special about $A$ and $C$ due to the graph symmetry, so this result holds for any two nodes in the 3-ring.

**Simple example extended to the N-ring $R_N$ — only use nodes 1 and N**

Let us consider a generalization of the 3-ring, that is we will look at an $N$-ring (see Fig. 3). Without loss of generality, we use a clockwise cyclic numbering of the nodes, 1 through $N$, and pick one particular node as the first node $v_1$ (as denoted in Fig. 3). We note that the simple ring structure and how nodes may be compromised has been well-studied in terms of rewiring (Hyden et al. 2017).

Now say that $v_1$ wishes to open a covert side channel to $v_N$. A message can be sent the short way $\langle v_1 | v_N \rangle$ or the long way $\langle v_1 | v_2 | v_3 | \cdots | v_N \rangle$. So now a symbol is either $t_1 = 1$ or $t_2 = N - 1$. From above, the capacity is, in units of bits per $t$,

$$C_t = \log \omega_1,$$

where $\omega$ is the unique positive root of the characteristic polynomial

$$\chi(x) := 1 - \sum_{i=1,N-1} x^{-i}, x \geq 0.$$  

(17)

Furthermore, the distribution on $X$ that achieves capacity is $p_c = \omega^{-1}$, thus $1 - p_c = \omega^{-(N-1)}$.  

As noted above, it is often computationally easier to express the characteristic polynomial with positive coefficients and find its root, that is find the positive solution of
\[ x^{N-1} - x^{N-2} - 1 = 0. \quad (18) \]
We see that as \( N \) grows large, \( C_t \) inversely approaches zero (for \( N = 101, C_t = .05 \), for \( N = 701, C_t = .01 \), and for \( N = 10,001, C_t = .001 \)).

**What if we use an intermediate node instead of the final node?**

For \( R_3 \) there are no intermediate nodes, however for \( N > 3 \) there are. For example, let \( N = 11 \) in the ring topology, which set-up gives us the highest capacity? Since \( t_2 > t_1 \) (nothing is lost by this, we are just flipping the short way with the long way) we see that \( t_2 = 6, 7, 8, 9, 10 \). Please see Table 1 for clarification.

For \( R_{11} \) we see that we achieve the maximum capacity when the difference between \( t_1 \) and \( t_2 \) is maximized. Can we prove this in general? First, let us look at an even case \( N = 10, R_{10} \) in Table 2.

The reason that the last row is different is that our symbols are the time values, if the time values are the same then the capacity must be zero. If we send different storage symbols at different times then we could the table giving \( p_c = 1/2, H(X) = 1, E(T) = 5, C_t = .2 \) bpt. Sending different storage symbols does not affect the other calculations.

**Theorem 2.** Given, \( R_N, N \geq 3 \) we let \( C_t(1, j) \) denote the covert channel timing capacity using nodes \( v_1, v_j \) (because there is no loss of generality we can always take the first node to be \( v_1 \)).\(^2\) \( C_t(1, j) \) is a monotonic decreasing function of \( j \) as

\[
    j \rightarrow \left\{ \begin{array}{ll}
        [N/2] + 1, & \text{for } N \text{ odd} \\
        N/2 + 2, & \text{for } N \text{ even}.
    \end{array} \right.
\]

**Proof.** We see that \( t_1 = N - j + 1, t_2 = j - 1, \leq t_1 < t_2 \leq N - 1 \). We extend from integers to real numbers and let \( a = t_1, b = N - a = t_2 \). We can write the characteristic polynomial \( \chi(x) \) as \( 1 - (x^{-b} + x^{b-N}) \), with \( x \in (0, 1) \), since \( 0 < C_t = \log \omega < 1 \). The root \( \omega \) solves \( 1 = x^{-b} + x^{b-N} \). For \( x \) fixed we will show that \( f(b) := x^{-b} + x^{b-N} \) is an increasing function of \( b \) for \( b > N/2 \). Therefore, as \( b \) decreases to \( N/2 \) so will \( \omega \) the intersection point with the line \( y = 1 \), and we will be done. (see Fig. 4) Note, since \( b \) is the bigger time value, as noted above, we have that \( b > N/2 \).

To show that \( f(b) \) is an increasing function of \( b \) we fix \( x \) at any value greater than 1 (the region of interest).

\[
    f(b) = x^{-b} + x^{b-N} \quad (19)
\]
\[
    = e^{-b} \ln(x) + e^{(b-N)} \ln(x) \quad (20)
\]
\[
    = e^{-b} \ln(x) - e^{(b-N)} \ln(x) \quad (21)
\]
\[
    df \over db \quad (22)
\]
\[
    = \ln(x) \cdot \left( e^{(b-N)} - e^{-b} \ln(x) \right) \quad (23)
\]
\[
    = \ln(x) \cdot (x^{b-N} - x^{-b}) \quad (24)
\]
\[
    \text{since } b > a \Rightarrow b = a + d, d > 0 \quad (25)
\]
\[
    = \ln(x) \cdot (x^a - x^{-b}) \quad (26)
\]
\[
    > 0 \quad (27)
\]

Thus, we have shown for the \( N \)-ring \( R_N \) that the best capacity is achieved for \( t_1 = 1, t_2 = N - 1 \), with the nodes, up to rotation, being \( v_1 \) and \( v_N \). Of course we have only been looking at noiseless binary channels timing channels (binary input, binary output, no noise) so far. This is because for a ring there are at most two paths of different lengths between any two vertices. We have not considered noise in the channel. See Table 3.

**Discussion of Simple IoT Topology**

For the \( N \)-ring our assumptions are straightforward. Assumption 1 is that the sender node can determine whether the message goes to the “left of right.” The second assumption is simply that the message does not double-back upon itself. Of course, for more complicated topologies Assumption 2 requires additional considerations. For theoretical clarity, we kept the assumptions here to a minimum. The result for the \( N \)-ring is that the covert timing channel capacity is obtained by using a sender node, and a receiver node 1-hop/\((N-1)\) hops away from it. This example was simplified to show the theoretical steganographic method. In this example we show the optimal maximization of capacity based on the topology for this ring structure. For more complicated topologies is unclear what the maximum will be and may represent a computationally intractable problem. In the next section we randomize and use expected capacity (e.g. a mean field approach) and illustrate the steganographic approach. This mean field method, may be more applicable for more complicated topologies that will be more common representations in IoT networks.

**Randomized IoT Communication and Expected Capacity**

Now let us analyze the situation where the transmitting node \( n_T \) does not pick the node it transmits to. If it can pick, and
Pepe’s the capacities reversing the order.

$\sum_{i=1}^{N} \omega_i = \sum_{i=1}^{N} \sum_{i=1}^{N} \omega_i \equiv \omega_i$.

The advance of the Internet of Things will revolutionize virtualized interactions with the physical world. In doing so, interconnected devices that form self-aware complex system-of-systems purposed to human objectives will have artificially intelligent (AI) behaviors and communicate between

\begin{table}[h]
\centering
\small\begin{tabular}{|c|c|c|c|c|c|}
\hline
Paths & Time Values & $\omega$ & $p_c$ & $H(\lambda)$ & $E(T)$ \tabularnewline \hline
$(v_1, v_2)$ & $t_2 = 10, t_1 = 1$ & 1.1975 & .8351 & .6460 & 2.4843 \tabularnewline
$(v_1, v_2)$ & $t_2 = 9, t_1 = 2$ & 1.1619 & .7408 & .8255 & 3.8144 \tabularnewline
$(v_1, v_2)$ & $t_2 = 8, t_1 = 4$ & 1.1461 & .6642 & .9208 & 4.6791 \tabularnewline
$(v_1, v_2)$ & $t_2 = 4, t_1 = 4$ & 1.1382 & .5989 & .9713 & 5.2124 \tabularnewline
$(v_1, v_2)$ & $t_2 = 6, t_1 = 5$ & 1.1347 & .5316 & .9971 & 5.4684 \tabularnewline
\hline
\end{tabular}
\caption{$N = 11$}
\end{table}

\begin{table}[h]
\centering
\small\begin{tabular}{|c|c|c|c|c|c|}
\hline
Paths & Time Values & $\omega$ & $p_c$ & $H(\lambda)$ & $E(T)$ \tabularnewline \hline
$(v_1, v_2)$ & $t_2 = 9, t_1 = 1$ & 1.2132 & .8243 & .6706 & 2.4056 \tabularnewline
$(v_1, v_2)$ & $t_2 = 8, t_1 = 2$ & 1.1749 & .7245 & .8493 & 3.6531 \tabularnewline
$(v_1, v_2)$ & $t_2 = 7, t_1 = 3$ & 1.1586 & .6431 & .9401 & 4.4278 \tabularnewline
$(v_1, v_2)$ & $t_2 = 6, t_1 = 4$ & 1.1510 & .5699 & .9859 & 4.8603 \tabularnewline
$(v_1, v_2)$ & $t_2 = 5, t_1 = 5$ & 1.1487 & .5048 & NA & 5.0286 \tabularnewline
\hline
\end{tabular}
\caption{$N = 10$}
\end{table}

\begin{table}[h]
\centering
\small\begin{tabular}{|c|c|}
\hline
$N$ & $C_t$ \tabularnewline \hline
3 & .6942 bpt \tabularnewline
5 & .4950 bpt \tabularnewline
10 & .2786 bpt \tabularnewline
20 & .1978 bpt \tabularnewline
50 & .0844 bpt \tabularnewline
100 & .0494 bpt \tabularnewline
\hline
\end{tabular}
\caption{$N$ Ring, $t_1 = 1, t_2 = N - 1$}
\end{table}

the criteria is maximizing capacity it would pick, if, without loss of generality, $n_{tr} = n_1$, then $n_R = n_N$ (same as using $n_2$) as discussed above. So now, given $R_N$ two nodes are chosen at random and we wish to determine the possible threat, as measure by the covert timing channel capacity between the two nodes as discussed above.

Let us chose two random nodes on $R_N$. The first node is randomly chosen, and without loss of generality, can be taken as $v_1$. Now, the probability that the next node is $v_j$, $j = 2, 3, ..., N$ is $\frac{1}{N - 1}$. Let us consider the parity of $N$.

$N$ Even: This follows on what we discussed above. The first node is $v_1$, with probability $\frac{1}{N - 1}$ the next randomly chosen node is $v_N$, which is a noiseless timing channel with times 1 and $N - 1$, which we denote as $C_t(1, N - 1)$, the next randomly chosen node can be taken as $v_{N-1}$ which similarly give us $C_t(2, N - 2)$. This analysis continues in this descending order until the second randomly chosen node is $\lceil N/2 \rceil + 1$, which gives us $C_t(\lfloor N/2 \rfloor, \lceil N/2 \rceil)$. It then repeats the capacities reversing the order.

Example: $N = 5$
$v_1, v_5$ results in $C_t(1, 4)$
$v_1, v_2$ results in $C_t(2, 3), 4 = \lceil 5/2 \rceil + 1$
$v_1, v_3$ results in $C_t(3, 2)$
$v_1, v_4$ results in $C_t(4, 2)$

Thus, for $N$ odd ($o$), we see that the average capacity $\overline{C}_t(N^o)$ is

$$\overline{C}_t(N^o) = \frac{2}{N - 1} \sum_{i=1}^{\lfloor N/2 \rfloor} C_t(i, N - i) \quad (28)$$

Which we may also express as

$$\overline{C}_t(N^o) = \frac{2}{N - 1} \sum_{i=1}^{\lfloor N/2 \rfloor} \log \omega_i, \quad (29)$$

where $\omega_i$ is the unique positive root of $1 - (x^{-(N-i)} + x^{-1})$. Which of course is,

$$\overline{C}_t(N^o) = \frac{2}{N - 1} \log \left( \prod_{i=1}^{\lfloor N/2 \rfloor} \omega_i \right). \quad (30)$$

$N$ Even: This case is slightly different because when we consider the nodes $v_1$ and $v_{N+1}$ we see that $t_1 = t_2 = N/2$. Hence the capacity is zero. However, it must be counted in average, which only manifests itself in dividing by $N - 1$.

Thus, for $N$ even ($e$), we see that the average capacity $\overline{C}_t(N^e)$ is

$$\overline{C}_t(N^e) = \frac{2}{N - 1} \sum_{i=1}^{\lceil N/2 \rceil} C_t(i, N - i) \quad (31)$$

Which of course is,

$$\overline{C}_t(N^e) = \frac{2}{N - 1} \log \left( \prod_{i=1}^{\lceil N/2 \rceil} \omega_i \right). \quad (32)$$

Where $\omega_i$ is the unique positive root of $1 - (x^{-(N-i)} + x^{-1})$.

In Fig. 5 we that the lowest curve is $\overline{C}_t(R_N)$, and the top plot is $\overline{C}_t(1, N - 1)$ for $R_N$.

Conclusion
The advance of the Internet of Things will revolutionize virtualized interactions with the physical world. In doing so, interconnected devices that form self-aware complex system-of-systems purposed to human objectives will have artificially intelligent (AI) behaviors and communicate between

257
themselves and with end-users along multiple, likely dynamic, communications channels. These AI capabilities also introduce a greater chance of side channel use for the purposes of covert messaging or network steganography, whether malicious or benign.

We illustrated a method for steganographic messaging that uses transmission timing to obscure symbols. We presented several examples in ring network configurations, given varying initial hop conditions. Through these formulations, we showed that IoT side channels are susceptible to network steganography and that it is possible to create a data-in-motion steganography method without network protocol modifications. While the examples do not account for empirical concerns such as noise, the described methods certainly would have implications for device network-marking methods, as well as theoretical covert channel risk quantification and threat countermeasures.

While the proposed method may appear general for networks and not necessarily IoT specific, it is the complications introduced by the autonomic nature, scope, and reach of IoT that takes the issue of network-steganography to another degree of importance. The significance of the protocol-agnostic method described in this paper presents a clear picture of the new challenges to the already difficult issue of providing network and information security. Moreover, the introduction of machine learning and AI capabilities, given a pervasive IoT with cyber-reach into the real-world, mandates a much deeper understanding of the susceptibility of communication protocols to all types of manipulation not just for automated learning and improved system resilience. Additional research in the area of network steganography may be helpful in this respect. This research paper is intended to form the groundwork for future research in the domain of IoT side channel network steganography. In future work we will consider more complicated IoT network topologies. Further, as this work is just a first theoretical step, we plan to implement the described protocol-agnostic network steganography method in an empirical IoT environment to elicit experimental results and explore the implications.

Acknowledgements

We thank the reviewers for their helpful comments and suggestions.

References


Figure 5: Plot of the average (blue, lower) capacity against best (green, upper, smoother) capacity.


Li, S., and Da Xu, L. 2017. Securing the Internet of Things. Syngress.


259