This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2008 proceedings.

iLOC: An *i*nvisible *LOC*alization Attack to Internet Threat Monitoring Systems

Xun Wang, Wei Yu, Xinwen Fu, Dong Xuan and Wei Zhao

***Abstract*—In this paper, we study a new class of attacks, the *i*nvisible *LOC*alization (*iLOC*) attack, which can accurately and invisibly localize monitors of Internet threat monitoring (ITM) systems, a class of widely deployed facilities to characterize Internet threats, such as worm propagation, denial-of-service (DoS) attacks. In the *iLOC* attack, the attacker launches low-rate port-scan traffic, encoded with a selected *pseudo-noise code* (PN-code), to targeted networks. While the secret PN-code is invisible to others, the attacker can accurately determine the existence of monitors in the targeted networks based on whether the PN-code is embedded in the report data queried from the data center of the ITM system. We conduct extensive simulations on the *iLOC* attack using real-world traces. Our data demonstrate that the *iLOC* attack can accurately identify monitors while remaining invisible to the ITM. Finally, we present a set of guidelines to counteract the *iLOC* attack.**

***Index Terms*—Internet threat monitoring systems, Invisible localization attack, PN-code, Security**

I. Introduction

In recent years, widespread attacks, such as active worms [1] and distributed denial-of-service (DDoS) attacks [2], have been major threats to the Internet. Due to the widespread nature of these attacks, large scale traffic monitoring across the Internet has become necessary in order to effectively detect and defend against them. Developing and deploying *Internet threat monitoring* (ITM) systems (or *motion sensor networks*) is one of the major efforts in this realm.

Generally, an ITM system consists of a number of monitors and a data center. The monitors are distributed across the Internet and can be deployed at hosts, routers, firewalls, etc. Each monitor is responsible for monitoring and collecting traffic targeting a range of IP addresses within a sub-network. The range of IP addresses covered by a monitor is also referred to as the *location* of the monitor. Periodically, the monitors send traffic logs to the data center. The data center aggregates and analyzes these logs and also publishes reports to the public. The reports provide critical insights into widespread Internet threats and attacks, and are used to detect and defend against such attacks. ITM systems have been successfully used to detect the worm outbreaks [3] and DDoS attacks [4].

However, the integrity and functionality of ITM systems largely depend on the anonymity of the IP addresses covered by their monitors, i.e., the *locations* of monitors. If the locations of monitors are identified, the attacker can deliberately avoid these monitors and directly attack the uncovered IP address space, which *significantly* degrades the effectiveness of the ITM system. Hence, it is important to have a thorough understanding of such attacks, in order to design efficient countermeasures against them, thereby protecting ITM systems.

In this paper, we investigate a new class of attacks, the *i*nvisible *LOC*alization (*iLOC*) attack. In the *iLOC* attack, the attacker launches low-rate port-scan traffic (also referred to as *attack traffic*) to target networks. The scan traffic is encoded with a carefully selected *pseudo-noise code* (PN-code), which is only known to the attacker. The PN-code embedded in traffic can be accurately recognized by the attacker even under interference from background traffic. Thus, the attacker is able to *accurately* determine the existence of monitors in the target networks based on whether the embedded PN-code is contained in the report data queried from the data center of the ITM system. The attack traffic modulated/embedded by the PN-code will appear as innocent noise in both the time and frequency domains, rendering it *invisible* to others. Only those aware of the original PN-code can correctly recover the encoded PN-code and identify the monitor locations. Therefore, using the *iLOC* technique, the attacker can accurately localize monitors while evading detection by others. The performance data demonstrate that the attack can accurately identify the locations of monitors, while evading detection by those unaware of the attacker-selected PN-code. Furthermore, we present a set of guidelines on how to counteract the *iLOC* attack.

A few works have been conducted on monitor localization attacks [5], [6]. However, our work is the first to address an attack aiming to achieve the objectives of both accuracy and invisibility. Notice that invisibility is critical for the attacker to evade these countermeasures. Work in [7] also studied how to use the PN-code to effectively track flows through anonymous systems. Since it is applied to a different problem domain, the solution in [7] is significantly different from the one in this paper, including the use of the PN code, designed algorithms, decision rule, and theoretical analysis.

Xun Wang and Dong Xuan are with the Department of Computer Science and Engineering, The Ohio State University, Columbus, OH 43210. E-mail: {wangxu, xuan}@cse.ohio-state.edu. Wei Yu is with the Department of Computer Science, Texas A&M University, College Station, TX 77843. E-mail: weiyu@cs.tamu.edu. Xinwen Fu is with the College of Business and Information Systems, Dakota State University, Madison, SD 57042. E-mail: xinwen.fu@dsu.edu. Wei Zhao is with the School of Science, Rensselaer Polytechnic Institute, Troy, NY 12180. E-mail: zhaow3@rpi.edu. The authors would like to acknowledge Adam Champion and Ms. Larisa Archer for their dedicated help to improve the paper. This work was supported in part by the US National Science Foundation (NSF) CAREER Award CCF-0546668 and the Army Research Office (ARO) under grant no. AMSRD-ACC-R 50521-CI. Any opinions, findings, conclusions, and recommendations in this paper are those of the authors and do not necessarily reflect the views of the funding agencies.

978-1-4244-2026-1/08/$25.00 © 2008 IEEE

The remainder of the paper is organized as follows. In Section II, we describe the *iLOC* attack in detail. In Section III, we report our performance evaluation results on the *iLOC* attack. In Section IV, we discuss some preliminary countermeasures against the *iLOC* attack. Finally we conclude this paper in Section V.

II. *iLOC* ATTACK

In this section, we discuss the *iLOC* attack in detail. We first give an overview of the *iLOC* attack, and then present the detailed stages of the attack, followed by additional discussions on its mechanisms.

A. Overview

Fig. 1 shows the basic workflow of the *iLOC* attack. This figure also illustrates the basic idea of the ITM system. In the ITM system, the monitors deployed at various networks record their observed port-scan traffic and continuously update their traffic logs to the data center. The data center first summarizes the volume of port-scan traffic towards (and reported by) all monitors, and then publishes the report data to the public in a timely fashion.

As shown in Fig. 1 (a) and (b) respectively, the *iLOC* attack consists of the following two stages:

1) Attack Traffic Generation: In this stage, as shown in Fig. 1 (a), the attacker first selects a code and then encodes the attack traffic by *embedding* the selected code into the traffic. Lastly, the attacker launches the attack traffic towards a target network (e.g., network A in Fig. 1 (a)). We denote such an *embedded code pattern* in the attack traffic as the *attack mark* of the *iLOC* attack, and denote the encoded attack traffic as *attack mark traffic*.

2) Attack Traffic Decoding: In this stage, as shown in Fig. 1 (b), the attacker first queries the data center for the traffic report data, which consists of both attack traffic and background traffic. After obtaining the report data, the attacker tries to recognize the attack mark (i.e., the code embedded in the *iLOC* attack traffic) by decoding the report data. If the attack mark is recognized, the report data must include the attack traffic, which means monitors are deployed in the target network and they are sending traffic reports to the ITM data center.

The *iLOC* attack adopts a code based approach to generate the attack traffic. Coding techniques have been widely implemented in secure communication; for example, *Morse code* is one such example. Without knowledge of Morse code, the receiver would find it impossible to interpret the carried information [8]. In the *iLOC* attack, the PN-code-based approach has three advantages. First, the code is embedded in traffic and can be correctly recognized by the attacker even under interference from background traffic, ensuring accuracy of the attack. Second, the code (of sufficient length) itself provides enough privacy. That is, the code is only known by the attacker, thereby the code pattern embedded in attack traffic can only be recognized by the attacker. Furthermore, the code can carry information. A longer code is more immune to interference, and requires comparatively lower-rate attack traffic as the carrier, which is harder to be detected. All these characteristics help to achieve the objectives of attack accuracy and invisibility. We will discuss the details of these two stages in the following.

B. Attack Traffic Generation Stage

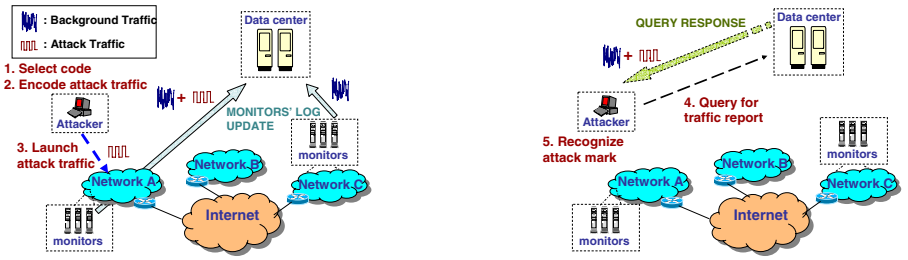
In this attack stage, the attacker: (1) selects the code, a *PN-code* in our case; (2) encodes the attack traffic using the selected PN-code; and (3) launches the encoded attack traffic towards the target network. For the third step, the attacker can coordinate a large number of compromised bots to launch the traffic [9]. However, this is not the focus of this paper. In the following, we will present detailed discussion on the first and second steps, respectively.

1) Code Selection: To evade detection by others, the attack traffic should be similar to the background traffic. From a large set of real-world background traffic traces obtained from SANS ISC [3], [10], we conclude that the background traffic shows random patterns in both time and frequency domains. The attack objectives of both accuracy and invisibility, and the attacker's desire for parallel attacks require that: (1) the encoded attack traffic should blend in with background traffic, i.e., be random in both the time and frequency domains, (2) the code embedded in the attack traffic should be easily recognizable to the attacker himself, and (3) the code should support parallel attacks.

To meet the above requirements, we choose the PN-code to encode the attack traffic. The PN-code in the *iLOC* attack is a sequence of -1 or $+1$ with the following features [11]. First, the PN-code is random and "balanced". This feature makes the attack traffic appear as noise and blend in with background traffic in both time and frequency domains. Second, The PN-code has a high correlation with itself and a low correlation with other signals (such as random noise). This feature makes it feasible for the attacker to accurately recognize attack traffic (encoded by the PN-code) from the traffic report data even under interference from background traffic. There are mature PN code generators such as m-sequences code, Barker code, gold codes and Hadamard-Walsh codes [11] that we may adopt. In our work, we use the m-sequence code, which has the best autocorrelation (it only highly correlates to itself with a sharp autocorrelation peak) [11]. The improved autocorrelation makes it easier for the attacker to accurately synchronize and recognize the pattern embedded in the probing traffic.

2) Attack Traffic Encoding: During the attack traffic encoding process, each bit in the selected PN-code is mapped to a unit time period T_s, denoted as *mark bit duration*. The entire duration of launched traffic (referred to as *traffic launch session*) is $T_s \cdot L$, where L is the length of the PN-code.

The encoding is carried out according to the following rules: each bit in the PN-code maps to a mark bit duration (T_s); when the PN-code bit is $+1$, port-scan traffic with a high rate, denoted as *mark traffic rate* V, is generated in the corresponding mark bit duration; when the code bit is -1, no port-scan traffic is generated in the corresponding



(a) Attack stage 1: attack traffic generation

(b) Attack stage 2: attack traffic decoding

Fig. 1. Workflow of the *iLOC* Attack

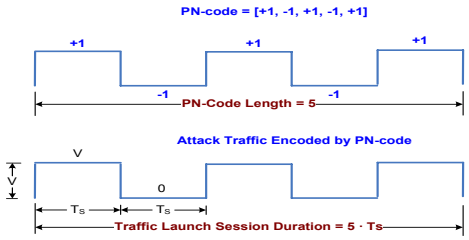


Fig. 2. PN-code and Encoded Attack Traffic

mark bit duration. Thus, the attacker embeds the attack traffic with a special pattern, i.e., the *original* PN-code. Recall that, after this encoding process, the PN-code pattern *embedded* in traffic is denoted as the *attack mark*. If we use $C_i = \langle C_{i,1}, C_{i,2}, \ldots, C_{i,L}\rangle \in \{-1,+1\}^L$ to represent the PN-code and use $\eta_i = \langle \eta_{i,1}, \eta_{i,2}, \ldots, \eta_{i,L}\rangle$ to represent the attack traffic, then we have $\eta_{i,j} = \frac{V}{2} \cdot C_{i,j} + \frac{V}{2}$. That is, $\eta_{i,j} = V$ if $C_{i,j} = +1$ and $\eta_{i,j} = 0$ if $C_{i,j} = -1$ ($j = 1, \ldots, L$). Fig. 2 shows an example of the PN-code and the corresponding encoded attack traffic.

C. Attack Traffic Decoding Stage

In this stage, the attacker takes the following two steps: (1) The attacker queries the data center for the traffic report data, which consists of both attack traffic and background traffic. (2) From the report data, the attacker attempts to recognize the embedded attack mark. The existence of the attack mark determines the deployment of monitors in the attack targeted network. As querying of traffic report data is relatively straightforward, here we only detail the second step, i.e., attack mark recognition, as follows.

In the report data queried from the data center, the attack traffic encoded with the attack mark is mixed with background traffic. It is critical for the *iLOC* attack to accurately recognize the attack mark from the traffic report data. To address this, we develop the correlation-based scheme. This scheme is motivated by the fact that the original PN-code (used to encode attack traffic) and its corresponding attack mark (embedded in the traffic report data) are highly correlated, in fact, they are actually the same.

The attack mark in the traffic report data is the *embedded form* of the original PN-code. The attack mark is similar to its original PN-code, although the background traffic may introduce interference and distortion into the attack mark. We adopt the following correlation degree to measure their similarity. Mathematically, *correlation degree* is defined as the inner product of two vectors. For two vectors $X = \langle X_1, X_2, \ldots, X_L\rangle$ and $Y = \langle Y_1, Y_2, \ldots, Y_L\rangle$ of length L, the correlation degree of vector X and Y is

$$\Gamma(X,Y) = X \odot Y = \frac{\sum_{i=1}^{L} X_i \cdot Y_i}{L}, \quad (1)$$

where $\odot$ represents the operator for the inner product of two vectors. Based on the above definition, we have $\Gamma(X,X) = \Gamma(Y,Y) = 1$, $\forall\ X, Y \in \{-1,+1\}^L$.

We use two vectors, $\eta_i = \langle \eta_{i,1}, \eta_{i,2}, \ldots, \eta_{i,L}\rangle$ and $\omega_i = \langle \omega_{i,1}, \omega_{i,2}, \ldots, \omega_{i,L}\rangle$ to represent attack traffic (embedded with the attack mark) and background traffic, respectively. We *shift* the above two vectors by subtracting the mean value from the original data, resulting in two new vectors, $\eta'_i = \langle \eta'_{i,1}, \eta'_{i,2}, \ldots, \eta'_{i,L}\rangle$ and $\omega'_i = \langle \omega'_{i,1}, \omega'_{i,2}, \ldots, \omega'_{i,L}\rangle$. We still use a vector $C_i = \langle C_{i,1}, C_{i,2}, \ldots, C_{i,L}\rangle \in \{-1,+1\}^L$ to represent the PN-code. Thus, the correlation degree between the PN-code and the (shifted) attack traffic can be obtained. Similarly, we can also obtain the correlation degree between the PN-code and the (shifted) background traffic as follows.

According to the rules of encoding attack traffic discussed in Section II-B2, $\eta_i = \frac{V}{2} \cdot C_i + \frac{V}{2}$. Thus, $\eta'_i = \eta_i - E(\eta_i) = \eta_i - \frac{V}{2} = \frac{V}{2} \cdot C_i$. Hence, the correlation degree between the original PN-code and the (shifted) attack traffic is $\Gamma(C_i, \eta'_i) = \frac{V}{2} \cdot \Gamma(C_i, C_i) = \frac{V}{2}$. Furthermore, we can also derive the correlation degree between the PN-code and the (shifted) background traffic, i.e., $\Gamma(C_i, \omega'_i)$. The mean of this correlation degree is close to 0, since the PN-code has low correlation with the (shifted) background traffic (i.e., $E[\Gamma(C_i, \omega'_i)] = \frac{1}{L} E[\sum_{j=1}^{L} (\omega'_{i,j} \cdot C_{i,j})] \approx 0$). If the standard deviation of the background traffic rate is σ_x, the variance of such correlation degree is

$$Var[\Gamma(C_i, \omega'_i)] = E[(\Gamma(C_i, \omega'_i) - 0)^2] \quad (2)$$

$$\approx \frac{1}{L^2} E[\sum_{j=1}^{L} {\omega'_{i,j}}^2] = \frac{{\sigma_x}^2}{L}. \quad (3)$$

Thus, the standard deviation of correlation degree between the PN-code and the (shifted) background traffic is $\Gamma(C_i, \omega'_i) \approx \frac{\sigma_x}{\sqrt{L}}$. Based on the above discussion, the attacker can set appropriate attack parameters (e.g., PN-code length L and mark traffic rate V) to make the correlation degree ($\frac{V}{2}$) between the PN-code and the attack mark traffic much larger than the correlation degree ($\frac{\sigma_x}{\sqrt{L}}$) between the PN-code and the background traffic. Consequently, the attacker can accurately distinguish the attack mark traffic from the background traffic.

In the attack mark recognition, vector λ_i is used to represent the queried report data, and vector λ'_i is used to represent the *shifted* report data (by subtracting $E(\lambda_{i,j})$ from λ_i). According to above discussion, $\lambda'_i = \eta'_i + \omega'_i$ (i.e., report data include the attack traffic and background traffic) or $\lambda'_i = \omega'_i$ (i.e., report data include only attack traffic). The attacker uses the correlation degree between λ'_i and his PN-code C_i, i.e., $\Gamma(C_i, \lambda'_i)$, to distinguish between the above two cases and determine the existence of PN-code in the report data. If $\Gamma(C_i, \lambda'_i)$ is larger than a threshold T_a , which is referred to as the *mark decoding threshold*, then the attacker determines that the report contains attack traffic as well as the PN-code C_i, and determines that monitors are deployed in the target network.

D. Discussions

In order to accurately and effectively recognize the attack mark (PN-code) from the report data, we need to find the segment of the report data containing the PN-code (i.e., we need to fulfill the synchronization between the port-scan traffic report data and the PN-code). For this purpose, we introduce an iterative sliding window based scheme. The basic idea is to allow the attacker to obtain enough report data with fine granularity. Then, a sliding window iteratively moves forward to capture a segment of the report data. For each segment, we apply the correlation-based scheme discussed in Section II-C to recognize whether or not the attack mark exists.

III. Performance Evaluation

A. Evaluation Methodology

In our evaluation, we use the real-world port-scan traces from SANS ISC (Internet Storm Center) including the detailed logs from 01/01/2005 to 01/15/2005 [3].[1]. The traces used in our study contain over 80 million records and the overall data volume exceeds 80 GB. We use these real-world traces as the background traffic. We merge records of simulated *iLOC* attack traffic into these traces and replay the merged data to emulate the *iLOC* attack traffic. We evaluate different attack scenarios by varying attack parameters. Here, we only show the data on port 135; simulations that use other ports result in similar observations.

In order to measure attack accuracy, we introduce the following two metrics. The first metric is the *attack successful rate* PA_D, which is the probability that an attacker correctly recognizes the fact that a selected target network is deployed with monitors. The higher PA_D is, the higher the attack accuracy is. The second metric is the *attack false positive rate* PA_F, which is the probability that the attacker mistakenly declares a target network as one deployed with monitors. The lower PA_F is, the higher is the attack accuracy. To measure attack invisibility in terms of how well the *iLOC* attack can evade the defender detection, we use the following two metrics. The first metric is the *defender detection rate* PD_D, the probability that the defender correctly detects the attack traffic introduced by the *iLOC* attack. The second one is the *defender false positive rate* PD_F, the probability that the defender mistakenly identifies the attack traffic.

[1]We thank the ISC for providing us valuable traces in this research.

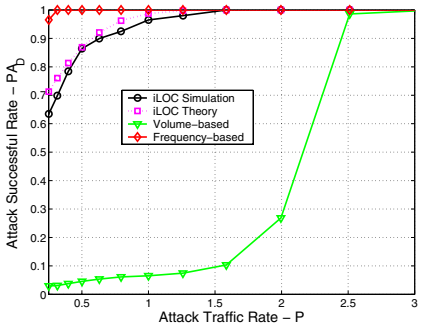


Fig. 3. Attack Successful Rate (Port 135)

We evaluate the *iLOC* attack in comparison with two other baseline attack schemes. The first one is the localization attack that launches a very high rate of port-scan traffic towards target networks as introduced in [5], [6]. We denote this attack as *volume-based attack*. The second baseline scheme embeds the attack traffic with a unique frequency pattern. In this attack, the attack traffic rate changes periodically. Then the attacker expects that the report data from the data center will show this unique frequency pattern if the selected target network is deployed with monitors. We denote this attack scheme as *frequency-based attack*. As we will illustrate in the following subsection, this attack scheme has high invisibility in the time domain. However, its invisibility cannot preserved in the frequency domain because there is a unique frequency pattern in the attack traffic. Specifically, when a *Fourier Transform* is applied to a traffic containing a periodic pattern, the periodic pattern emerges as obvious in the frequency domain to the defender.

In the interest of fairness, we adjust the detection thresholds in all schemes so that the *attack false positive rate* PA_F and *defender false positive rate* PD_F have reasonable values (below 1%). For the *iLOC* attack, we use the normalized attack traffic rate P, which is defined as $P = V/\sigma_x$ for *iLOC* attack, where σ_x is the standard variation of background traffic rate.

B. Results

1) Attack Accuracy: To compare the attack accuracy of the *iLOC* attack with those of volume and frequency-based attack schemes, we plot the attack successful rate PA_D under different attack traffic rates (i.e., $P \in [0.01, 3]$) as shown in Fig. 3. From this figure, we observe that both *iLOC* and frequency-based attacks consistently achieve a much higher attack successful rate PA_D than the volume-based scheme. This difference in PA_D is more significant when the attack traffic rate is lower, which can be explained as follows. For the *iLOC* scheme, the PN-code-based encoding and decoding make the recognition of attack marks robust to interference from the background traffic. For the frequency-based scheme, the invariant frequency in the attack traffic is also robust to background traffic interference. Both schemes can distinguish their attack traffic accurately even when the attack traffic rate

TABLE I
DEFENDER DETECTION RATE PD_D (PORT 135)

PA_D	*iLOC*(Time)	*iLOC*(Freq)	Volume-based attack (Time)	Frequency-based attack (Freq)	Frequency-based attack (Time)
90%	2.5%	2.2%	90%	90%	2.9%
95%	2.8%	2.4%	95%	95%	3.1%
98%	3.1%	2.8%	98%	98%	3.3%

(i.e., P) is small. Nevertheless, the volume-based scheme relies on a high rate of attack traffic (i.e., large P), and thus is very sensitive to the interference from background traffic.

2) Attack invisibility: To compare the attack invisibility performance of the *iLOC* attack with the that of other two attack schemes, we show the defender detection rate PD_D on port 135 in Table I. This table shows the attacker-achieved defender detection rate PD_D given different localization successful rates PA_D (90%, 95%, and 98%). Recall that the defender sets the detection threshold to make the defender false positive rate PD_F below 1%. In the table, "(Time)" and "(Freq)" mean that the defender adopts the *time-domain* and *frequency-domain* analytical techniques to detect attacks. An observation from this table is that our *iLOC* scheme consistently achieves a much lower defender detection rate PD_D than the other two schemes do, which means the *iLOC* attack achieves the best attack invisibility performance. As expected, the defender can easily detect the frequency-based attack by the frequency domain analytical technique, since there is a unique frequency pattern in its attack traffic.

IV. GUIDELINES OF COUNTERMEASURE

It is relatively easy to defend against volume-based and frequency-based localization attacks. The reason is that they either embed a spike (using high-rate scan traffic) [5], [6] or an invariable frequency (using a certain frequency pattern), and thus show strong signatures in the attack traffic (in either the time domain or frequency domain). However, defending against the *iLOC* attack is much more challenging due to its invisibility feature. In the following, we present some general guidelines for counteracting the *iLOC* attack, while complete countermeasures against it is a part of our future research efforts.

1) Perturbing the Information: Recall that in the *iLOC* attack, the attacker has to recognize the encoded attack traffic. Thus, the *quality* of reports plays an important role in this recognition. To reduce the effectiveness of *iLOC* attack, we may *perturb* the published report data by adding some random noise and even randomizing the data publishing delay. This principle is similar to data perturbation in the private data sharing realm [12]. The date center can also confuse the attacker by setting a random and dynamic *dormant monitor set* whose traffic logs will not be aggregated into the *currently* published traffic report. Perturbing report data can degrade the attack accuracy of *iLOC* attack. However, it will also impact the data accuracy and usage of ITM systems. Studying this trade-off will be one aspect of our future work.

2) Limiting Information Access: Recall that in the *iLOC* attack, the attacker must query the traffic report from the data center of ITM systems in order to accurately recognize the encoded attack traffic. We may explore this attack behavior feature to reduce the effectiveness of *iLOC* attack. To do so, the data center may throttle the query request rate or require strict authenticated in order to access the traffic report. However, these limitations on information access may also reduce the accessibility and thus the usage of ITM systems.

V. FINAL REMARKS

In this paper, we investigated a new class of attacks, i.e., the *i*nvisible *LOC*alization (*iLOC*) attack. It can accurately and invisibly localize monitors of ITM systems. Its effectiveness is demonstrated by theoretical analysis and simulations with real-world Internet traffic trace. We believe that this paper lays the foundation for ongoing studies of attacks that intelligently adapt attack traffic to defenses. Our study is critical for securing and improving ITM systems.

Detection of invisible attacks such as *iLOC* attacks and design of corresponding countermeasures against them remain challenging tasks and we will investigate them in our future research. Also, we believe that other vulnerabilities exist in ITM systems and we plan to conduct a thorough investigation of them and develop corresponding countermeasures.

REFERENCES

[1] D. Moore, C. Shannon, and J. Brown, "Code-red: a case study on the spread and victims of an internet worm," in *Proceedings of the 2-th Internet Measurement Workshop (IMW)*, November 2002.
[2] J. Mirkovic and P. Reiher, "A taxonomy of ddos attack and ddos defense mechanisms," *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 2, pp. 39–54, 2004.
[3] SANS, *Internet Storm Center*. http://isc.sans.org/.
[4] D. Moore, G. M. Voelker, and S. Savage, "Infering internet deny-of-service activity," in *Proceedings of the 10-th USENIX Security Symposium (SECURITY)*, Auguest 2001.
[5] J. Bethencourt, J. Frankin, and M. Vernon, "Mapping internet sensors with probe response attacks," in *Proceedings of the 14-th USENIX Security Symposium*, July-August 2005.
[6] Y. Shinoda, K. Ikai, and M. Itoh, "Vulnerabilities of passive internet threat monitors," in *Proceedings of the 14-th USENIX Security Symposium*, July-August 2005.
[7] W. Yu, X. Fu, S. Graham, D. Xuan, and W. Zhao, "Dsss-based flow marking technique for invisible traceback," in *Proceedings of IEEE Symposium on Security and Privacy (S&P)*, May 2007.
[8] L. Y. Chuang, C. H. Yang, C. H. Yang, and S. L. Lin, "An interactive training system for morse code users," in *Proceedings of Internet and Multimedia Systems and Applications*, August 2002.
[9] R. Naraine, *Botnet Hunters Search for Command and Control Servers*. http://www.eweek.com/article2/0,1759,1829347,00.asp.
[10] Dshield, *Distributed Intrusion Detection System*. http://www.dshield.org/.
[11] R. K. Pickholtz, D. L. Schilling, and L. B. Milstein, "Theory of spead-spectrum communication - tutorial," *IEEE Transaction on Communication*, vol. 30, no. 5, pp. 855–884, 1982.
[12] R. Agrawal, A. Evfimievski, and R. Srikant, "Information sharing across private database," in *Proceeding of the 22-th SIGMOD International Conference on Management of Data*, July 2003.