Agent-enabled Anomaly Detection in Resource Constrained Wireless Sensor Networks

Muhammad Usman
School of Information and Communication Technology, Griffith University, Gold Coast, Australia
email: *muhammad.usman3@griffithuni.edu.au*

***Abstract*—The state-of-the-art network anomaly detection systems are often unable to correctly verify source of anomalies; they have occurred *in situ* or *in transit*. To address this issue, this study proposes an agent-enabled anomaly detection system in which mobile agents perform *in situ* verification of sensor nodes to verify source of anomalies. Moreover, mobile agent cannot be freely transmitted over the network, as transmission is several hundred times expensive operation as compare to processing. We optimize agent transmission through *2-sigma* method. The formal behavior analysis of *2-sigma* method is performed using Petri net theory to analyze its correctness. In future work, we aim to exploit cross-layer (XL) interaction to efficiently transmit mobile agents.**

I. Motivation

The advent of shared sensor network has given birth to new applications such as wireless sensing for the built environment. The reliability of such applications is highly dependent on the accuracy of received data. However, transmitted data is vulnerable to *in situ* and *in transit* anomalies due to faults and attacks. In such cases, an effective Anomaly Detection System (ADS) should be able to identify the source of anomalies to appropriately mitigate them. One possible approach for verification of source of anomalies can be physical diagnosis of the suspicious sensor node [1]. However, it is not possible to intervene privacy of residents every now and then to perform physical diagnosis of the node. Therefore, we propose a software mobile agent-enabled ADS that addresses following research challenges: (1) How can an ADS be deployed to detect complex nature of anomalies if a sensor node behaves anomalous? (2) How can *in situ* verification of anomalous node be performed by a component of an ADS? (3) How can transmission of mobile agent be efficiently managed?

II. System Model

In this section, we present system model and high level working of our proposed methods to address research issues raised in the preceding section. We consider Wireless Sensor Network (WSN), as an extraverted bidirected graph, G = (V, E), where V and E denotes vertices (nodes) and edges (communication links), respectively. The nodes set $\boldsymbol{V} = \bigcup_{i=1}^{3} V_i$; where V_1 is laptop class device and acts as a controller to the network, $V_2 = \bigcup_{j=1}^{m} \mathsf{v}_j$ are resource rich nodes that works as regional chiefs of their constituencies, $V_3 = \bigcup_{k=1}^{n} v_k$ are low resource nodes that collect readings from their ambient environment, store them in their memory for the purpose of *in situ* verification, and transmit to the corresponding v_j node. It is assumed that $\mid V_1 \mid \leq \mid V_2 \mid \leq \mid V_3 \mid$. The v_j nodes are aware of resource status of member v_k nodes through hardware state sharing mechanism, namely, Coordinated Resource Management (CRM) [2]. This facilitates system administrator not only to allocate more processing to those v_k nodes which has higher energy resources but also use gathered information for anomaly detection and *in situ* verification process.

Each v_j is equipped with an anomaly detection module (ADM). The ADM is constituted of three logical units, viz., Coordination Unit (CU), repository, and Mobile Agent (MA). The CU is responsible for intra-module communication as well as with other entities of the network such as v_1, other v_j, and member v_k nodes. The repository stores normal profile and anomaly detection rules for member v_k nodes. The structure of repository and normal profile is described in section II-A. The ADM, at v_j node, initiates anomaly detection process upon receipt of network traffic from member v_k nodes. If received observation, $f_i \in FS$, is found normal then it is aggregated for the predefined period of time before its transmission to the v_1 node, where $FS = \{ f_1, f_2, ..., f_p\}$ is the set of observations that are sequentially transmitted from v_k to v_j nodes. On the other hand, if f_i is found anomalous, as per pre-defined normal behavior of v_k node, then MA is transmitted for *in situ* verification of suspicious behavior of sensor node. MA carries historical values of f_i, that are gathered through CRM mechanism, to verify that anomalies have caused *in-situ* or *in transit*.

Transmission of MA may yield one of three scenarios, namely, (1) successful execution of MA at suspicious v_k node, (2) suspicious v_k node may defend execution of MA and does not send any result to respective v_j, and (3) migration of AA to suspicious v_k node is vulnerable to agent execution manipulation attack. For case (1), MA will send desired result to v_j. For case (2), if suspicious v_k node defends execution of MA and does not send any result to v_j, it will confirm its anomalous status. For case (3), MA execution integrity protection mechanism can be employed [3]. Note that case (3) is only applicable for those situations where v_k nodes are vulnerable to more sophisticated attacks and antagonist takes full control of the node which is not very common. More so, MA execution integrity protection mechanism is not required for anomalies that are caused due to faults or other types of attacks.

A. Anomaly detection

The proposed structure of repository is an extension to the Denning's six tuples [4].

$$N_{prf} =< N, R, Prf, Au_{rec}, An_{rec}, Act_{set} > \qquad (1)$$

where N, R, Prf, Au_{rec}, An_{rec}, and Act_{set} denotes node identities, their resource status, normal profile, audit record, anomaly record, and action set, respectively. The $N = \{N_1, N_2, ..., N_m\}^T$ is a column vector, R is $m \times n$ matrix, where m denotes sensor node and n are types of resources. The basic structure of normal profile is, $Prf = < \lambda, \tau, \phi, \upsilon, F >$, where λ, τ, ϕ, υ, and F denotes sensor reading, time interval, entitled actions (such as read, write, and sense), resource status, and received packet count, respectively. To discover range of anomalies, we exploit association among features of Prf to establish first-order joins. Given below are two sample first-order joins.

$$N_{reg}(\lambda, \tau) = \int_{\lambda_i}^{\lambda_f} \int_{\tau_i}^{\tau_f} f(\lambda, \tau)\, \mathrm{d}\tau \mathrm{d}\lambda. \tag{2}$$

$$N_{reg}(\phi, \tau) = \sum_{\phi_i}^{\phi_f} \int_{\tau_i}^{\tau_f} f(\phi, \tau)\, \mathrm{d}\tau \mathrm{d}\phi. \tag{3}$$

Au_{rec} store values of previous observations for anomaly detection decision making process. An_{rec} records anomalous observations before their transmission to V_1 and V_2 nodes (if required in latter case). $Act_{set} = \langle \alpha, \beta, \gamma \rangle$, where α, β, and γ denotes anomaly detection, periodic anomaly detection, and tuning actions, respectively, and executes as per security requirements of the application.

B. Agent transmission

MA cannot be freely transmitted over the network due to expensive nature of communication operation. However, curtailment of MA transmission should be carefully designed so that it cannot affect the overall performance of detection scheme. Our *2-sgima* method employs two standard deviations to define curtailment zone over the probability distributions of Prf features. The observation that lies between first and second standard deviations (*i.e.*, $1\sigma < f_i \leq 2\sigma$ or $-1\sigma > f_i \geq -2\sigma$) is considered as tolerated and that lies outside two standard deviations (*i.e.*, $f_i > 2\sigma$ or $f_i < -2\sigma$) is treated as suitable to trigger MA for *in situ* verification. The curtailment is only for suspicious node's verification process. Meanwhile, ADM may take other usual actions such as decrement in trust level, reducing communication with node, and isolating node from rest of the network. Setting curtailment zone causes less frequent transmissions of MAs. This method reduces resource consumption by both V_2 and V_3 nodes, hence, increases lifetime of the network.

III. Performance Evaluation

We present partial theoretical results of the proposed model. Our formal behavior analysis for *2-sigma* method is based on Petri net theory [5]. The formal specifications of agent transmission method is given below.

Agent transmission net: The net, (PN_{AT}), is a 5-tuple net: $PN_{AT} = (P, T, F, W, M_0)$, where $P = \{p_1, p_2, ..., p_6\}$ and $T = \{t_1, t_2, t_3\}$ are non-empty, finite, and disjoint sets of places and transitions, respectively. $F = \{(p_1, t_1), (t_1, p_2), (t_1, p_3), (p_3, t_2), (t_2, p_4), (t_2, p_5), (p_5, t_3), (t_3, p_6)\}$ is set of arcs, W = 1 is weight for all arcs, and $M_0 = p_1$ denotes token in the first place of net.

Based on above formal specifications, we can prove following results.

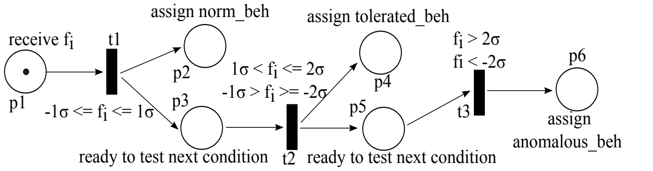

Figure 1: The Petri net for *2-sigma* method

Theorem 1. *The net, PN_{AT}, is safe and level 4 live.*

Proof: $\forall\ p \in P$ (i.e., p_1 to p_6) marking $M(p) = 1$. As flow of single token constitutes workflow of the system. This yields net PN_{AT} is safe. The terminal transition t_3 is live at level 4 as there exist firing sequence $t_1 \rightarrow t_2 \rightarrow t_3$. Similar is the case for other transition, i.e., t_2 which has firing sequence $t_1 \rightarrow t_2$, as shown in Figure 1. Thus, PN_{AT} is level 4 live which means there is no deadlock in the system. ■

Theorem 2. *The net, PN_{AT}, is sequentially executable to optimize agent transmission.*

Proof: The transition t_1, denoting condition $-1\sigma \leq f_i \leq 1\sigma$ is enabled after receipt of token in place p_1, representing receipt of feature set f_i. The firing of transition t_1 yields token either in p_2 or p_3 showing normal behavior or ready to execute next condition status, respectively. The token in place p_3 enables transition t_2 denoting condition $1\sigma < f_i \leq 2\sigma$ and $-1\sigma > f_i \geq -2\sigma$. Firing of transition t_2 yields token in places p_4 and p_5. The token in p_5 enables transition t_3 representing condition $f_i > 2\sigma$ and $f_i < -2\sigma$. The firing of transition t_3 yields token in place p_6 denoting anomalous behavior assigned to node. This shows sequential workflow of PN_{AT}. ■

IV. Conclusions

In this work, we have presented an agent-enabled anomaly detection system. We outlined theoretical model and presented partial theoretical results. In a parallel work, as part of this study, we have performed detailed quantitative analysis of proposed methods through real implementation on Mica2 sensor motes and simulated experiments. In future work, we plan to exploit cross layer association for effective transmission of mobile agents.

Acknowledgment

The author is grateful to Dr. Vallipuram Muthukkumarasamy and Dr. Xin-Wen Wu for their devoted guidance for this study.

References

[1] J. Xiong, Y. Zhou, X.C. Lyu, E.F.Y. Young, and M.R. Lyu "MDiag: Mobility-assisted Diagnosis for Wireless Sensor Networks" *Journal of Network and Computer Applications*, Vol. 36, Issue 1, 167-177, 2013.

[2] J. Waterman, G. W. Challen, and M. Welsh, "Peloton: Coordinated resource management for sensor networks," *In proc. of 12th Workshop on Hot Topics in Operating Systems*, pp. 5, 2009.

[3] O. Esparza, J. L. Munoz, J. Tomas-Builart, and M. Soriano, "An infrastructure for detecting and punishing malicious hosts using mobile agent watermarking," *Wireless Communications and Mobile Computing*, Vol. 11, Issue 11, pp. 1446-1462, 2011.

[4] D. E. Denning, "An Intrusion-Detection Model," *IEEE Transactions on software engineering*, Vol. 13, Issue 2, pp. 222-232, 1987.

[5] T. Murata, "Petri Nets: Properties, Analysis, and Application," *In proc. of the IEEE,* Vol. 77, Issue 4, pp. 541-580, 1989.