POSTER: SMURFEN: A Rule Sharing Collaborative Intrusion Detection Network

Carol Fung[1], Quanyan Zhu[2], Raouf Boutaba[1], and Tamer Başar[2]
[1]Cheriton School of Computer Science, University of Waterloo, Ontario, Canada
{j22fung, rboutaba}@uwaterloo.ca
[2]Coordinated Science Laboratory, University of Illinois, Urbana, IL, USA
{zhu31, basar1}@illinois.edu

ABSTRACT

Intrusion Detection Systems (IDSs) are designed to monitor network traffic and computer activities in order to alert users about suspicious intrusions. Collaboration among IDSs allows users to benefit from the collective knowledge and information from their collaborators and achieve more accurate intrusion detection. However, most existing collaborative intrusion detection networks rely on the exchange of intrusion data which raises privacy concerns. To overcome this problem, we propose SMURFEN: a knowledge-based intrusion detection network, which provides a platform for IDS users to effectively share their customized detection knowledge in an IDS community. An automatic knowledge propagation mechanism is proposed based on a decentralized two-level optimization problem formulation, leading to a Nash equilibrium solution which is proved to be scalable, incentive compatible, fair, efficient and robust.

1. INTRODUCTION

Internet intrusions have become more sophisticated and difficult to detect. With the increasing complexity of software and systems, hackers can easily explore exposed software vulnerabilities and compromise the host computers. Not only private data and identify information are harvested from the compromised computers, hackers can also launch attacks to other computers such as Distributed Denial of Service (DDoS) attacks using the compromised computers.

As a counter measurement, Intrusion Detection Systems (IDSs) are designed to monitor network traffic and computer activities by raising intrusion alerts to network administrators or security officers. Traditional IDSs work independently from each other and rely on downloading new signatures or detection rules from the corresponding security vendor's signature/rule base to remain synchronized with new detection knowledge, such as Snort [1]. However, the increasing number and diversity of intrusions render it not effective to rely on the detection knowledge from a single vendor, since not a single vendor can cover all the possible intrusions due to limited human resource and available technology. Collaborative intrusion detection networks (CIDNs) provide a platform for IDSs to take advantage of the collective knowledge from collaborators to improve the overall detection capability and accuracy. However, most existing CIDNs rely on the sharing of intrusion data with others, which raises privacy concerns. Instead, sharing detection knowledge such as malware signatures and intrusion detection rules, causes less privacy concern.

In reality, expert IDS users, including security analysts, network administrators, and security system programmers, create their own detection rules or customize existing ones to improve detection accuracy specifically for their individual environment [3]. A new detection rule created by one user may be adopted directly by another user if they have similar network/computer configurations. Sharing rules among a large group of users can be an effective way to improve the overall security among all users.

In this work, we leverage the benefit of intrusion detection knowledge sharing and propose SMURFEN, a knowledge-based collaborative intrusion detection network, where intrusion detection knowledge is shared among users who share similar interests in the community. The system is built upon a peer-to-peer overlay network for its scalability. An automatic knowledge dissemination mechanism is used to allow peers effectively share detection rules with others without overwhelming their receiving capacities. We demonstrate using simulation that the proposed rule sharing mechanism can effectively improve the overall security of the community and provides incentive-compatibility and fairness to the collaborators.

2. THE SMURFEN FRAMEWORK

The SMURFEN framework is built upon a Chord [4] peer-to-peer (p2p) communication overlay as illustrated in Fig. 1. Each node also maintains a list of neighbors to communicate and exchange intrusion detection rules with. We call such a list the *acquaintance list.* Note that the acquaintance relationship is symmetric, i.e., if node i is in node j's acquaintance list, then node j is in node i's acquaintance list.

A user on the receiver side evaluates rules sent from its neighbors and may choose to "*accept*" or "*reject*" the rule. The decision is then recorded by a Bayesian learning algorithm [5] to update the *trust* value of the sender. False positives revealed afterwards will be treated as double rejections in the trust calculation. The *trust* from i to j is the probability that the rules from the sender i are useful to the receiver j. The higher a collaborator's trust, the more helpful it is in collaboration. The decision of accepting a rule or not is also sent to a corresponding rule feedback collector. The feedback collector is a random node in the p2p network, determined by a hashed key of the rule ID. The corresponding hosting node holds the feedback of the rule.

CCS'11, October 17–21, 2011, Chicago, Illinois, USA.
ACM 978-1-4503-0948-6/11/10.

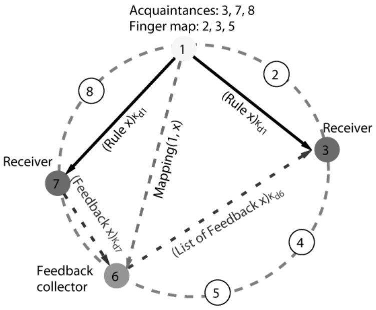

Figure 1: SMURFEN design of 8 nodes on a Chord ring: nodes 3 and 7 receive a rule from node 1. The feedbacks are collected by node 6.

Inexperienced users can check feedback from others before they make their decisions whether to accept the rule or not.

To prevent the man-in-the-middle attack, the communication between each pair of nodes is signed by the private key of the sender. When a new node joins the network, it creates a public/private key pair (K_e, K_d), and registers a new ID into the p2p network by sending a join request to any node in the network. After that, the new node sends connection requests to random nodes in the network and acquaintance relationships are established when the requests are accepted. When a node leaves a network, it is not required to send a notification to other nodes. When a node does not receive response from an acquaintance, it automatically sets the acquaintance status to be inactive and seeks new replacement.

A SMURFEN system feedback collectors is shown in Fig. 2, where rule author "A" propagates a new rule i to its acquaintances R_1 and R_2. Both rule receivers can retrieve and send feedback from/to the feedback collector C. Replicas collectors can be used to improve the availability of feedback collector service. All feedbacks are signed by their authors to prevent from malicious tampering.

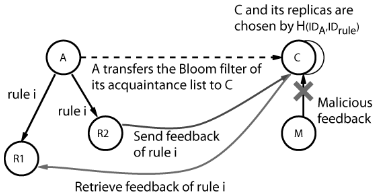
C and its replicas are chosen by H(ID_A,ID_rule)
A
C
A transfers the Bloom filter of its acquaintance list to C
rule i
rule i
R1
R2
M
Send feedback of rule i
Malicious feedback
Retrieve feedback of rule i

Figure 2: Feedback Collection in SMURFEN. The malicious node M attempts to leave fraudulent feedback but was blocked since it does not match the Bloom filter on the feedback collector.

Moreover, to avoid feedback fraudulence, each feedback collector maintains a Bloom filter [2] of the authorized nodes list. The rule author hashes all of its acquaintances into a Bloom filter and passes it to the feedback collector. Only nodes with hashed IDs matching the Bloom filter are allowed to leave feedback on the collector. The use of Bloom filter not only reduces the communication overhead to transfer long acquaintance lists, it also avoids unnecessary information leaking from the rule author.

3. KNOWLEDGE SHARING AND PROPAGATION MODEL

Intrusion detection knowledge propagation mechanism is an essential part of the SMURFEN system, where IDSs decide the propagation rates to their neighbors. An appropriate propagation design will not only provide incentive-compatibility which discourages free-riders and rewards contributors, it will also provide fairness to all participants and be robust to malicious insiders. In this work, we use a game-theoretical approach for each IDS to decide its rule propagation rates and we prove that the system yields to a Nash equilibrium.

We model our system based on a two-level optimization problem, i.e., a *public* utility optimization together with a *private* utility optimization. Each IDS i controls two decision variables, namely, $\vec{r_i}$ and $\vec{R_i}$. $\vec{r_i}$ is the *rule propagation rate* from node i to its neighbors. To prevent from denial of service attacks from malicious neighbors, a node i also sets a *requested sending rate* $\vec{R_i}$, which sets the upper bound of the sending rates from all neighbors. At the lower level, a node i solves the public optimization problem (PPi) where it chooses $\vec{r_i}$ to maximize the aggregated satisfaction levels of its neighbors. At the upper level, a node i determines $\vec{R_i}$ to solve a private optimization problem (Pi) to maximize the total return benefit from all neighbors. The choice of $\vec{R_i}$ at the upper level influences the decision-making at the lower public optimization level.

The public optimization problem (PPi) seen by each node $i, i \in \mathcal{N}$, is given by

$$\begin{aligned}
(\mathrm{PP}i) \max_{\vec{r_i} \in \mathbb{R}^{n_i}} U_i^r(\vec{r_i}) \quad &:= \quad \sum_{j \in \mathcal{N}_i} T_{ji} S_{ij}(r_{ij}) && (1)\\
\sum_{j \in \mathcal{N}_i} r_{ij} \quad &\leq \quad M_i, && (2)\\
r_{ij} \quad &\leq \quad R_{ij}, && (3)\\
0 \ \leq \ r_{ij} \quad &\leq \quad \widehat{r}_i, && (4)
\end{aligned}$$

where $S_{ij} : \mathbb{R} \to \mathbb{R}$ is the satisfaction level of node j in response to the propagation rate r_{ij} of node i. We let S_{ij} take the following form

$$S_{ij}(r_{ij}) := T_{ij} \log\left(1 + \frac{r_{ij}}{R_{ij}}\right). \tag{5}$$

The concavity and monotonicity of the satisfaction level indicate that a recipient becomes increasingly pleased when more rules are received but the marginal satisfaction decreases as number of received rules increases. The parameter T_{ij} in (5) suggests that a node j is more content when the trust or usefulness of rules sent from node i is high.

The objective function $U_i^r : \mathbb{R}^{n_i} \to \mathbb{R}$ in (1) aggregates the satisfaction level S_{ij} of node j by the trust factor T_{ji}. The utility U_i^r can be viewed as a public altruistic utility in that a node i seeks to satisfy its neighbors by choosing rule propagation rates $\vec{r_i}$. The problem (PPi) is constrained by (2) in that the total sending rate of a node i is upper bounded by its communication capacity. The additional constraint (4) ensures that the propagation rate does not exceed its rule contribution rate $\widehat{r}_i$. Note that the constraint (3) is imposed by its recipient j while constraint (4) is set by node i itself.

A node i has another degree of freedom to choose its level of requested sending rate R_{ji} of its neighbors. R_{ji} states the

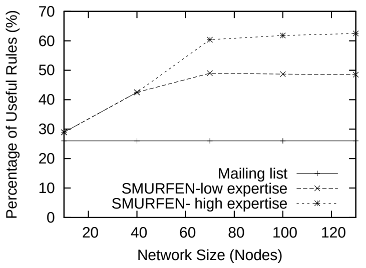

Figure 3: Propagation Efficiency Comparison

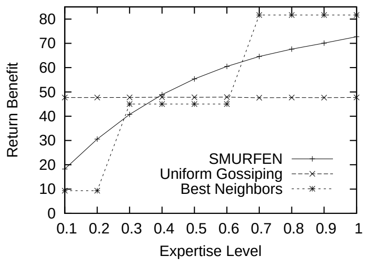

Figure 4: Incentive on Expertise Levels

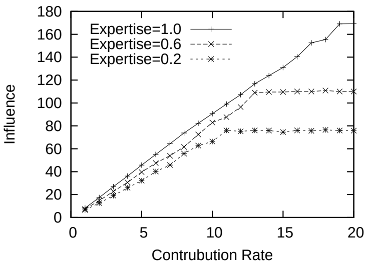

Figure 5: The Influence vs. Sending Rate

maximum rule propagation rate from node j to i that node i can accept. In contrast to the public utility optimization, the optimization at this level is inherently non-altruistic or private. The objective of a node i is to choose $\vec{R}_i$ so that its private utility $U_i^b : \mathbb{R}_+^{n_i} \to \mathbb{R}$ is maximized, i.e.,

$$(\mathrm{P}i) \quad \max_{\vec{R}_i \in \mathbb{R}_+^{n_i}} U_i^b(\vec{R}_i) \quad := \quad \sum_{j \in \mathcal{N}_i} T_{ji} \log(1 + r^{\star}_{ji}), \qquad (6)$$

subject to $\sum_{j \in \mathcal{N}_i} R_{ji} \leq Q_i$, where Q_i is the total receiving capacity; $r^{\star}_{ji}$ is the optimal solution attained at (PPi). The log function indicates that a node intends to maximize its own level of satisfaction by choosing an appropriate level of request. The request capacity is imposed to prevent excessive incoming traffic as a result of high level of requests.

In a collaboration network, each node i responds to other nodes by choosing optimal propagation rates $\vec{r}_i$ and request rates $\vec{R}_i$. The two-level optimization problem leads to a game-theoretic framework $\mathbf{G} := \langle \mathcal{N}, \{\vec{r}_i, \vec{R}_i\}_{i \in \mathcal{N}}, \{U_i^r, U_i^b\}_{i \in \mathcal{N}} \rangle$. In [5], we have shown the existence of Nash equilibrium that satisfies the property that $r^*_{ij} = R^*_{ij}$, $\forall i, j \in \mathcal{N}$.

4. EVALUATION

We simulate a network of n nodes. Each node $i \in \{1, 2, \cdots, n\}$ is labeled with an expertise level $e_i \in [0, 1], \forall j \in \mathcal{N}$, which is the probability that a rule propagated by node i is effective for intrusion detection. Note that the higher the expertise level, the higher the trust value. Each node i contributes detection rules to the network following a Poisson distribution with an average arrival rate $\widehat{r}_i$. T_{ij} is learned by j through past experiences using the Bayesian learning method described in [5]. The rule propagation follows the two-level game design described in Section 3. In this section, we show some selected results on propagation efficiency, incentive compatibility, fairness, and robustness of the system.

Fig. 3 shows the propagation efficiency for both the mailing list and SMURFEN system. We define the propagation efficiency to be the percentage of useful rules that nodes receive. We see that when using the SMURFEN system, the information qualities received by both the low-expertise and the high-expertise nodes are significantly improved compared to the mailing list method. The high-expertise nodes receive higher quality rules than low-expertise nodes, which reflects the incentive-compatibility of the system.

Fig. 4 shows that uniform gossiping provides no incentive to nodes with higher trust values. On the other hand, the best neighbor propagation scheme provides incentive but no fairness. Nodes of the same trust values may have very different return benefit. This is because under the best neighbor mechanism, nodes form collaboration groups. Nodes of the same trust value may join different groups. Since the return benefit largely depends on which group a node belongs to, nodes with the same trust values may have significantly different return benefit. On the contrary, SMURFEN has a continuous concave utility on the return benefit over trust values. It ensures incentive compatibility as well as fairness.

Fig. 5 is to demonstrate the robustness of the system in the face of insider denial-of-service attacks. We can see that the influence of a node is bounded in the system. This is because the SMURFEN system enforces propagation agreements between each pair of nodes. Each node sets a rule propagation limit to all its neighbors using the two-level game (see Section 3). Therefore, when a node intends to launch a DoS attack, the amount of rules it is allowed to send to others is bounded by the limits set by its neighbors. Nodes sending excessive traffic to neighbors will be revealed as potential malicious nodes, and thus removed from the neighbor list of others.

5. CONCLUSION

We have introduced a peer-to-peer rule sharing framework called SMURFEN for collaborative intrusion detection and used a game-theoretic model for its protocol design. We have shown that our system effectively improves the system-wide intrusion detection accuracy, and has the properties of incentive compatibility, fairness, scalability, and robustness to denial-of-service attacks. By simulation, we have corroborated these important CIDN properties. As future work, we intend to show system robustness to different insider attacks.

6. REFERENCES

[1] Snort. `http://www.snort.org/` [Last accessed in July 6, 2011].

[2] A. Broder and M. Mitzenmacher. Network applications of bloom filters: A survey. *Internet Mathematics*, 1(4):485–509, 2004.

[3] J. Goodall, W. Lutters, and A. Komlodi. I know my network: collaboration and expertise in intrusion detection. In *ACM conf. on Computer supported cooperative work*, 2004.

[4] I. Stoica, R. Morris, D. Karger, M. Kaashoek, and H. Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. In *ACM SIGCOMM*, pages 149–160. ACM, 2001.

[5] Q. Zhu, F. Fung, R. Boutaba, and T. Başar. A game-theoretic approach to rule sharing mechanism in networked intrusion detection systems. In *Proc. of 50th IEEE CDC and ECC, To appear*, 2011.