A Trust-based Approach to Mitigate Rerouting Attacks

Jesus M. Gonzalez[1], Mohd Anwar[2], James B.D. Joshi[3]

Graduate Program in Telecommunications and Networking, University of Pittsburgh, PA, USA
jmg93@pitt.edu, manwar@pitt.edu, jjoshi@sis.pitt.edu

Abstract— **One of the ways a malicious router can launch a Denial of Service (DoS) attack is by rerouting IP-packets of other destinations to the victim node. In this paper, based on the observed traffic anomalies, we propose using a Markov chain model to calculate trustworthiness of routers in order to isolate the malicious ones. Furthermore, our approach reduces the false positives by including context information, such as traffic congestion and packet corruption. By means of simulation, we validate our proposed approach in both connection-oriented (i.e., TCP) and connection-less (i.e., UDP) environments.**

Index Terms— **Rerouting Attacks, Denial of Service Attacks, Markov chain, Trust, Autonomous System.**

I. Introduction

Communication, business, education, financial transactions, and many day-to-day activities nowadays rely on computer networks. Much effort has been put into protecting underlying network infrastructures that make such activities possible. However, many threats still exist in such network environments. One of such threats is the Denial of Service (DoS) attacks [1]. The key goal of a DoS attack is to disrupt or block accesses to services. One of the factors that make DoS attacks so easy to launch is that attackers do not need to own or compromise entire cyber physical infrastructure to generate all the required traffic to block accesses to a service. Attackers who do not own the required equipment first compromise the necessary number of routers to carry out the attack [2, 3, 4]. By compromising a router, the attacker has access to far more traffic than what it could generate by compromising end-hosts.

In this paper, we are interested in developing a *reputation management* (*RM*) system that can efficiently calculate the trustworthiness of routers in order to isolate the ones that perform *rerouting attacks*. By rerouting attack, in this paper, we mean an attack where a malicious router intentionally diverts the packet to a node that is not the original intended destination. Some approaches to address rerouting attacks have been proposed in the literature. In [1,5] authors acknowledge the severity of rerouting attacks and propose an application-layer approach. The approach, however, is computationally expensive. Our proposed approach first looks for anomalies in the way connections are established or looks at the actual destination of the traffic. The actual destination and the intended destination are not the same in a rerouting attack [10]. Once an anomaly is detected, the end-host in collaboration with the non-malicious routers in the path identifies the malicious routers and isolates them. The malicious routers are identified by using probe packets. Based on the results of probing, we model three states: 1) *modified state*, where the destination IP addresses of the probes are modified, 2) *not-modified state*, where the destination IP addresses of the probes are not modified, and 3) *unknown state* when no acknowledgement for the probes are received. With the results of the probes, the state transition probabilities are fed into a Markov chain to evaluate the trustworthiness of each router in the path. When the steady state of the Markov chain is calculated, a router can be labeled as *trusted* or *nontrusted*. When a router is assessed as nontrusted, it is no longer used to forward packets.

The rest of this paper is organized as follows. In Section II, we present an overview of our proposed detection approach. Our proposed trust-based approach is presented in section III. Simulation results are presented in section IV. In Section V, we present conclusions and future work.

II. Overview of the proposed Detection Approach

We use an anomaly-based detection system that resides on the end-host. The proposed detection method can be divided into two complementary cases:

1) Connection-oriented Traffic: If the traffic being sent requires an acknowledgement (i.e, TCP traffic) and the sender never receives such acknowledgements or receives just a few of them, a bad router is suspected. A particular case of redirected traffic is when malicious nodes try to launch a TCP-based DoS attack. Any TCP server can only handle a finite number of TCP connections at a time. Further, TCP connections require a three-way handshake before the end-hosts start to exchange information. Then, if a router modifies the destination IP-address of the handshake packets the connection would be established with an unwanted destination, which would be detected by the source end-host, consequently detecting the rerouting attack. One important

[1] Jesus M. Gonzalez is a Graduate student at the TeleNet Department, University of Pittsburgh, Pittsburgh, PA 15260, e-mail: jmg93@pitt.edu.

[2] Mohd Anwar is a Research Assistant Professor at the School of Information Sciences, University of Pittsburgh, Pittsburgh, PA 15260, email: manwar@pitt.edu.

[3] James B.D. Joshi is an Associate Professor at the TeleNet department, University of Pittsburgh, Pittsburgh, PA 15260, e-mail: jjoshi@sis.pitt.edu.

COLLABORATECOM 2011, October 15-18, Orlando, United States

DOI 10.4108/icst.collaboratecom.2011.247157

note is that if the malicious router also modifies the source address in an attempt to get around our detection mechanism, it would be detected with our previous approach on IP-spoofing detection [9].

2) Connectionless Traffic: If the traffic does not require an acknowledgement (i.e. *UDP* traffic) the source end-host does not have a way of knowing whether the traffic is being received at the other end or not. However, if the end-host detects unexpected traffic coming from any source it can easily notify the sender that that traffic is unexpected. Similarly, if the malicious router modifies the source IP-address, it can be detected by our previous work [9].

When bad router is detected the source end-host initiates a process to identify the malicious router. For routing purposes within an Autonomous System (*AS*), the most widely used routing protocol is the Open Shortest Path First (*OSPF*) [11]. Therefore, in this paper we use *OSPF* as the routing protocol.

III. The Proposed Trust-Based Approach

Here, we are interested in developing a reputation management (*RM*) system that can effectively calculate the trustworthiness of routers. Using the trust values, we isolate the malicious routers so as to detect and mitigate the effects of the IP-rerouting attacks. Once an end-host has determined that there is a malicious node within a routing path following steps are taken:

A. Evidence Collection

In this step, we gather evidence to determine which routers can be trusted or nontrusted. This process is best explained with an example: assume that the suspicious path is as shown in Fig 1:

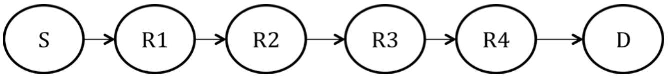

Fig. 1. One of multiple possible paths for IP traffic from source S to destination D.

Where:

S	is the source end-host.
R1...*R4*	are the intermediate routers.
D	is the destination end-host.

The process is as follows:

1) S sends a set of probe packets to *R2.* Each packet requires a reply. When a reply is requested for a packet the destination simply switches the source and destination *IP* addresses and sends an echo packet. Three cases can happen: a) the source *IP*-address is the expected one and the packet is labeled as *not-modified*, b) the source IP-address is not the expected one and the packet is labeled as *modified*, and c) the packet is not acknowledged. *S* waits for a period of time to receive all the acknowledgements and then looks at the transitions between the consecutive replies. The results of the transitions are then fed to a Markov chain to determine whether *R1* is trusted. If *S* has determined that *R1* is trusted then *S* sends another set of probe packets to *R3*, using the same path, to calculate whether *R2* can be trusted. However, if *S* has judged that *R1* is not trusted then *S* looks for a different path to test *R2.* Recent networks require redundancy in the paths. The same process is repeated for the remaining routers, until all of them have been evaluated. One limitation is when no alternate path exists to reach a router (*R3* in this case) when the previous node is been classified as non-trusted. We address this issue in future research

B. Trust Calculation

We propose to use a Markov chain model to determine the probability of the steady trust-state for each router in the path. Our main goal is to evaluate the router's trust value based on its historical behavior, which can accurately determine which routers should be avoided to keep a trusted environment.

It was proven in [8] that an ergodic Continuous-time Markov chain (CTMC) model can be used to determine the trust value by obtaining the unique steady-state probability vector. We first define variables used throughout the section in Table 1.

TABLE I

Variables used throught the section

Variable	Meaning
X_t,	a random variable that represent the state of a Markov chain at any time step *t.*
x_t	the observed state at time *t*. and. $x_t \in$ {modified, not modified, unknown}
t	time step $t \in T$
T	The set of all time steps
P	The transitioning probability matrix
${p^{(n)}}_{yx}$	The probability of transitioning from state *x* to state *y* in *n* time steps.
π_t	State transition probability distribution at time step *t*

A Markov chain is a mathematical system that models probabilistic transitions from one state to another, with a finite number of possible states, in a chainlike manner. Trust can be defined as a stochastic process that is a Markov chain is at state *x* at time *t* $\{x_t \colon t \in T\}$, where any time $\mathrm{t_i} \in \mathbb{R}_0^+$, with $t_0 < t_1 \ldots < t_n < t_{n+1}$, $\forall \mathrm{n} \in \mathbb{N}$. In line with the Markov chain model, we assume that the state sojourn time (i.e. the amount of time spent at any state) of the trust states in the Markov chain is exponentially distributed, and therefore is memoryless, Then, the probability of transition from state *y* to state *x* in *n* steps is given by the Chapman-Kolmogorov equation:

$$p_{yx}^{(n)} = \sum_{r \in S} p_{yr}^{(n)} p_{rx}^{(n-k)} \tag{1}$$

In our approach, the transition probabilities are determined based on the results obtained from the probes in section 4.2. The transition probabilities are increased or decreased as more probes are sent and more acknowledgements are received. From Fig. 2, we can see that the Markov chain is forced to transition at every time-step (may transit to the same state) and there is not absorbing states, then $\Sigma p_{y,x} = 1$. This means that the homogeneous model is irreducible and has the initial-state independent property, i.e, there exists a π_{j}, such that:

$$\pi_x = \sum_{i \in S} \pi_i P\left(X_{t+1} = x \mid X_t = y\right) \tag{2}$$

where π_{i} is an equilibrium distribution of the Markov chain.

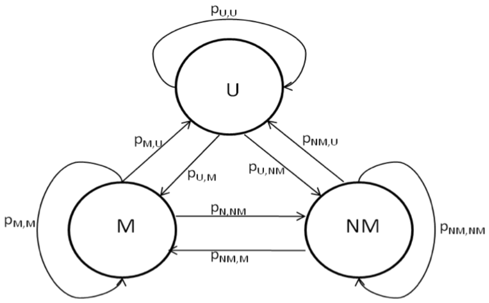

Fig. 2. The three-state CTMC used in trust calculations, where: M = modified state, NM= not-modified state, U=unknown state, and $p_{y,x}$ = the transition probabilities.

Also, because of the unpredictable nature of a malicious router the state transitions are considered to be aperiodic.

In equation (2), $\pi^{(0)}$ is the initial state vector; hence, the state vector at time step 1 is given by:

$$\pi^{(1)} = \pi^{(0)} P\left(X_{t+1} = x \mid X_t = y\right) \tag{3}$$

In general, for time step n the state vector is given by:

$$\pi^{(n)} = \pi^{(0)} \left(P\left(X_{t+1} = x \mid X_t = y\right)\right)^n \tag{4}$$

However, by using equation (4) predictions of states on more distant time steps become increasingly inaccurate and tend towards a steady state vector. This vector represents the probability of each state at all time steps and is independent of the initial state distribution [12]. For a more accurate calculation of the steady state, we use the formulae given in [12] and define the steady state as:

$$q = \lim_{n\to\infty} \pi^{(n)} \tag{5}$$

Because a CTMC is memoryless q is independent from initial state and it must be unchanged when transformed by P. This makes it an eigenvector with an eigenvalue 1, it means it can be derived from P. P is the transition matrix: $qP = q$, then $q = qI$

Subtracting q from both sides and factoring then yields:

$$q(P - I) = 0 \tag{6}$$

Then, by solving the resulting stochastic matrix we can calculate the steady state vector q.

C. Trust Decision

A router is considered to be *trusted* if the Markov chain calculates that the router will stay in the *not modified* state with a high probability when it reaches a steady state, this probability becomes the trust value. Similarly, it is considered to be *nontrusted* if the Markov chain calculates that the router will stay in the "modified" state with a high probability. When the calculated trust value for one of the routers in a suspicious path falls below a certain threshold the trust value is lowered to zero; after a random period of time the trust value is restored to *trusted* and the same test is performed. In case the trust value falls below the threshold again the trust value is lowered to zero again and it is kept like that for an exponentially increased period. The process is repeated.

IV. Evaluation

A. Simulation Setup

To capture the IP-rerouting scenario as explained in *Section III.B* we use the topology shown in Fig. 3.

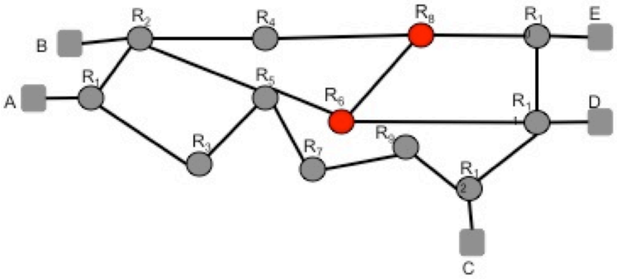

Fig. 3. The network topology used for simulation runs.

In the topology R_6 and R_8 are two malicious routers that are rerouting traffic towards end-host *C*, which is not the intended destination. Nodes *A* and *B* are two non-malicious end-hosts that want to establish connections with end-hosts *D* and *E,* respectively. A hundred packets are sent every time an end-host needs to perform trust evaluation over each router in a suspicious path. To maintain a realistic scenario, we also model other reasons that could influence the results, such as, traffic congestion and packet corruption. In the simulation there is a 7% probability that the router drops a packet because of traffic congestion and a 3% probability that the packet is dropped because of packet corruption. To measure the performance of our approach, we simulate each end-host attempting to open ten *TCP* connections at each time step even though it is significantly more than that the number of connections in real situations. Every time a connection establishment fails the end-host retries up to 5 times before giving up. For *UDP* traffic each end-host transmits at a Constant Bit Rate (*CBR*) of 2Kbps. Finally, we use the number of hops as the metric for the routing protocol (*OSPF*).

B. Results

In the following results, the behavior of Routers *R6* and *R8* are malicious. To avoid detection the routers intermittently reroute packets instead of constantly rerouting the packets.

Fig. 4 (a) shows the trust values calculated using the Markov chain. Our algorithm also specifies that if the trust value falls below a threshold then the Markov chain remains at *modified* state. In other words, the router is not trusted and its trust value is reduced to zero. Fig. 4 (a) shows how end-host *A* and *B* evaluate trust over router *R6* and *R8,* respectively. When the Markov chain calculates *R6* and *R8* to be not trusted enough to be part of the path then nodes *A* and *B* recalculate the paths to exclude the non-trusted routers.

Fig. 4 (c) shows that during the period of time when our approach keeps both *R6* and *R8* (both malicious nodes) as *trusted,* the number of successfully established connections decreases, even after five retransmission attempts. Recall that our approach allows for trust recovery by fully trusting the node after certain period of time and then recalculating the trust value. It also shows that when the routers are kept as *non-*

trusted, all the connections are successfully established. Similarly Fig. 4 (c) shows that when R6 and R8 are trusted the number of retransmissions increase and when they are non-trusted the retransmission is reduced. Note that the number of retransmissions is never zero; this is because we are including traffic congestion and packet corruption. When using *UDP* the only relevant metric for evaluating the performance of our approach is the amount of traffic that is rerouted towards the end-host *C*. Fig. 5(a) shows that most of the traffic (measured in bits) is rerouted towards node *C* in the presence of the malicious routers. On the other hand, when our approach isolates routers *R6* and *R8* the traffic that is rerouted towards node *C* is considerably reduced.

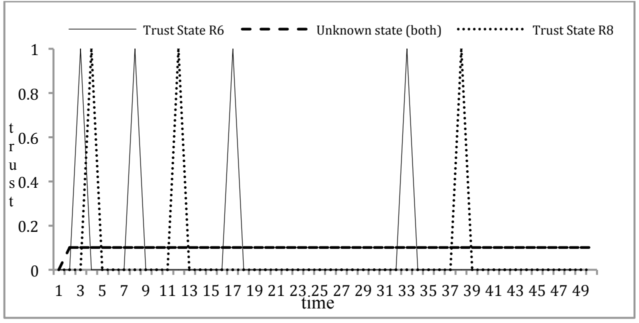

Fig. 4 (a) shows trust values for R6 and R8 as evaluated by hosts A and B respectively.

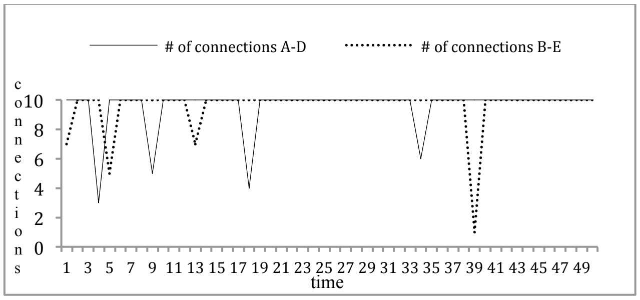

Fig. 4 (b) shows the number of established connections

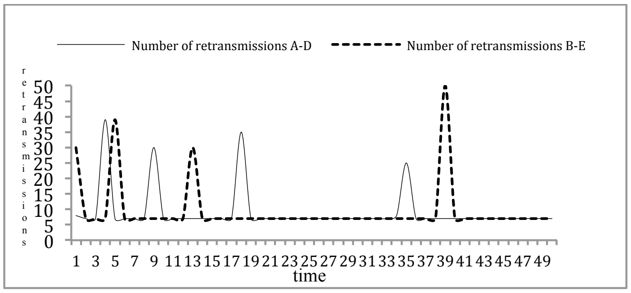

Fig. 4 (c) shows the number of retransmissions

V. Conclusions and Future Work

In this paper, we presented a trust-based approach to detect and isolate routers that reroute IP packets to an unintended destination. Our proposed approach achieves three objectives: *a)* isolates the malicious routers as demonstrated by simulation results; *b)* the amount of overhead is considerably low and depends mostly on the number of routers in the path; *c)* reduces the number of false positives by including context information, such as traffic congestion and packet corruption. For future research we plan to include cooperation with other AS to give our approach a more far-reaching scope.

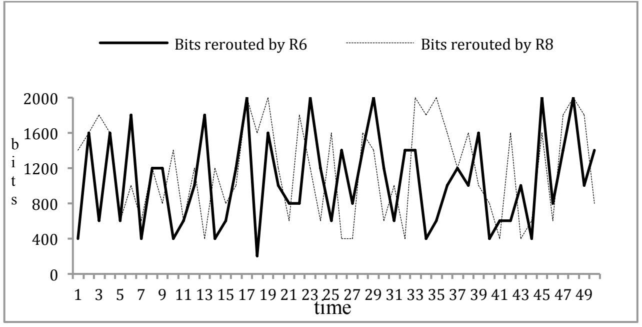

Fig. 5 (a) amount of redirected traffic before our approach is introduced.

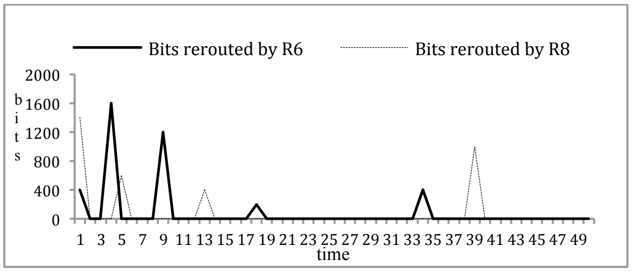

Fig. 5 (b) amount of redirected traffic aftero our approach is introduced.

Acknowledgment

James Joshi has been supported by the US National Science Foundation award IIS-0545912.

References

[1] Chen, X., Li, S., Ma, J., Li, J., "Quantitative threat assessment of denial of service attacks on service availability," *Computer Science and Automation Engineering (CSAE), 2011 IEEE International Conference on,* vol.1, pp.220-224, 10-12 June 2011.

[2] Mizrak, A., Cheng, Y., Marzullo, K., Savage, S., "Fatih: detecting and isolating malicious routers," *Dependable Systems and Networks, 2005. DSN 2005. Proceedings. International Conference on,* pp. 538- 547, 28 June-1 July 2005.

[3] Harris, J., Hill, R., "StaticTrust: A Practical Framework for Trusted Networked Devices," *System Sciences (HICSS), 2011 44th Hawaii International Conference on,* pp.1-10, 4-7 Jan. 2011.

[4] Chen, C., Chang, C., "A two-tier coordinated defense scheme against DDoS attacks," *Computer Science and Service System (CSSS), 2011 International Conference on,* pp.148-151, 27-29 June 2011.

[5] Chen, J., Wang, W., Zhuge, B., Dong, L., "A Method of Bandwidth Allocation Mechanism in ForCES Transport Mapping Layer," Networks, 2007. ICON 2007. 15th IEEE International Conference on, pp.48-52, 19-21 Nov. 2007.

[6] Marti, S., Giuli, T., Lai, K., Baker, M., Mitigating routing misbehavior in mobile ad hoc networks," *International Conference on Mobile Computing and Networking (MobiCom 2000)* pp. 255–265., 2000

[7] Zouridaki, C., Mark, B., Hejmo, M., Thomas, R., "A Quantitative Trust Establishment Framework for Reliable Data Packet Delivery in MANETs," *Proc. 3rd ACM Wksp. Sec. Ad Hoc and Sensor Networks,* Alexandria, VA, Nov. 7, 2005.

[8] Chang, B., Kuo, S., Liang, Y., Wang, D., "Markov Chain-Based Trust Model for Analyzing Trust Value in Distributed Multicasting Mobile Ad Hoc Networks," *Asia-Pacific Services Computing Conference, 2008. APSCC '08. IEEE,* pp.156-161, 9-12 Dec. 2008.

[9] Gonzalez, J., Anwar M., Joshi, J., "A trust-based approach against IP-Spoofing attacks," *ninth annual conference on privacy, trust and security*, Montreal 2011.

[10] Ngadi, M., Rasheed, K., Satria, M., "A review current routing attacks in mobile ad-hoc networks." *International Journal of Computer Science and Security,* 2 (3). pp. 18-29. 2008.

[11] Perlman, R., "A comparison between two routing protocols: OSPF and IS-IS," *Network, IEEE* , vol.5, no.5, pp.18-24, Sep 1991.

[12] Gross D., Shortle, J., Thompson J., Harris, C., Fundamentals of Queuing Theory, Wiley 2008.