Swarm Intelligence Routing Resilient to Byzantine Adversaries

Baruch Awerbuch David Holmer Herbert Rubens *

Abstract

An ad hoc wireless network is an autonomous self-organizing system of mobile nodes connected by wireless links where nodes not in direct range communicate via intermediary nodes. Routing in ad hoc networks is a challenging problem as a result of highly dynamic topology as well as bandwidth and energy constraints. The Swarm Intelligence paradigm has recently been demonstrated as an effective approach for routing in small static network configurations with no adversarial intervention. These algorithms have also been proven to be robust and resilient to changes in node configuration. However, none of the existing routing algorithms can withstand a dynamic proactive adversarial attack, where the network may be completely controlled by byzantine adversaries.

The routing protocol presented in this work attempts to provide throughput competitive route selection against an adversary which is essentially unlimited; more specifically, the adversary benefits from complete collusion of adversarial nodes, can engage in arbitrary byzantine behavior and can mount arbitrary selective adaptive attacks, dynamically changing its attack with each new packet. In this work, we show how to use the Swarm Intelligence paradigm and Distributed Reinforcement Learning in order to develop provably secure routing against byzantine adversaries. Preliminary simulation results are presented.

1 Introduction

The motivation for this work is to design robust routing for wired overlay networks or mobile ad hoc wireless networks (MANET's). Although routing in ad hoc wireless networks has unique aspects, many of the security problems faced in ad hoc routing protocols are similar to those faced by wired networks. Wireless security issues are more acute, due to the inherent vulnerabilities of a wireless environment and are the focus of this paper; however our ideas are applicable to wired overlay networks as well.

In general, routing protocols are susceptible to a wide variety of attacks. A malicious node may advertise false routing information, try to redirect routes, or perform a denial of service attack by engaging a node in resource consuming activities such as routing packets in a loop. Furthermore, due to their cooperative nature and the broadcast medium, ad hoc wireless networks are more vulnerable to attacks in practice [9].

A great deal of work has been done in terms of guaranteeing practical security considerations in existing network protocols. In practice, adversarial attacks observed and documented in ad hoc networks might not be overly sophisticated. The ease of access to the medium has allowed extremely basic attacks to cause a great deal of damage. Consequently, such attacks can be thwarted by simple yet effective methods. For example: a mis-routing attack can be easily detected by authenticating the packet path; consistent traffic blocking can eventually be detected; a single adversary that does not collude can be detected by its neighbors; trusted servers or sensors an be used to monitor truthfulness of link state databases, etc. For each of these important special cases, a lot of great work was done, which is extremely important in practice. However, existing work does not come anywhere close in either addressing the sophisticated attacks arising in our dynamic adversarial model, or attempting to prove analytical bounds on packet loss.

Our Contribution: The goal of this paper is to design an on-demand flooding-free routing protocol in a dynamic byzantine adversarial environment. We propose a generic framework for rout-

*Authors are with the Computer Science Department, Johns Hopkins University, Baltimore, Maryland. {baruch, dholmer, herb}@cs.jhu.edu

ing protocols which are appropriate for networks operating under this extremely strong adversarial model. Such strong models have not been considered in the literature to the best of our knowledge. In fact, one does not even need to consider the full power of byzantine attacks. Even relatively benign adaptive dynamic denial of service attacks are already sufficient to break most existing algorithmic work.

What is remarkable about our result is the ability to prove *near-optimality* bounds under *completely arbitrary* adversarial behavior, with essentially no assumptions about either the network, or the underlying security infrastructure. We use techniques similar to the "Swarm Intelligence" and Distributed Reinforcement Learning paradigm. Swarm Intelligence is a set of learning and biologically-inspired approaches to solve hard optimization problems using distributed cooperative agents. Motivation comes from work which explored the behaviors of ants and how they coordinate each others selections of routes based on a pheromone secretion.

As in [15], one can imagine a model of network routing, such that the network is populated by artificial ants (packets) that make use of the trail laying principle; at each node an ant encounters on the journey to its destination, it leaves an amount of pheromone which evaporates with time, but is an increasing function of the frequency of traversal of that location. The ant then selects the next node on its journey on the basis of the local pheromone distribution [15]. The routing decision is determined by these pheromone distributions.

In our Byzantine routing approach, we act similarly: the process of route detection and fault avoidance is carried out by a distributed process of "learning" fault free paths, in spite of deceptive techniques pursued by adversaries. The routing process creates and adjusts a probability distribution at each node for the node's neighbors. The probability associated with a neighbor reflects the relative likelihood of that neighbor forwarding and eventually delivering the packet to the destination.

2 Major ideas of our algorithm

The algorithm we present is executed at each source node with respect to a specific destination. The actual data packets are source routed and the source route is protected using an onion encryption technique. Every source is maintaining its own graph and probabilities of reaching specific destinations. This information is not shared between nodes because of the byzantine adversarial model which is assumed. Source nodes only rely on other nodes in the network to forward packets and return acknowledgments. It is also assumed that the nodes may choose not to forward the packets or not to return acknowledgements and the source adjusts its probability distribution accordingly. It is important to note that this approach does not rely on intermediary nodes to make hop by hop routing decisions as in other approaches, since those approaches are sure to fail under this strong model. The following description provides an overview of the major ideas of our algorithm.

The main idea of this algorithm is the use of a "Distributed Reinforcement Learning" technique. Specifically, each node is attempting to pick a parent edge towards the source. The set of all parent edges forms a tree rooted at the source. The packets are sent on the unique path in this tree from the sender (the root) to the receiver, and are being acknowledged by the receiver. However, a potential failure may decompose the path into two parts: the part closest to the source that succeeded to acknowledge the packet, and the rest of the path that failed to return an acknowledgement. We now adjust the choice of parent edges as follows. The part of the path that succeeds in acking reinforces its confidence in the parent, while the other part of the path reduces its confidence. The parent will be chosen probabilistically based on the confidences acquired.

The intuition is that confidence in the parent reflects not only the reliability of the link between child and parent, but also the fact that the parent is "intelligent" enough to pick the right (grand)-parent. In this context, there is no way to distinguish byzantine nodes from nodes that are unlucky in choosing their parent. Notice that by authenticating all messages, byzantine nodes are limited in their ability to mislead the source since some edge

controlled by the adversary must be exposed to the source, each time an edge fails. The adversary can choose *not to* expose itself, and indeed this complicates the proof somewhat. We need to show that the adversary has nothing to gain from taking control of an edge, and then not killing packets on this edge. By using a probabilistic distribution over the parent edges we generate a probability distribution over all source-rooted trees, such that probability of each tree is growing exponential in its performance. This discriminates edges controlled by the adversary, and "reinforces" edges where the adversary is absent. More intuition can be obtained by work on non-stochastic multi-armed bandit [1] and its distributed analog in [2].

3 Simulation Results

In order to substantiate the claims made in this work, simulations were conducted to investigate the convergence time of the algorithm. The simulations were conducted by developing a simple program which would simulate the decision making process of the algorithm and examine its performance against adversarial inputs. The simulation consisted of a source selecting a path to the destination at each unit of time. An adversarial model would then select which nodes at the current unit of time were faulty. The packet would traverse the graph and receive positive feedback from the destination if there were no faulty nodes on the path, or from the last non-faulty node before the packet was dropped. Using this feedback the algorithm would adjust its probabilities and compute a new path for the next packet.

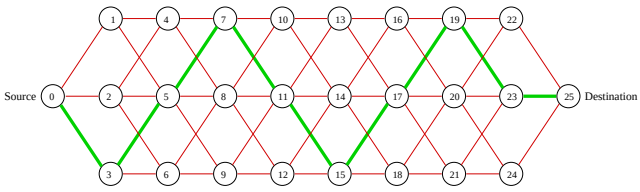

Figure 1: Simulation Topology

In this example we consider a network with 25 nodes forming a 10 layered graph, with 3 nodes at each layer (except at the source and destination). The topology of the network is indicated in Figure 1. In this example there is exists one optimal path from the source to the destination which experiences no loss. This optimal path is indicated in Figure 1 by the bold line. All other links in the network exhibit 10% loss, meaning that when we sample them they will successfully forward the packet 90% of the time. This network topology contains 1394 possible paths of which only one is optimal and the the others have total loss rates ranging from 19% to 61% depending on how many optimal links they contain. This setup should be challenging for our algorithm due to the large number of total paths, and due to the fact that the non-optimal links in the sampled path only experience marginal loss.

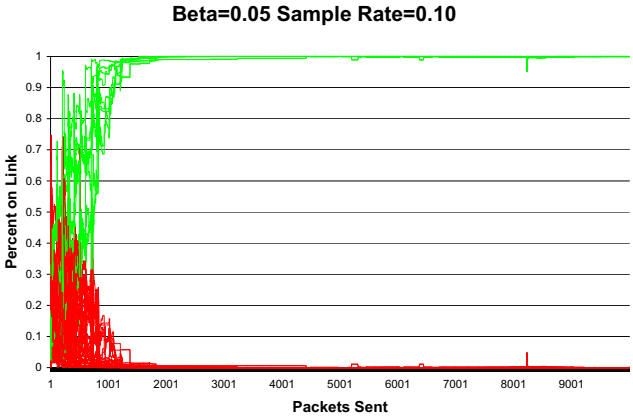

Figure 2: Simulation Results

This simulation consisted of a source attempting to deliver 10,000 packets to the destination. The graphs in Figure 2 shows the results using the algorithm parameters of 10% random sampling and a β of 0.05. This graph consists of a line for each link in the network. The lines representing the optimal links are colored in green and the lines representing the the non-optimal links are red. The vertical axis of the graph represents the probability that our algorithm will select the given link to be part of the path selected to send the next packet.

The results indicate that the algorithm begins to shift towards using the optimal links immediately, and is able to almost fully converge to the optimal path after approximately 1000 packets are sent. Once the algorithm has learned the best path it is able to send approximately 99% of its traffic successfully to the destination. The graph visually indicates this by showing the sources link preferences at every unit of time. When the simulation begins the source considers all of the links in the network to be equal and then learns their relia-

bility by sending traffic across the links and receiving feedback. As the number of packets (or trials) increases the sources knowledge of the network continuously becomes more accurate. This is evident as the reliable paths become separated from the less reliable paths and selected with near 100% probability. Since the source is continuously sampling the less desirable edges it is able to respond quickly to changes in the adversarial fault pattern.

4 Related Work

Interest in applications of ant-based routing in MANETs has risen and many recent papers have addressed the subject [11, 3]. Gunes et al [12] considers ant-based approach to routing in MANETs, with a completely reactive algorithm. Marwaha et al. [14] studies a hybrid approach using both AODV and reactive Ant-based exploration. Baras et al [3] describes a new algorithm that utilizes the inherent broadcast nature of wireless networks to multicast control and signaling packets (ants). ARAMA [11] uses analogous approach. Work on Swarm Intelligence is described in [10, 5, 16, 7, 6, 13, 15, 4, 8, 11, 3].

5 Conclusion

In this work we have presented an online algorithm which is based on the Swarm Intelligence paradigm and Distributed Reinforcement Learning approach. Through mathematical analysis and simulation results we have shown the algorithms competitive performance under a strong adversarial model consisting of dynamic proactive adversarial attacks, where the network may be completely controlled by byzantine adversaries.

The results of this work indicate validity of our approach and motivate the need for future work in this direction. We intend on implementing this protocol in a more realistic simulation environment and exploring the effects of both mobility and more sophisticated active adversarial attacks on the algorithm.

References

[1] Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. Gambling in a rigged casino: the adversarial multi-armed bandit problem. In *Proceedings of the 36th Annual Symposium on Foundations of Computer Science*, pages 322–331. IEEE Computer Society Press, Los Alamitos, CA, 1995.

[2] Baruch Awerbuch and Yishay Mansour. Online learning of reliable network paths. In *PODC*, 2003. to appear.

[3] John S. Baras and Harsh Mehta. Dynamic adaptive routing in manets. In *Procedings of Annual ARL CTA Symposium*, 2003.

[4] E. Bonabeau, F. Henaux, S. Guérin, D. Snyers, P. Kuntz, and G. Theraulaz. Routing in telecommunications networks with "smart" ant-like agents. In *Intelligent Agents for Telecommunications Applications '98 (IATA'98)*, 1998.

[5] G. Di Caro and M. Dorigo. AntNet: a mobile agents approach to adaptive routing. Technical Report IRIDIA/97-12, Université Libre de Bruxelles, Belgium.

[6] Gianni Di Caro and Marco Dorigo. Antnet: Distributed stigmergetic control for communications networks. *Journal of Artificial Intelligence Research*, 9:317–365, 1998.

[7] P. Druschel D. Subramaniam and J. Chen. Ants and reinforcement learning : A case study in routing in dynamic networks. In *Proceedings of IEEE MILCOM, Atlantic City, NJ*, 1997.

[8] Marco Dorigo, Vittorio Maniezzo, and Alberto Colorni. The Ant System: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics*, 26(1):29–41, 1996.

[9] Yih-Chun Hu, Adrian Perrig, and David B. Johnson. Ariadne: A secure on-demand routing protocol for ad hoc networks. Technical Report TR01-383, Department of Computer Science, Rice University, December 2001.

[10] M. Littman and J. Boyan. A distributed reinforcement learning scheme for network routing. In *Proceedings of the International Workshop on Applications of Neural Networks to Telecommunications. Alspector, J., Goodman, R. and Brown, T. X. (Ed.)*, pages 45–51, 1993.

[11] U. Sorges M. Gunes and I. Bouazizi. The ant colony based routing algorithm for manets. In *Proc. of the 2002 ICPP Workshop on Ad Hoc Networks (IWAHN 2002)*, pages 79–85.

[12] U. Sorges M. Gunes and I. Bouazizi. ara - the ant colony based routing algorithm for manets. In *Proc. of the 2002 ICPP Workshop on Ad Hoc Networks (IWAHN 2002)*, pages 79–85.

[13] O. E. Holland R. Schoonderwoerd and J. L. Bruten. Ant-like agents for load balancing in telecommunications networks. In *Proc. First ACM International Conference on Autonomous Agents, Marina del Rey, California*, pages 209–216, 1997.

[14] C. K. Tham S. Marwaha and D. Srinivasan. Mobile agents based routing protocol for mobile ad hoc networks. In *in Proceedings of IEEE Globecom,*, 2002.

[15] Ruud Schoonderwoerd, Owen E. Holland, Janet L. Bruten, and Leon J. M. Rothkrantz. Ant-based load balancing in telecommunications networks. *Adaptive Behavior*, (2):169–207, 1996.

[16] Devika Subramanian, Peter Druschel, and Johnny Chen. Ants and reinforcement learning: A case study in routing in dynamic networks. In *IJCAI (2)*, pages 832–839, 1997.