Security of Emergent Properties in Ad-Hoc Networks

Virgil D. Gligor
Electrical and Computer Engineering Department
University of Maryland
College Park, Maryalnd 20742

Extended Abstract

The notion of *ad-hoc networking* refers to the spontaneous formation of a network of computing nodes without reliance on a specific infrastructure. Typical ad-hoc networks consist of mobile nodes — ranging from laptops and pocket PCs to miniature sensor devices — with widely different computation and communication capabilities. Node mobility may be limited to deployment, as in the case of some sensor networks, or may continue throughout the lifetime of the network, as is the case in mobile ad-hoc networks (MANETs). Ad-hoc networks may also be formed by application-level software agents executing on top of a fixed communication infrastructure. In such networks, node mobility can take the guise of software-agent migration among different hosts, or even of autonomous peers (e.g., principals, domains) joining or leaving peer-to-peer (P2P) networks formed without the benefit of a separate, fixed collaboration infrastructure.

Ad-hoc networks emerge through the collaboration of every node with its neighbors which usually involves sharing resources, communication links, and applications. For example, in mobile ad-hoc networks, a topology emerges as every node exchanges information with its neighbors to establish connectivity and forwards packets to its neighbors toward chosen destinations to establish the routing infrastructure. The basic infrastructure thus created is highly dynamic not only because of node mobility but also because of lack of guaranteed network connectivity. For instance, sensor-node deployment cannot usually guarantee direct connectivity between all pairs of nodes due to sensor scattering. Further, the node membership to the network that emerges after deployment may change in time because sensor nodes exhaust their power and are replaced by new nodes that are scattered in the same geographical area. In contrast, in ad-hoc networks formed at application levels of the Internet, membership is typically unaffected by unreliable link-layer communication or by power exhaustion. Instead, dynamic node-membership changes can be caused by potentially unanticipated joins and departures of autonomous nodes; e.g., joins and departures of mobile software agents in an application-level network, or of autonomous domains in a P2P network. At these levels, collaboration infrastructures must be created and changed by the network nodes themselves usually based on a dynamically-negotiated sharing of objects and applications.

A common characteristic of all ad-hoc networks is that of *emergent properties*. Intuitively, emergent properties are features that cannot be provided by individual network nodes themselves but instead result from interaction and collaboration among network nodes. Although one may think of the creation of an ad-hoc network as a set of emergent connectivity and routing properties, our primary focus is on the specific properties that may emerge after the ad-hoc networks are thus established. The emergent properties and their security characteristics we propose to study are different from traditional network properties established via protocol interactions in several fundamental ways. First, it is possible that neither the time nor the locus of emergence of these properties can be easily anticipated. Second, the emergence of these properties may be uncertain, in the sense that it may be probabilistic. Third, these properties may be transient, in the sense that they may disappear from the ad-hoc network during normal operation and not as a result of exceptional events; e.g., node or protocol failures.

Although not identified as emergent in the past, several important properties of concurrent and distributed systems share some characteristics of emergent properties in ad-hoc networks. For example, the place and time of emergence of some non-transient safety properties, such as deadlocks in distributed systems [1], cannot be anticipated and their emergence is uncertain; i.e., some detected deadlocks may, in fact, be false [17]. Similarly, process starvation in some systems with distributed control [10, 11] and many denial of service instances [18, 19, 42, 33] also share some characteristics of emergent properties. Emergent properties have also been identified in system and network security, primarily in the area of policy composition [28, 23, 32, 22, 30]. For instance, even under simple criteria such as conjunction the composition of non-discretionary policies yields new properties that are not present in the individual policies under composition [29, 41, 20]. In contrast with the extensive research on such properties in distributed systems, there is relatively little work on the analysis of such emergent behavior in ad-hoc networks, and even less focusing on the security implications thereof.

Why Security of Emergent Properties ?

Security analyses of emergent properties in ad-hoc networks is essential for several reasons. First, many emergent properties represent new features of ad-hoc networks that are fundamental to ensuring network security and robustness. For instance, these properties may be used to help establish or revoke trust relations that are necessary for both node and message authentication. Second, some emergent properties are undesirable and may lead to security violations and, for this reason, their detection and handling help maintain the security and robustness of the network. For example, emergent properties may indicate (in a manner which bypasses traditional intrusion-detection techniques) that a certain network node is captured by an adversary. Third, inadequate assessment of emergent properties such as false detection may also lead to security and robustness violations. For example, the false detection of node capture may lead to network partitioning and denial of service in an ad-hoc network by the unnecessary revocation of node membership. Fourth, the understanding of the characteristics of emergent properties helps determine the scalability and resilience of ad-hoc networks. For example, desirable emergent properties (such as establishment of secure communication paths in sensor networks via random key pre-distribution) may place constraints on the network size but may also imply resilience of network communications below a certain threshold of compromised nodes [8]. To illustrate the distinguishing characteristics of emergent properties and their security implications, we present several examples of desirable and undesirable emergent properties in different types of ad-hoc networks.

Example 1. Trust Establishment in Mobile Ad-hoc Networks.

We view the notion of trust establishment as the application of evaluation metrics to a body of (trust) evidence. An outcome of trust establishment is a trust relation. A simple example of a trust relation that needs to be established arises in the context of determining whether some certification authority's signature on a public-key certificate asserting an <identity, public key> or <identity, attributes> association represents sufficient evidence for the validity of that certificate. In traditional public-key infrastructures, the trust relation that helps guarantee the validity of such a certificate is established between a certificate user and the signing authority, either directly (e.g., at user registration) or indirectly (e.g., by evaluating multiple trust relations that form a trust path between the certificate user and the signing authority) [24, 39]. In contrast, in mobile ad-hoc networks where public-key infrastructures may not exist or may not be accessible due to connectivity limitations, trust relations may need to be established among nodes after network emergence. Here, every node could potentially become a "certification authority," yet the identities, attributes, and configuration properties of the connected nodes may remain un-verified until network emergence. Hence, trust establishment has to be based on dynamic evaluation of signed evidence about a node (e.g., location, identity, and configuration attributes) and not just on statically defined relationships at, or prior to, network emergence (e.g., on static determination of a node's trustworthiness derived from its design and implementation characteristics). This means that neither the time nor the locus of emergence of a trust relation can be anticipated, the emergence of trust relations becomes uncertain, and trust relations may be transient. Furthermore, the dynamic establishment of trust relations must be accompanied by the dynamic detection of undesirable trust relations that may also emerge. It is possible, and indeed very likely, that network nodes captured by an adversary will introduce false evidence about trust-relation emergence and revocation. Hence, the design and use of evidence-evaluation metrics must be able to assign very low certainty to evidence assessed to be questionable while still achieving an acceptable number of false positives [14].

Example 2. Establishing Secure Communication Paths in Distributed Sensor Networks.

Distributed Sensor Networks (DSNs) are ad-hoc networks formed after large numbers of nodes with very limited computation, memory, and RF communication capabilities are scattered across potentially hostile environments where, e.g., communication may be monitored and nodes may be captured and surreptitiously used by an adversary. DSNs are dynamic in the sense that they allow addition and deletion of sensor nodes after deployment to grow the network or to replace failing and unreliable nodes [9]. Although DSNs require cryptographic protection of communications, use of traditional public-key cryptography is usually not possible due to its excessive energy requirements. Also, pairwise sharing of symmetric keying material between all network nodes is ruled out by the sensor-memory size constraints anticipated for the foreseeable future, by the inability to communicate directly with all nodes, and by the need to confine communication compromise to small neighborhoods of captured nodes. To date, all practical keying schemes require the pre-distribution of a small number of encryption keys to sensor nodes prior to deployment [15, 8]. All such schemes rely on a classic emergent property of random graphs (namely, the result of Erdös and Renyì [12, 37] which assures a path between any pair of nodes with high probability after DSN deployment) to establish secure node-to-node paths with a small number of shared keys per sensor node. Although anticipated, this emergent property is probabilistic and its stability can only be guaranteed until a certain number of sensor nodes fail (e.g., due to battery exhaustion) or are captured by an adversary. Hence, the disappearance of secure node-to-node paths must be detected to enable confinement of secure communication failures and compromises to small subnetworks.

Example 3. Establishing Common Access States in Dynamic Coalitions.

Dynamic coalitions are peer-to-peer networks of autonomous domains that share resources to enable the execution of common applications. The shared resources and their permissions are said to form the *common access state* of the coalition members. Dynamic coalitions require repeated, on-line negotiations to establish a common access state [21]. Since coalition membership varies dynamically, off-line, one-time negotiation of sharing agreements on common access states are ruled out. Further, these negotiations are different from typical client-server trust negotiations [36, 40], since these negotiations have different goals, are peer-to-peer rather than client-server, and cannot necessarily be expected to terminate (successfully or not) in a fixed number of rounds. The negotiation of resource contributions to the common access state may be conducted by each domain under different types of constraints, some of which cannot be revealed to other members. The satisfaction of all individual constraints and the emergence of a common access state by access negotiation is a desirable property. However, the emergence of the common access state may have unintended, undesirable properties such as the unintended revealing of private resources or the private negotiation constraints of a domain. Reaching a common access state in a dynamic coalition has the typical characteristics of an emergent property. Although the locus of emergence can be anticipated, the time and certainty of emergence cannot be guaranteed. Furthermore, the stability of the common access state can only be guaranteed until a membership change (i.e., the dynamic departure or join of one or more domains).

Example 4. Establishing Secure Routes in Mobile Ad-hoc Networks using Swarm Intelligence

The notion of swarm intelligence has been used for a variety of routing problems [6, 7, 35], the most recent of which is routing in mobile ad-hoc networks with resource-constrained nodes that change topology frequently and have unpredictable connectivity [4, 31]. In such routing applications, autonomous agents simulating ants discover routes between network nodes and construct a trail of local route information in each node representing ants' pheromone deposits. As with ant behavior, these autonomous agents proactively explore and reinforce available routes by updating and propagating routing-table information among network nodes in an attempt to accurately reflect changing traffic conditions and network connectivity. As a result of the agent interaction via local route information, the shortest path between two nodes emerges. Routing algorithms based on swarm intelligence can be executed in parallel and are scalable, fault-tolerant, and adaptive. However, capture of mobile routing agents by an adversary leads to the denial of routes, while injection of corrupted agents leads to the autocatalytic proliferation of false routes (i.e., routes which always include the adversary's nodes). Hence, in mobile ad-hoc networks where nodes are subject to capture by adversaries, it becomes important to detect emergent false routes and malicious routing agents, and assess their impact. Secure route establishment using swarm intelligence has all the typical characteristics of emergent properties: namely, neither the time nor the locus of (real or false) route emergence is anticipated, and route emergence is uncertain and transient.

The common theme of most research in the security of emergent properties of ad-hoc networks is that of a threat model whereby the adversary may compromise nodes in the network. Unlike most previous research on ad-hoc networks which limits an adversary to "man-in-the-middle" attacks and relies on end-to-end security solutions to counter such attacks [34], our research extends the threat model in two ways. First, we allow that the adversary can, in fact, be one or more of the network nodes. This is clearly the case since ad-hoc network nodes (e.g., sensors, mobile agents) often operate unattended in hostile environments. Second, we allow that the impact of the adversary's actions may extend to network nodes beyond those captured, and that the adversary's actions cannot be detected by traditional network intrusion detection techniques. This is true because emergent properties result from collaboration among multiple nodes and hence a few miscreant nodes may affect others. It is also a necessary assumption because the communication patterns of an adversary-operated node — which traditional network intrusion detection techniques focus on — may be indistinguishable from those of a legitimate node. Although the resilience of different protocols to captured client or server devices [26] and to Byzantine behavior of participants [25, 16] has been the subject of prior work, the scope and nature of an adversary's actions must be expanded. For instance, in contrast to the usual client-server protocols, in ad-hoc networks it is often impractical to design protocols that would neutralize the adversary's actions so that other nodes are unaffected. Furthermore, the behavior of an adversary in this environment is not necessarily Byzantine; e.g., it is not necessarily directed toward the disruption of a multi-party agreement protocol. Instead, the adversary may seek to corrupt data that compromises the entire application in an undetectable manner. In fact, the adversary's captured nodes may collaborate and execute network protocols ipretending to be legitimate network nodes aiming to sabotage network operation. In this setting, node-to-node and message authentication are insufficient to counter such attacks.

References

[1] B. Alpern and F. Schneider. Defining Liveness. *Information Processing Letters* 21(4): 181-185, 1985.

[2] AntHill Simulation Tool, http://www.cs.unibo.it/projects/anthill/download.html.

[3] O. Babaoglu, H. Meling, and A. Montresor, "Anthill: A Framework for the Development of Agent-Based Peer-to-Peer System," Technical Report UBLCS-2001-09, University of Bologna, Italy.

[4] J. S. Baras and H. Mehta, "A Probabilistic Emergent Routing Algorithm for Mobile Ad Hoc Networks," in Proc. of the Conf. on Modeling and Optimization in Wireless, Mobile, and Ad Hoc Networks (WiOpt 03), Sophia-Antipolis, France, March 2003.

[5] R.B. Bobba, L. Eschenauer, V.D. Gligor, and W.A. Arbaugh. Bootstrapping Security Associations for Routing in Mobile Ad Hoc Networks. Institute for Systems Research, Technical Report 2002-44, May 2002. Available at http://bellatrix.isr.umd.edu/TechReports/ISR/2002/TR-2002-44/TR-2002-44.pdf.

[6] E. Bonabeau, M. Dorigo and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*, Santa Fe Institute on the Sciences of Complexity, Oxford University Press, July 1999.

[7] Di Caro, G. and M. Dorigo, "AntNet: Distributed Stigmergetic Control for Communication Networks", *Journal of Artificial Intelligence Research*, Vol 9 pp 317-365, 1998.

[8] H. Chan, A. Perrig, and D. Song, "Random Key Predistribution Schemes for Sensor Networks," Proc. of the IEEE Security and Privacy Symposium, Berkeley, CA, May 2003 (available at *http://www.ece.cmu.edu/˜adrian*).

[9] D. W. Carman, P. S. Kruus and B. J. Matt,"Constraints and Approaches for Distributed Sensor Network Security", dated September 1, 2000. NAI Labs Technical Report No. 00-010, available at http://download.nai.com /products/media/nai/zip/nailabs-report-00-010-final.zip

[10] E. W. Dijkstra. A Class of Allocation Strategies Inducing Bounded Delays Only. Proc. SJCC 1972, vol. 42, AFIPS Press, Montvale, N.J., pp. 933-936.

[11] E. W. Dijkstra. Self Stabilization in Spite of Distributed Control. *Comm. ACM*, vol. 17, pp. 643-644, 1974.

[12] P. Erdös and A. Renyì. On the evolution of random graphs, Magy. Tud. Akad. Mat. Kut. Intez. Kozl. 5 (1960), 17–61.

[13] L. Eschenauer, J. S. Baras and V.D Gligor, "Distributed Trust Establishment in MANETs: Swarm Intelligence," Proc. of 2003 Collaborative Technology Alliance Conference, April 2003.

[14] L. Eschenauer, V.D Gligor, and J.S. Baras, "On Trust Establishment in Mobile Ad-Hoc Networks", in *Security Protocols*, Christianson *et al.* (eds.), Cambridge, UK, April 2002. To appear in Lecture Notes in Computer Science, Springer-Verlag, 2003. (available at *http://www.ee.umd.edu/˜gligor*)

[15] L. Eschenauer and V.D. Gligor, "A Key-Management Scheme for Distributed Sensor Networks," Proc. of the ACM Conference on Computer and Communications Security, Washington D.C., November 2002, pp. 41-47.

[16] M. Franklin and R. Wright. "Secure Communication in Minimal Connectivity Models," *J. Crypto* 13(1): 9–30 (2000).

[17] V.D. Gligor and S. Shattuck, "On Deadlock Detection in Distributed Systems," *IEEE Transactions on Software Engineering*, SE-6, Vol. 5, (September 1980).

[18] V. D. Gligor, "A Note on Denial of Service in Operating Systems," *IEEE Transactions on Software Engineering*, SE-10, No. 3, (May 1984).

[19] V.D. Gligor, "On Denial of Service in Computer Networks," Proc. of Int'l Conference on Data Engineering, Los Angeles, California, February 1986, pp. 608-617.

[20] V.D. Gligor and S. I. Gavrila, "Application-Oriented Security Policies and Their Composition," in *Security Protocols*, B. Christianson, B. Crispo, M.Roe (eds.), Lecture Notes in Computer Science 1550, Springer Verlag, 1999, pp. 67-75.

[21] V. D. Gligor, H. Khurana, R. K. Koleva, V. G. Bharadwaj, and J. S. Baras. On the negotiation of access control policies. In B. Christianson *et al.*, editors, *Security Protocols* in Lecture Notes in Computer Science, vol. 2467, pp. 188–201, Springer Verlag, 2002. Also see transcript of discussion, pp. 202-212.

[22] Hinton, H.M., and E.S. Lee. The Compatibility of Policies. Proc. ACM Conference on Computer and Communications Security, 1994, pp. 258-269.

[23] D. Johnson, and E.J. Thyer, "Security and the Composition of Machines," Proc. of the Computer security Foundations Workshop, Franconia, N.H., June 1988, pp. 72-89.

[24] B. W. Lampson, M. Abadi, M. Burrows, and Edward Wobber. "Authentication in Distributed Systems: Theory and Practice." *ACM Transactions on Computer Systems*, 10(4): 265–310, 1992.

[25] N. Lynch. *Distributed Algorithms*, Morgan Kaufmann Publishers, 1997.

[26] P. MacKenzie and M. Reiter, "Network Cryptographic Devices Resilient to Capture," Proc. IEEE Security and Privacy Symposium, Berkeley, California, May 2001 (and updated version in DIMACS TR 2001-19, May 2001).

[27] U. Maurer, "Modeling a Public-Key Infrastructure." in Proc. ESORICS '96 (4th European Symposium on Research in Computer Security), Rome, LNCS 1146, Springer-Verlag, Berlin 1996, 325–350.

[28] D. McCullough. Specifications for Multilevel Security and a Hook-Up Property. Proc. IEEE Symposium on Security and Privacy, 1987, pp. 161–166.

[29] D. McCullough. Noninterference and Composability of Security Properties. Proc. IEEE Symposium on Security and Privacy, 1988, pp. 177–186.

[30] J. McLean. A General Theory of Composition for a Class of "Possibilistic" Properties. *IEEE Transactions on Software Engineering* 22(1): 53–66, 1996.

[31] H. Mehta, "Dynamic Adaptive Routing in Mobile Ad Hoc Networks," M.S. Thesis, University of Maryland, College Park, December 2002.

[32] J. Millen, "Hookup Security for Synchronous Machines," Proc. of Computer Security Foundations Workshop, Franconia, New Hampshire, June 1990, pp. 84-90.

[33] J. Millen. "A Resource Allocation Model for Denial of Service." Proc. IEEE Security and Privacy Symposium, 1992, pp. 137–147.

[34] J. H. Saltzer, D.P. Reed, and D.D. Clark, "End-To-End Arguments in System Design," in *ACM Transactions on Computer Systems*, vol. 2, no. 4, Nov. 1984, pp. 277 - 288.

[35] R. Schoonderwoerd, O.E. Holland, J. Bruten, L. Rothkrantz, "Ant-based load balancing in telecommunications networks", HP Labs Technical Report, HPL-96-76, May 21, 1996.

[36] K. Seamons, M. Winslett, and T. Yu, "Limiting the Disclosure of Access Control Policies during Automated Trust Negotiation", In Proceedings of the Network and Distributed System Security Symposium, San Diego, CA, April 2001.

[37] J. Spencer, The Strange Logic of Random Graphs, *Algorithms and Combinatorics* 22, Springer-Verlag 2000, ISBN 3-540-41654-4

[38] Swarm Development Group, http://www.swarm.org

[39] E. Wobber, M. Abadi, M. Burrows, and B. Lampson, "Authentication in the Taos operating system," ACM Transactions on Computer Systems, 12(1):3–32, Feb. 1994.

[40] W. Winsborough, K. Seamons, and V. Jones, "Automated Trust Negotiation," DARPA Information Survivability Conference and Exposition (DISCEX '2000), January, 2000.

[41] Zakinthinos, A., and E. S. Lee, "A General Theory of Security Properties," Proc. of 1997 IEEE Symposium on Security and Privacy, Oakland, California, May 1997, pp. 94-100.

[42] C.-F. Yu and V.D. Gligor, A Specification and Verification Method for Preventing Denial of Service, *IEEE Transactions on Software Engineering*, Vol. SE-16, No. 6, June 1990, pp. 581 - 592.