Using Cryptography in Trust Computing for Networked Communications

Dang-Quan Nguyen, Louise Lamont

Communications Research Centre
Ottawa, Ontario, K2H 8S2, Canada

Abstract

We highlight some major difficulties encountered by current approaches that try to model trust computing in a realistic networked communications system. We characterize these approaches as *top-down* since they assume that trust is universal and readily quantifiable. Our main concern with these approaches is that their quest to define a universal trust often ends up with a loose, context-dependent definition of trust value. Based on this shortcoming of the *top-down* approaches, we propose another consideration of trust for networked communications. Namely, we underline the fact that trust is first and foremost a security constraint that exists in each specific network's security operation (*i.e.*, attack and defense). Hence, its definition depends on the attack and defense being involved. We call this approach *bottom-up* and discuss the close relationship between trust and cryptography through some examples of network's attack and defense.

Keywords: *Trust computing, Cryptography, Security, Networked communications.*

1. Introduction

We present our position regarding a fundamental question about *trust*, considered as a component of security in networked communications: ``*how to efficiently and creatively capture the concept of trust computing in networked communications?*''

Throughout this paper, our objective is two-fold. First, we highlight some major difficulties encountered by the current approaches that try to model trust computing in a realistic networked communications system. We characterize these approaches as *top-down* since they assume that trust is universal and readily quantifiable. Hence they often rely on applications such as Intrusion Detection Systems as a determinant to provide a trust value for various devices in the network. Also, they tend to tailor the underlying networking mechanisms (such as admission control, routing protocols, etc.) to accommodate the constraints induced by trust. In this case, trust is often considered as a network parameter, along with the end-to-end throughput and delay, and is accounted for as such. Our main concern with these approaches is that their quest to define a universal trust often ends up with a loose, context-dependent definition of trust value. Moreover, from a practical point of view, it is quite unclear what kinds of network attack these approaches aim to resolve.

Based on this shortcoming of the *top-down* approaches, we propose another consideration of trust for networked communications. Namely, we reverse the above process and underline the fact that trust is first and foremost a security constraint that exists in each specific network's security operation (*i.e.*, attack and defense). Hence, trust does not necessarily have a universal definition, but its definition depends on the attack and defense being involved. Therefore, the implementation of trust is automatically embedded within each basic component of the security operations. Trust in this case adds a new dimension of consideration to the existing security operations and also offers guidelines to the design of a new paradigm. We call this approach *bottom-up*.

As an example, consider a large MANET whose topology is not guaranteed to be continuously connected (such as in sparse networks). The nodes are soldiers on the ground. When soldiers from a coalition force meet and they have no connectivity to the authentication server, and if they need to share some valuable information then soldiers from both sides will have to decide whether they can trust each other based on a shared passphrase. However, revealing a secret passphrase to an unknown soldier is dangerous. Therefore, the soldiers may need an exchange protocol that allows for the mutual verification that both sides actually *know* the secret but without revealing anything on the secret itself, not even a hash of the passphrase since an attacker can reuse the hash values to fool other soldiers. *Top-down* approaches cannot solve this problem other than waiting for the connection with the authentication server to come up. Reputation and recommendation cannot be used since both soldiers are unknown to each other prior to the meeting. We will see in the detailed discussion of Section

3.3 that this problem can be solved at the low level networking mechanisms using a cryptography protocol called *zero knowledge proof*.

To narrow down the scope of security operations that we can afford to work on, while maintaining a subject rich in illustrations and promising in exciting research topics, we limit our security operations to cryptography and cryptanalysis mechanisms that are used to secure and to break the networked communications. We also discuss how the *bottom-up* approach releases the dependence of trust on the network components and blends itself well to distributed systems such as mobile ad hoc networks. We also provide concrete examples and highlight the link with trust in each example.

The rest of the paper is organized as follows. In Section 2, we formulate the issue of trust computing and its importance in secured communications. Section 3 presents the existing proposals of trust computing schemes that we call *top-down* approaches. We argue, by presenting some counter-examples, that *top-down* approaches may not be pertinent. We then present our position on trust, based on the observation that its importance must be granted to solve low-level problems (via some practical attacks on networked communications). Section 3 also shows the close connection between cryptography and trust computing. We conclude this paper in Section 4.

2. Issue and its Importance

Future land operations will be undertaken by geographically dispersed teams in order to gain a better understanding of the battle-space through information gathered by the dispersed teams. In certain situations, a dispersed force must be capable of rapid aggregation in order to conduct operations as a larger aggregated force. The constantly changing nature of the battle-space necessitates adaptive forces equally capable of operating in a dispersed or aggregated posture. Additionally in order to gain greater mission effectiveness, information sharing will extend beyond the land force to allies and joint forces. More than ever we need to develop a framework that can be used to secure network communications against attack.

The trust component is an important concept in network security, as it is the set of relations among agents participating in the network activities. Trust becomes even more challenging in wireless multi-hop and distributed networks where soldiers join and leave the network to support dispersion and rapid aggregation. Trust management is often interpreted as a multifunctional control mechanism, in which the most important aspect is to establish trust from a small set of agents who are known to be trustworthy. Another important aspect in trust management is trust revocation, which reverses previous trust opinions of agents based on newly obtained evidence with regard to those agents. Trust propagation can also be seen as another component of trust management where a source node must rely on other nodes to forward its packets on multi-hop routes to the destination.

We argue that the fundamentals techniques of cryptography and cryptanalysis need to be refined to ensure that nodes can exchange information once they are trustworthy without starting as a premise that a set of nodes are trustworthy from the first encounter. Furthermore, security in computer networks usually relies on central authorities, certificates directories, or some preinstalled keys and procedures. However, the past decade has witnessed the emergence of Mobile Ad hoc Networks (MANETs) which are characterized as self-organizing systems. These centralized services may not always be accessible in self-organized systems. To address this problem, we propose to develop a trust management scheme where key distribution, key revocation and enforcement are tackled by using light-weight cryptography techniques and key distribution protocols that use broadcasting with selective encryption to guarantee that only a subgroup of members can decrypt the message.

3. Trust Computing and Cryptography

In this section, we first present some existing proposals of trust computing schemes. We characterize these approaches as *top-down* since they consider trust as universal and readily quantifiable by an existing system such as intrusion detection system (IDS). Counter-examples are presented to show that *top-down* approaches may not be pertinent to solve practical security problems involving trust. Based on this observation, we then introduce our *bottom-up* approaches consisting of considering trust as a security constraint proper to each networked communication attack or defense mechanism. Thus, we show that cryptography can be employed to solve practical problems of trust computing. At a lower level, we also discuss how this combination of cryptography and trust computing can provide inputs to other security mechanisms such as admission control, key management and secure routing.

3.1 Existing proposals on the issue

Currently, the existing approaches to trust computing in networked communications consider *trust* as a quantifiable

network metric. We classify existing trust computing proposals into two categories: low-level trust computing focussed on the evaluation of trust metrics and how to propagate such metrics in the network; and high-level trust computing focussed on the creation of frameworks and policies for trust computing.

3.1.1 Low-level trust computing

Low-level trust computing proposals include schemes designed for trust evaluation and trust propagation.

For trust evaluation, the proposals rely on an existing monitoring entity having the ability to observe and to compare nodes' behaviors to report any wrong doings or policy infringement, Cf. [1, 4, 13, 19, 20, 15, 9].

The proposals usually assume that the monitoring entity is an intrusion detection system (IDS) or a watchdog. While these systems are available in the resourceful and centralized networks such as the Internet, it may not be possible to implement them in distributed systems such as MANETs which have very strong constraints on the network resources. For example, IDSes are centralized and heavily rely on the pre-established behavioral rules, the efficiency of IDSes are yet to be proven in highly dynamic networks. In order to prevent serious attacks at low level networking, policy-based IDSes might not be adaptable or sufficiently responsive.

For trust propagation (Cf. [8, 7, 21, 6]), distributed networks such as MANETs require trust metrics to be forwarded on a multi-hop basis. The distant nodes then make assumptions on the degree of the trustworthiness of intermediate recommenders. For example: if node B observes that node C is only 30% trustworthy and node A to node B is 40% then many trust propagation protocols assume that node C is 30%*40% = 12% trustworthy to A. Notice that this multiplicative rule over the trust values assumes that the trustworthiness is independent for the intermediate nodes. This assumption is most likely unrealistic since nodes interact with each other to evaluate their mutual trust values.

To sum up, probabilistic trust evaluation based on behavioral observations relies on many assumptions that may not be true and also on external components that are yet to be proven practical.

3.1.2 High-level trust computing

The existing proposals characterized as high-level trust computing include frameworks or architectures that incorporate trust evaluation, propagation, IDS policies and security components such as secured routing, admission control, etc. (Cf. [10, 11, 12, 18]). High-level trust computing has many advantages since it offers a unified view on trust throughout the network, and thus facilitates security collaborations of nodes from different domains.

However, in this context, trust is often considered as an issue of quality of service rather than one of security. It also becomes less obvious from the high-level point of view how the frameworks can practically be used to secure communications against attacks that happen at the low-level of the network, since these attacks are specific to many networking properties: wired/wireless, fixed/mobile, centralized/distributed, etc.

To illustrate that the existing *top-down* proposals may not be best suitable for trust computing, we present in the next section some counter-examples in which the network may need other considerations of trust computing to defend itself from some practical attacks.

3.2 Counter-examples for *top-down* trust computing

In the literature of trust computing for networked communications, *top-down* approaches extensively rely on behavioral monitoring to evaluate the trustworthiness of a node. Typical malicious behavior that has been discussed includes *selfishness* and *packet dropping*.

The example that has often been used to illustrate node selfishness is the following. Assuming a distributed communication system in which nodes use cognitive radios to advertise their transmission load and to make reservation of the medium access. A node may behave selfishly by falsely declaring that it has a high amount of priority data to transmit or to forward. Since there is no centralized entity to coordinate the reservation, there will be no easy way for each node to verify the veracity of all the claims of bandwidth prior to the actual transmissions. A trust system would then rely on the monitoring of the actual transmissions for verification. This means each node has to put its network interface into promiscuous mode in order to capture all the transmission around it, decode all the packets and analyze them. This monitoring is usually referred to as *watch dog*. There are many inconveniences of doing so in a distributed, dynamic network.

Firstly, there is no doubt that keeping the network interface in a permanent wake-up mode, in order to decode and process all packets in the transmission range is a very costly operation. In particular, doing so will consume a great deal of the node's battery power. It is worth pointing out that every practical MAC design seeks to put the network interface into idle mode whenever possible to save energy. Therefore, keeping the network interface in

permanent wake-up mode will most certainly not gain wide acceptance.

Secondly, every peer-to-peer transmission in a secured network should be encrypted by some encryption algorithms to ensure that an attacker cannot analyze the network's traffic. Allowing nodes in the network to analyze transmissions around them creates a breach in security if we have not already established trust between all nodes. Even a small amount of information on a data packet such as the source, destination addresses and the data priority, if made available, can be exploitable by the attacker.

Thirdly, an inconsistent behavior of a node may accurately be detectable in a wired network, for example: by monitoring the amount of data that actually came out of a link and by comparing this amount to the bandwidth that the node had reserved on this same link. It is not easy to have such detection in a wireless, error-prone, dynamic network. At the beginning, a node has an actual amount of data to transmit to its peer. After the reservation process, this node may not be able to transmit the data because the peer could have moved away, or some environmental factors (such as interference, physical obstacle) could have blocked the signal. Even if this node has transmitted the data correctly as reserved, the detection by measuring the amount of transmitted data can face the same inaccuracies due to the dynamic environmental factors as the transmission itself.

Finally, using the node selfishness to illustrate a security threat that appeals to trust computing could be inadequate, since the node selfishness is more a protocol design problem (in this case a problem of the bandwidth reservation protocol) than a security problem. Fixing the reservation protocol is deemed to be sufficient to address this issue. Note that a similar problem has been found in some implementations of the CSMA/CA protocol that allow users to manually change the value of the contention window (CW) so that they were always the first ones who occupied the medium.

Similar to the node's selfishness example, *packet dropping* is commonly cited as a relevant example of an attack that can be prevented by trust computing. In this attack, malicious nodes drop packets to be forwarded at some specific rates. All nodes monitor the packet loss rates and evaluate probabilistically the trustworthiness of their peers. We argue that this attack is more a case study of quality of service (with delivery rate acting as the principal metric to optimize) than a problem of security. Many QoS routing protocols can deal efficiently with this issue by avoiding the congested area of the network. Also, a serious attacker is more inclined to conceal his misbehavior by performing his routing task well and, at the same time, he tries to copy and to decrypt the information, rather than trigger suspicion by dropping packets for no reasons. There are other more efficient ways to willingly disrupt the network, such as distributed denial-of-service (DDoS), than just dropping packets arbitrarily. Furthermore, the environmental factors explained above indicate that relying on the loss rates to estimate the maliciousness of nodes can lead to wrong trust evaluation. Hence, doing so potentially damages the network even more because the trust component may exclude good nodes from the network.

Since we have shown that trust in these two examples cannot be monitored easily and consistently, the trust system, if it relies on this monitoring, cannot produce an accurate trust evaluation that is consistent with the security expectation.

3.3 Our proposal of *bottom-up* trust computing

Based on the shortcomings of the *top-down* approaches, we argue that trust must be considered with all the characteristics of each security defense and attack mechanism. There should not be one single definition of trust throughout the network, but the definition of trust must lie in the context of each security scenario. We call this approach *bottom-up* since it allows for the consideration of the role of trust in each practical, specific scenario. These scenarios or examples will then serve as building blocks for any collection of security solutions. In particular, cryptography and cryptanalysis contain some excellent examples of the strong connection between trust and communications security.

To clarify *bottom-up* trust computing, we present some security examples that appeal to the concept of trust. We acknowledge that a significant amount of work needs to be done in order to determine if (and how) existing cryptography solutions can be implemented in large networks made of low-capacity (and possibly cheap) communicating devices. This should be the starting point of research in trust computing for networked communications.

3.3.1 The requirement for trust and cryptography solution

Starting from the practical security attack and defense mechanisms, trust computing is required in the following examples:

- **Key management problems (Cf. [2, 3, 16]):**

A key management is composed of mechanisms for key distribution, key revocation and must enforce *forward* and *backward security*.

With *forward security*, when an existing node leaves, either voluntarily or by being forced, a group with a shared group key, the key management protocol needs to securely and efficiently change the group key such that all future communications of the group are kept secret to this node. It is natural to distrust a leaving node.

With *backward security*, when a new node joins a group, even if we can trust this node with all future exchanges, we may not want it to be able to decrypt past communications of the group. Hence a new group key must be generated and distributed.

In this context, the key management protocol must implement broadcasting with selective encryption (Cf. [14]), in order to guarantee that only an authorized subgroup of members can decrypt the broadcast message containing the new group key.

- **Zero-knowledge proofs (Cf. [5]):**

As discussed at the beginning of the paper, this issue arises when two parties wish to establish their trust relationship prior to the exchanges of some valuable information, but neither authentication server nor third-party recommender is immediately available. Then both parties must prove to each other that they actually know a shared passphrase without revealing any other information on the passphrase itself. Until trust is established, revealing any information on the passphrase (such as its hash values) could lead to a breach in security since an attacker can collect the information and replay it later. The zero-knowledge proof protocols allow both parties to exchange the proof of their knowledge without revealing it.

- **Good/bad mouthing, data corruption or, more generally, message integrity (Cf. [17]):**

Good and bad mouthing are typical attacks that threaten the security protocols which use some form of reputation and recommendation to establish communications between distant nodes. Malicious nodes may consistently send good reports about other malicious nodes and bad reports about good nodes. They may also modify reports that they forward on behalf of other nodes.

To deal with this problem, nodes may be required to digitally sign their reports, that is, to identify in an unambiguous way the source of the reports. Also, a digital signature ensures that any modification of the report can be properly detected or invalidate the report.

3.3.2 Practical consideration when using cryptography for trust computing

Difficulties of practical cryptography implementation arise when we place ourselves in the context of working on resource constrained networks. The main obstacles include the overhead in computational power of the device and in bandwidth for the transmissions.

Some constraints also apply for each specific cryptography mechanism, for example: the message to be signed must be large enough to prevent brute-force attacks from recovering the secret key; the ciphertext should be split into several parts and transmitted separately one after another to avoid man-in-the-middle attack. Taking into consideration all these constraints in the real implementation also means demanding a certain level of trust in the low level security mechanisms such as the key length or the strength of the hash functions.

3.4 Inputs to other security mechanisms

As our primary objective is to stay close to practical security scenarios, while aiming for strong impacts at the low level of networking mechanisms, our proposal of using cryptography for trust computing can provide many inputs to the networking mechanisms with security requirements, such as: admission control, routing in multi-hop networks, key management, etc. A few examples can be cited below.

- Zero-knowledge proofs can provide trust collaboration in dynamic, distributed environments where centralized authentication is not always available. Therefore, it offers a possibility to perform admission control in this context.

- Cryptography for selective broadcasting can assist key management systems in renewing keys.

- Digital signature can be used to enforce message integrity and to detect any modifications made to a message. Thus, it provides means to secure routing in multi-hop networks.

4. Conclusions

Trust is an important concept in network security. It becomes even more challenging in wireless multi-hop and distributed networks where dismounted soldiers join and leave the network to support dispersion and rapid aggregation. Trust management is often interpreted as a

multifunctional control mechanism that establishes trust among the participants (or nodes). Traditionally, trust has been considered as a universal metric throughout the network. Nodes try to determine the degree of trustworthiness of other nodes based on the observables. This approach naturally leads to the notion of behavior monitoring and probabilistic estimation based on predefined rules. For example, intrusion detection systems are assumed to play a major role in trust evaluation. Another component of trust management is trust propagation where the nodes relay trust metrics and disseminate them into the network so that distant nodes can also estimate indirectly the trustworthiness of their distant partners.

This view of trust as a single metric, measurable and universal, is simple. However, it has a shortcoming with regards to its applicability. For example, it is not always clearly indicated the kinds of attacks on the network this approach of trust aims at, or is defined for. In practice, most IDSes are centralized and rely on pre-established behavioral rules. Therefore, the efficiency of IDSes in highly dynamic, distributed systems such as MANETs is still an open question. Policy-based IDSes might not be adaptable or sufficiently responsive to the attacks that take place at the low level of the network. Other trust management mechanisms, such as trust propagation, also need to be examined more carefully since their assumptions rely more on qualitative than on quantitative arguments.

Based on the shortcomings of the traditional, *top-down* approaches, we argue that the definition of trust must be found in the context of each networking security scenario. There should not be one single, universal definition of trust in the network but trust should be embedded in each practical, specific networking attack and defense mechanism. These defense scenarios then serve as the building blocks for any collection of security solutions. In particular, quantitative and provable security mechanisms should be the backbone on which trust is built. We show in this paper some typical examples of the strong connection between trust and cryptology. Therefore, we consider trust principally as a requirement, specific to each low level security mechanism. We call this approach *bottom-up*. As our primary objective is to address practical security scenarios and to seek strong impacts at the low level networking mechanisms, our proposal of using cryptography for trust computing can provide many inputs to any collection of security solutions such as admission control, routing in multi-hop networks, key management, etc.

As a promising approach that is positioned half-way between applied and theoretical research on trust computing for communications security, this proposal will need to address the practical challenges posed to the cryptography methods by some resource constrained networks. The main difficulties include a shifting from a centralized infrastructure with heavy key management servers to distributed systems, as well as the overhead in computational power of the devices and in bandwidth for the transmission. Other challenges that are more research-oriented also need to be solved, such as designing new cryptography methods to address the specific requirements of trust in some particular attacks on the network.

Acknowledgments

This work is funded by Defence Research & Development Canada (DRDC).

References

[1] T. Beth, M. Borcherding, and B. Klein, “Valuation of Trust in Open Networks.” In proceedings of the 3rd European Symposium on Research in Computer Security, ESORICS ’94, pp. 3–18, London, UK. Springer-Verlag, 1994.

[2] G. Dini and I. Savino, “S2RP: a Secure and Scalable Rekeying Protocol for Wireless Sensor Networks.” In proceedings of IEEE International Conference on Mobile Adhoc and Sensor Systems Conference, pp. 457–466, 2006.

[3] L. Eschenauer and V.-D. Gligor, “A Key-Management Scheme for Distributed Sensor Networks.” In proceedings of the 9th ACM Conference on Computer and Communications Security, CCS ’02, pp. 41–47, New York, NY, USA. ACM, 2002.

[4] L. Eschenauer, V.-D. Gligor, and J. Baras, “On Trust Establishment in Mobile Ad-hoc Networks.” In proceedings of Security Protocols Workshop, pp. 47–66. Springer-Verlag, 2002.

[5] O. Goldreich, S. Micali, and A. Wigderson, “Proofs That Yield Nothing But Their Validity Or All Languages in NP Have Zero-Knowledge Proof Systems.” J. ACM, pp. 690–728, July 1991.

[6] E. Gray, J.-M. Seigneur, Y. Chen, and C. Jensen, “Trust Propagation in Small Worlds.” In proceedings of the 1st International Conference on Trust Management, iTrust’03, pp. 239–254, Berlin, Heidelberg. Springer-Verlag, 2003.

[7] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins, “Propagation of Trust and Distrust.” ACM Press, pp. 403–412. ACM, 2004.

[8] A. Josang, E. Gray, and M. Kinateder, “Analysing Topologies of Transitive Trust.” In proceedings of the 1st International Workshop on Formal Aspects in Security and Trust FAST’03, pp. 9–22, September 2003.

[9] S. D. Kamvar, M.-T. Schlosser, and H. Garcia-Molina, “The Eigentrust Algorithm for Reputation Management in P2P Networks.” In proceedings of the 12th International

Conference on World Wide Web, WWW '03, pp. 640–651, New York, NY, USA. ACM, 2003.

[10] M. Kinateder and K. Rothermel, "Architecture and Algorithms for a Distributed Reputation System." In proceedings of the 1st International Conference on Trust Management, iTrust'03, Berlin, Heidelberg. Springer-Verlag, 2003.

[11] Y. Lacharite, D.-Q. Nguyen, M. Wang, and L. Lamont, "A Trust-Based Security Architecture for Tactical Manets." In proceedings of IEEE Military Communications Conference. MILCOM 2008.

[12] F. Martinelli and M. Petrocchi, "A Uniform Framework for Security and Trust Modeling and Analysis with Crypto-CCS." Journal of Electron. Notes Theor. Comput. Sci., pp. 85–99, July 2007.

[13] L. Mui, M. Mohtashemi, and A. Halberstadt, "A Computational Model of Trust and Reputation for E-Businesses." In proceedings of IEEE Hawaii International Conference on System Sciences, pp. 188–193, Los Alamitos, CA, USA. IEEE Computer Society, 2002.

[14] D.-Q. Nguyen and L. Louise, "A New Cryptography Scheme for Selective Broadcasting." In proceedings of IEEE International Conference on Information and Computer Networks, ICICN 2011, Guiyang, China, 2011.

[15] D.-Q. Nguyen, L. Lamont, and P.-C. Mason, "On Trust Evaluation in Mobile Ad-hoc Networks." In proceedings of Proceedings of the 1st International ICST Conference on Security and Privacy in Mobile Information and Communication Systems, MobiSec '09, Turin, Italy, 2009.

[16] A. Perrig, R. Szewczyk, J.-D. Tygar, V. Wen, and D.-E. Culler, "Spins: Security Protocols for Sensor Networks." Wireless Networks, pp. 521–534, 2002.

[17] B. Schneier, "Applied Cryptography (2nd Ed.): Protocols, Algorithms, and Source Code in C." John Wiley & Sons, Inc., New York, NY, USA, 1995.

[18] N. Stakhanova, S. Basu, J. Wong, and O. Stakhanov, "Trust Framework for P2P Networks Using Peer-Profile Based Anomaly Technique." In proceedings of the 2nd International Workshop on Security in Distributed Computing Systems (SDCS), ICDCSW '05, pp. 203–209, Washington, DC, USA. IEEE Computer Society, 2005.

[19] Y. Sun, W. Yu, Z. Han, and K.-J. Liu, "Trust Modeling and Evaluation in Ad-hoc Networks." In proceedings of IEEE Global Telecommunications Conference, GLOBECOM '05, St. Louis, MO, USA, 2005.

[20] G. Theodorakopoulos and J.-S. Baras, "Trust Evaluation in Ad-hoc Networks." In proceedings of the 3rd ACM workshop on Wireless security, WiSe '04, New York, NY, USA. ACM, 2004.

[21] C.-N. Ziegler and G. Lausen, "Propagation Models for Trust and Distrust in Social Networks." Information Systems Frontiers, pp. 337–358, December 2005.

Dang-Quan Nguyen is a research scientist at the Communications Research Centre, Ottawa, Canada. He has obtained his Master degree in 2003 and his PhD in 2006, both in Computer Science at University of Paris VI (Pierre and Marie Curie), France. He has undergone various research projects at INRIA (France), Orange's Labs (France) and CRC (Canada) under major government grants and contracts. His research interest includes quality of service in MANETs, trust-based security and cryptography.

Louise Lamont is the Research Manager for the Mobile ad hoc and Sensor Networking Group at the Communications Research centre. In this position Louise is responsible for identifying novel research areas for study at CRC and for proposing, implementing and securing funds for new projects in support of major client requirements such as DND. She manages several state-of-the-art laboratories for the conduct of research as well as technical demonstration in the area of mobile ad hoc and sensor networks. She is also responsible for establishing and maintaining liaison, collaboration and partnership with R&D groups at CRC and with external national and international organizations.