Reliable and Secure Broadcast Communication Over Resource Constrained Systems

Arzad A. Kherani
GM-India Science Laboratory
Bangalore, India
arzad.kherani@gm.com

Skanda N. Muthaiah
GM-India Science Laboratory
Bangalore, India
skanda.muthaiah@gm.com

Smruti Padhy
Dept. of Comp. Sci. & Engg.
IIT Delhi, New Delhi, India
spadhy@cse.iitd.ernet.in

Debojyoti Bhattacharya
GM-India Science Laboratory
Bangalore, India
debojyoti.bhattacharya@gm.com

ABSTRACT

Compute platforms for wireless sensor networks and Vehicle-to-Vehicle (V2V) communications employ random channel access for message transmission and typically suffer from limited processing capability and on-board memory on a per-application basis because of the multiple processes going on in parallel. Appending digital signatures to transmitted messages in such systems increases *information reliability*, but requires an intemperate use of scarce resources, more so with an increased security requirement.

It thus appears imperative to tradeoff security for network performance to conserve scarce resources for a given resource constrained platform. This requires a good understanding of the communication performance of these systems. We observe that resource constraints in these systems induce complex interaction between the security and MAC layers at a node, obviating the possibility of layer specific optimizations to improve system performance for broadcast applications. For example, reducing the channel access probability at the MAC layer reduces collision probabilities, in turn increasing the verification load on the security layer. There is hence a need to take a holistic approach to dimension such systems to improve performance. In achieving this objective, we provide:

- an analytical framework to model these systems with or without an impersonation attacker. This analysis also takes into consideration the possibility of multiple digital signatures being attached to a message, so that the receiver has a choice of verifying one of these.
- a characterization of the stability region of the system.
- an information-theoretic approach towards reliable communication of application data over these systems. We view the combination of security-MAC-security layers as an erasure channel and provide attainable rates, jointly optimizing on sampling rate and channel transmission rate.

Valuetools 2009 October 20-22, 2009 - Pisa, Italy

1. INTRODUCTION AND MOTIVATION

Our work is motivated by two applications of secure wireless systems (a) sensor networks, and (b) vehicle-to-vehicle communication, both of which typically require broadcast authentication and employ IEEE 802.11-like [1] random wireless channel access. These platforms are typically constrained in processing power and onboard storage capacity [2], [3].

In these systems, a packet generated at the application layer of a node is sent to the node's security layer for cryptographic operations, for example, generating/verifying Digital Signatures (DS). At the security layer, the packet contends with other to-be-signed and to-be-verified (from other nodes) packets for the scarce processing and storage resources, possibly resulting in the packet being dropped. After signature generation at the security layer, the signed packet (Figure 1) is passed on to the MAC layer that implements a, possibly CSMA based, random access scheme for wireless channel access. On successful reception at the receiver, the packet is passed on to the receiver's security layer where the packet again contends for storage and processor service. A schematic representation of this process for a two node system complete with the packet flow, per-packet operation and interaction at the various layers is illustrated in Figure 1. Resource constraints at the security layer and at the MAC layer thus introduce complex interaction of the message streams generated and received by a node, accounting for two major sources of information loss:

- Packets dropped at the security layer owing to the finite byte-storage available, and
- Packets not received by the receiver(s) owing to multiple messages transmitted simultaneously by various nodes.

Various remedial steps aimed at improving the end-to-end performance for a given compute platform specification have traditionally been myopic as they attempt at rectifying only one performance bottleneck, without considering the impact of such an attempt on other aspects. For example, TESLA [4] aims at reducing the per-message verification

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2009.7643
http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7643

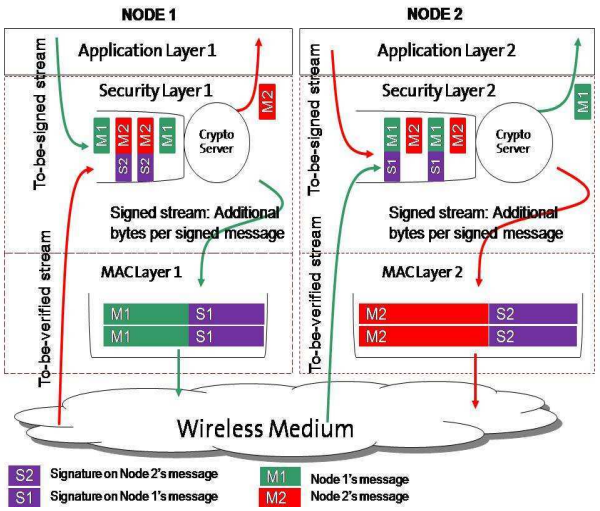

Figure 1: Figure depicting the path of a packet.

time, but requires the receiver to store more messages, thus possibly implying (a) reduced effective storage space at the receiver, and (b) more load on the MAC layer, increasing the collision probability. These intuitions are confirmed in [5, 6].

We advocate adopting a holistic approach at improving the end-to-end performance of secure broadcast communications over resource constrained [1] systems. For any security or MAC layer scheme aimed at improving the end-to-end *performance* of these systems, one should look at its impact on all the layers. Such an effort thus requires efficient broadcast authentication protocols that work well when used in conjunction with the random access MAC. We would like to emphasize that the holistic approach in parameter tuning and protocol selection is *not* same as the cross-layer optimization philosophy where various layers are typically required to exchange information.

Designing efficient broadcast authentication protocols over resource constrained platforms is known to present tradeoffs in seven dimensions [7]: (a) Resistance to node compromise, (b) Low Computation Overhead, (c) Low Communication Overhead, (d) Robustness to packet loss, (e) Immediate authentication, (f) Message sent at irregular times, and (g) High message entropy, with a note that not all of these desired objectives are expected to be achieved simultaneously. The authors did not consider the finite storage capacity available at the security buffer. For us, the *storage overhead* is another such dimension. It is to be noted that *communication overhead* and *storage overhead* are not necessarily proportional, for example, in the case of TESLA [4] one can have smaller communication and computation overheads but at the cost of possibly significant *storage overhead* (and, of course, delayed authentication) due to keeping the to-be-verified messages longer in the security layer buffer. Schemes which can tradeoff one dimension with other(s) are thus required, and [7] indeed provides some of the possibilities. Further, a proper holistic performance analysis of such schemes needs to be done before declaring them fit for the target application.

Towards achieving these objectives of improving the system performance in a holistic manner, a *multiple digital signature* scheme has been proposed in Section 3 as a way to tradeoff *resistance to node compromise*[2] with *computation overhead*. A fixed point approach to quantify the network performance (end-to-end drop probability) of this scheme can be found in Appendix. These explorations again display complex coupling between the security layer and the MAC layer, and provide some counter-intuitive behavior like (a) an *efficient* scheduling scheme at the security layer actually does worse than a simple FCFS scheduling, and (b) an attempt at separating the to-be-signed and to-be-verified traffic could be counter-productive.

In Section B, we characterize the information capacity of these broadcast systems and provide a stability condition for the security and MAC layers in an effort to attain the best achievable end-to-end per-unit-time information transmission capacity of these systems. One-time signatures [7] is then employed to achieve a tradeoff between computation, communication and storage overheads with respect to maximizing the information capacity of these systems. We provide instances where a high verification times provide the best information capacity. Conclusion and future work for this paper are spread across the various relevant sections.

2. SYSTEM MODEL AND ASSUMPTIONS

The various layers of a node are depicted in Figure 1. The description is based on IEEE 1609.x protocol suite since it is representative of the protocol stack that is used for broadcast communication in sensor networks.

Application Layer: The application layer generates broadcast messages to other nodes in the system. In the V2V context, these messages could be *persistent*, conveying position information or *event-based*, triggered in response to a specific action such as Emergency Electronic Brake Light (EEBL), Stopped Vehicle Alert (SVA) etc. In the sensor network context, the *persistent* application captures, for example, the periodic nature of the information exchange for secure routing protocols requiring broadcast communication

[1]The term *Resource Constrained* used in this study is specifically aimed at *computing-platforms* or nodes that have limited computational and storage/memory resources *available to the application under consideration*. This could be the case when the compute platform itself is constrained or because of the other processes sharing the same platform. For example, the VIIC and CAMP consortium have considered 400MHz-processor Denso Wireless Safety Units, WSUs, for V2X applications, requiring around $15-20$ms for $256-$bit ECDSA signing/verification operations. This, along with the expected message generation rate of 10 messages per second, indicates that the security layer could be a severe performance bottleneck. We consider a source of minimum power P, where, $P \geq \lambda(P_S + P_T)(1-P_B) + (N-1)\lambda(1-P_B)^2(1-P_C)(P_R+P_V)$ is available at each node through energy harvesting techniques or other alternate power sources. As such, we neither consider nor model the energy saved due to the use of our proposed scheme. P_S and P_V is the power required to sign and verify messages with appropriate digital signatures, while P_T and P_R are the powers required to transmit and receive a message respectively. λ is the rate at which packets are generated at the application layer.

[2]Our notion of this term is slightly general than that of [7]. We do not necessarily mean that one node compromise impacts on network wide security properties, a notion assumed in [7]. For example, a node compromise could merely imply bad application performance; see Section 3.2.

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2009.7643
http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7643

in sensor networks, and the *event-based* application is an example where a detected event (say, the radioactivity level above a threshold) is broadcast locally for distributed data aggregation/processing. We let t_p denote the mean inter-message-generation time.

MAC Layer: The MAC layer in our analysis is modeled based on the IEEE 802.11p [8] MAC standard and employs random channel access to transmit packets over the wireless channel. We will be working with random access schemes with and without carrier sensing; the use of carrier sensing is known to introduce dependence among the MAC layer buffer occupancy of various nodes [9]. Since we are considering broadcast wireless networks, acknowledgements or retransmission requests are not part of our model and each packet is transmitted exactly once.

Physical Layer: A *single-cell* approach is used to model the physical channel. In a *single-cell* model, constituent nodes can hear every other node and a packet transmission is successful if no more than one node is transmitting in a given time unit. Packet transmissions are expected to take a finite amount of time depending on the packet data size and the channel data-rate. In this model, propagation delays and turnaround times are ignored. This model is used to understand the performance of dense networks so that even a moderate or small application layer message generation rate (large value of t_p) could lead to contention at the MAC and security layers. In the V2V context this is just a congestion scenario where traffic movement is very slow and the number of neighbors seen by any node is approximately constant, so that the network appears to behave like a single-cell to any given node, especially so because of the absence of RTS/CTS handshake and lack of acknowledgements due to the MAC broadcast. In the sensor network context, this again is like assuming a uniform spatial node density. It is to be noted that we would be interested in understanding the impact of high node density (number of interferers per node) on the performance of the system. We do not account for electromagnetic propagation, but believe that extension of our work in that direction is straightforward.

Security Layer: The IEEE 1609.2 standard [10] for V2V communication recommends the use of Elliptic Curve (EC) Digital Signature Algorithms (ECDSA) to authenticate messages at the security layer. The recommendation of 256 bit key for use with ECDSA scheme in [10] is to achieve 128 bit security; in ECDSA-based schemes, X-bit key provides $\frac{X}{2}$ bit security [11]. However, the ECDSA-based schemes are known to be computationally expensive [3], let alone the extra per-message bytes added to a signed message (Figure 1). An X-bit key ECDSA signature size is $2X$ bits in length [10, 12]. It is also known that the computational complexity of ECDSA-based signature generation/verification schemes are exponential in the key size. Owing to the single-cell assumption, we will not be modeling the performance impact of certificate-exchange mechanisms. This could further bring down the performance of such systems, and such a study is clearly a natural extension of our present work.

3. MULTIPLE DIGITAL SIGNATURES

ECDSA-based security operations over resource constrained platforms could lead to the security layer being the performance bottleneck. However we know from [13] that reducing security processing time by the use of efficient broadcast authentication schemes, or possibly faster security processing hardware, may merely shift the performance bottleneck from the security to the MAC layer[3]. ECDSA-based schemes are constrained in the dimension of computation complexity, hence in the light of [7], improving system performance requires us to tradeoff the computation complexity with communication overhead and/or security-related property. One such scheme is proposed in this section.

We propose a security mechanism based on *multiple digital signatures* in which a sender appends K ECDSA-based signatures to outgoing messages. These K signatures use varying key sizes with the property that the bit size of the keys progressively decreases as more signatures are appended. The receiver is required to intelligently choose among the various key sizes while verifying a signed message. This scheme is an instance of the general scheme [14].

Let $s_j^{(i)}, 1 \leq j \leq K$, represent the j^{th} private key used by the security layer of node i, and let $v_j^{(i)}$ be the corresponding public key. The sender appends all the K signatures for each signed message; other intelligent ways of using these K private keys can be considered similarly. Since we are considering a static scenario[4], certificate exchanges are not considered, and we assume that each node in the system has access to the K public keys, $v_j^{(i)}, 1 \leq j \leq K$, for each node i. A message from the application layer is appended with K signatures, signed using private keys $s_j^{(i)}, 1 \leq j \leq K$.

Let $t_{s,j,i}$ be the time required to generate (verify) a EC based DS using private (public) key $s_j^{(i)}(v_j^{(i)})$; and the total message signing time, $t_s = \sum_{j=1}^{K} t_{s,j,i}$. Here we have assumed without loss of generality that signing and verifying messages of a particular key-size are identical. For a given byte storage of b bytes available at the security layer, the effective storage capacity in terms of number of packets would be $B(K) = b/(MessageSize + \sum_j SignatureSize(s_j^{(i)}))$, assuming $SignatureSize(s_j^{(i)}) = SignatureSize(s_j^{(l)})$ for $i \neq l$ and ignoring the fact that a to-be-signed message occupies only $MessageSize$ space.

A verifier maintains a K-dimensional vector of weights $L_j^{(i)}$ for the sender i, with $L_1^{(i)} = 1$; these weights are verifier-dependent, though not explicitly indicated. $L_j^{(i)}$ indicates the number of verifications of the j^{th} signature (using public key $v_j^{(i)}$) for each verification of 1^{st} signature for messages from node i. The verifier could ensure these target relative frequencies by either using a deterministic policy, or a probabilistic approach, or possibly some other approach. The

[3]The key intuition from [13] in advocating a holistic approach is that one may not get the best system performance by optimizing each layer independently of each other. The MAC-layer collision probability and security layer blocking probability are both increasing functions of the rate of traffic into these layers [13]. Thus, in a tandem arrangement of security-MAC-security layers as in Figure 1, one may improve the overall system performance by dropping some messages at the security and MAC layers so that the subsequent layers' performance improves significantly. This dependence will also become clear in the mathematical setting of Appendixwhere we actually track the traffic flow between these layers

[4]Or, a slowly changing topology as in the V2V context under road congestion so that the certificate exchange is done at very small frequency. Another possibility is that the certificate exchange could be done over a *control channel* that MAC protocols provide [8], [10].

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2009.7643
http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7643

weights $L_j^{(i)}$ could be dynamic, dictated by the network conditions local to the verifier[5], or possibly some other criteria. This description allows for possibility of not verifying every received message. In Section 3.1 we will consider several strategies of scheduling among the existing flows (messages from various senders) at the security layer.

For example, when $K = 2$, $L_2^{(i)} = 4(8)$ indicates 1(1) heavy-weight verification (say, 256 bit, $v_1^{(i)}$) for every 3(7) light-weight verifications (say, 160 bit, $v_2^{(i)}$) (respectively). For the special case of $K = 2$, let L denote $L_2^{(i)}$.

The *multiple digital signature* scheme proposed has several properties similar to TESLA [4] where the dissemination of the first message, the *message anchor* is accomplished through a PKI based 256 bit digital signature. Subsequent L messages are then authenticated, albeit, in a time-delayed fashion by computing One Way Functions (OWF). Evidently, the verification of the *message anchor* is similar to our heavy weight verification and verification of subsequent messages within the L interval is similar to our light weight verifications. Assuming that all the messages are dropped with the same probability, on an average, the receiver (verifier) will perform L light-weight verifications per heavy-weight verification. Note that, in the case of standard TESLA [4], it is the sender which dictates the value of L, thus ruling out the possibility of receiver-based adaptation.

TESLA and the proposed multiple-signature scheme add significant variance in message verification times at the security layer queue. It is traditionally known [15] that higher variance in service times increases the drop probabilities at a $./G/1/B$ queue. Hence, our work also gives the dependence of drop probabilities on the variance of service times.

Summarizing, appending multiple signatures to a message has the following effects: (a) Increased message size reduces the packet storage capacity at security buffer, and also increases the over-the-air transmission time, (b) Appending multiple signatures per message increases the packet signing time, and (c) Lightweight verification reduces the per-message verification time.

Increase in end-to-end packet drop probability due to (a) and (b), could be compensated by (c). However, lightweight verification may reduce the *resistance to node compromise*. An understanding of the combined effect of (a), (b) and (c), along with security analysis, is thus required. We do this in the subsequent subsections and the Appendix using simulation and analysis. A fixed point approach to obtain the end-to-end drop probability is provided in Appendix.

3.1 Network Layer Performance Analysis

We present simulation study of the performance of multiple digital signature scheme proposed earlier. To keep the presentation simple and to understand the impact of underlying parameters better, we will restrict ourselves to the case of $K = 2$ [6] so that the verification algorithm is determined using only one parameter $L = L_2^{(i)}$. To achieve the level of non-repudiation prescribed by IEEE 1609.2, the largest key size used is 256 bits. To achieve a significant variation in the computation complexity of the second signature, we use a 160 bit key for the second signature. The average signature generation and verification times, as obtained from a 400MHz PowerPC processor coupled with a port of *OpenSSL* [16] cryptographic library, were 32ms for 256 bit keys and 12ms for 160 bit keys [17]. Signing a packet thus requires 44ms, for the two signatures per-message.

At the receiver, a to-be-verified packet entering the security queue is *marked to be light-weight verified* (using 160 bit key) with probability $\frac{L}{L+1}$, thus fixing its verification time.

A scheduling policy at the security layer selects a packet to be processed. For example, in the case of First Come First Serve (FCFS) Scheduling, any data packet at the head of the security queue is selected for packet processing. In the non-preemptive Shortest Remaining Processing Time (SRPT) Scheduling [18], a packet with the least *expected* security processing requirement is given highest priority. For example, when $K = 2$ and service times are deterministic, SRPT prioritizes light-weight verification *first*, heavy-weight verification *second*, and signing a packet *last* as these three operations require, respectively, 12, 32 and 44 ms. Yet another scheduling policy we consider is Weighted Fair Queuing (WFQ) among the two classes of to-be-signed and to-be-verified messages; scheduling within a class is assumed to be FCFS. A packet is chosen from the signing queue with a probability ρ and from the verification queue, with a probability $1 - \rho$.

Appending the additional 160 bit key signature to the packet increases the packet payload by 44 bytes. Assuming a 228 byte data packet [10], this reduces the number of packets that can be stored in the security buffer by 13%, as compared to a single signature scheme ($K = 1$). The PHY data rate considered is 6 Mbps [8]. These results have been used in the C++ based slotted time simulator used in [13] to model multiple signature schemes. The results of the simulation are plotted in Figure 2; N, P_B and P_C denotes the number of nodes in the single cell, the blocking probability at the security layer and the collision probability at MAC layer respectively. These observations follow:

- **Effect of** L: $K = 1$ with 256-bit key is expected to perform better than $K = 2$ with $L = 0$ because $K = 2$ with $L = 0$ indicates unused added per-packet bytes corresponding to the 160 bit signature. This increases P_B and P_C for a given byte-storage capacity at security layer. This effect however starts getting compensated by faster average verification times as L increases, hence P_B for $K = 2$ starts decreasing with L. However, reduction in P_B implies an increase in the security→MAC traffic, eventually leading to MAC being bottleneck, so that after some time the collision probability starts dominating and the MAC layer becomes the performance bottleneck. This phenomenon observed in Figure 2 is also captured in the analysis of the proposed multiple digital signature scheme (Appendix); in the figures we use "N" in legend to emphasize numerical computation results obtained from the fixed point approach of Appendix. The results from

[5]These include, for example, the number of neighboring nodes, the application tolerance for message delays and/or losses, the rate at which packets are being generated at the application layer, the number of messages already verified from a particular sender etc.

[6]In principle K can take any value ($K \geq 2$) and is only upper-bounded by the minimum throughput, communication and storage overheads tolerated by the applicaions at each nodes. It is straightforward to extend the tools developed in this study for larger values of K. Note that the focus of this study is not aimed at providing guidelines nor optimal values for the parameter K.

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2009.7643
http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7643

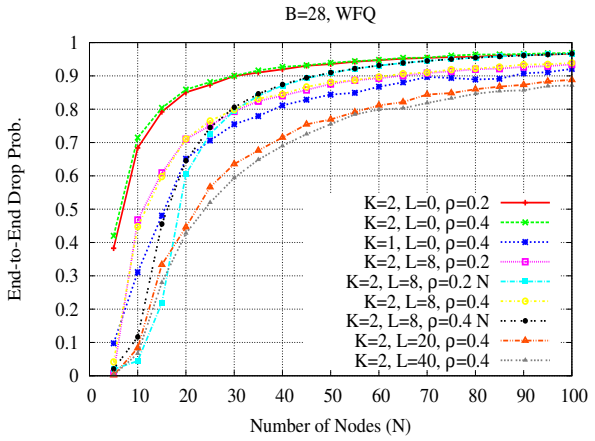

Figure 2: End-to-End Drop Probability v/s Number of Nodes (N).

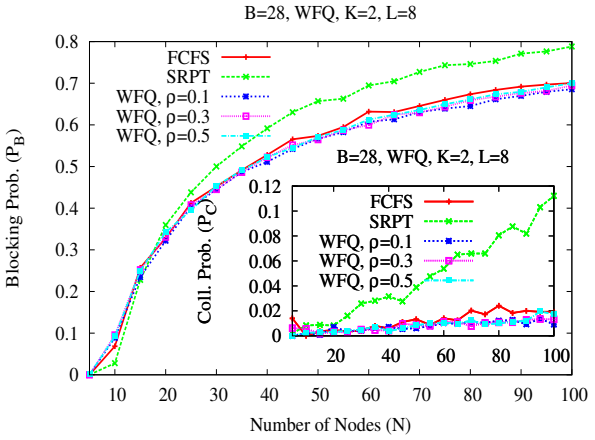

Figure 3: System losses under various scheduling policies illustrated for a buffer size $B = 28$packets.

simulations and fixed point approach are close to each other for large values of N.

- **Effect of Number of Nodes**: Observe from Figure 2 that for a given number of nodes N in the single cell, there is a threshold, say $L^*(K, N)$, such that the $K = 2$ with $L > L^*(2, N)$ performs better (in terms of P_B, P_C and λ_{e2e}) than $K = 1$ scheme, even with higher signature generation time and reduced packet storage capacity because of an increase in the to-be-verified traffic which has smaller average service requirement. The function $L^*(K, N)$ is observed to be non-decreasing in N.

  This clearly implies that there is no single value L^* that is optimal for all values of N. For the case of $N = 5, L = 8$, the proposed scheme is more efficient compared to the single signature scheme ($K = 1$), while for $N \geq 10$, the proposed scheme can be made more efficient by increasing L ($L \geq 20$, as observed in Figure 2). It is hence imperative that schemes that dynamically adapt the value of L to changing environment or topologies be devised to ensure *multiple digital signature schemes* to be efficient over all values of N as compared to single signature schemes.

- **Separating the to-be-signed and to-be-verified flows**: One would be inclined towards avoiding the security-MAC interaction as observed in the previous point. Use of weighted fair queueing to guarantee a minimum service rate to the to-be-signed and to-be-verified traffic appears to be a natural way to bound the maximum load on the MAC layer. The extensive simulation and numerical results including those illustrated in Figure 2 indicate low sensitivity of the system performance to WFQ weights.

- **Flow Level Scheduling at the Security Layer**: SRPT is known to minimize the number of customers in a queue sample path-wise [18]. Implementing SRPT at the security layer is hence expected to reduce the blocking probability P_B. The simulation results of Figure 3 however provide an instance of a completely opposite behavior, wherein FCFS/WFQ scheduling results in smaller P_B for large values of N.

We wish to emphasize that all the explanations provided for the various observed phenomenon have been verified by us using extensive simulations; we are not able to provide the supporting evidence due to page restriction. Further, these intuitions are also strengthened by the detailed flow balance provided in fixed point approach of Appendix.

3.2 A Note on Security Performance

The key sizes employed in the construction of *multiple digital signatures* determines the system security or *resistance to node compromise*. To understand the impact on system security, consider the "*identity spoofing*" attack model, where a malicious node on successfully compromising a signature of a weak key (typically the smallest key size used) of node i, assumes its identity and generates packets with incorrect data at a rate $\frac{1}{t'_p}$. Since, not all the keys of node i are compromised, the attacker substitutes "junk" values as signatures for the rest of the uncompromised signatures and transmits the data packet. The security layer of a receiving node will declare the message as authentic only if it happens to use the compromised key for verification of such packets.

For a receiving node, let P_d represent the rate at which an uncompromised signature is attempted for verification at the security layer or the "attack detect" probability. Assuming that the attacker does not impact the network performance (P_B, P_C etc.), we have the following proposition,

PROPOSITION 1. *For the case of $K = 2$,*

$$P_d = \frac{\frac{1}{t'_p}(1 - P_C)(1 - P_B)}{L + 1}.$$

The time needed to detect an attack is

$$\lambda_d = \frac{1}{P_d} = \frac{L + 1}{\frac{1}{t'_p}(1 - P_C)(1 - P_B)} \quad \textit{slots.}$$

The expressions can be modified to account for multiple attackers.

This direct dependence of λ_d on L needs to be considered in deploying *multiple digital signature* scheme since, where

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2009.7643
http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7643

increased L results in improved system performance, it also increases the time to detect a successful signature compromise, λ_d. Balancing these two conflicting requirements is hence critical.

Depending on the application's tolerance to incorrect information, L could be tuned to ensure a bound on the frequency of accepting a forged message or delay in detecting of a node compromise. For example, when sensor nodes are required to track a slowly changing spatial random field and report only a local-in-time average, some anomalous reading can clearly be tolerated. Further, the first point of detection in any case results in eviction of the compromised node. Similarly, in the V2V context, a vehicle tracking other vehicles trajectory can also have some tolerance to anomalous data.

Remark: In the preceding sections, we have indicated the similarity between TESLA and our proposed scheme in terms of system performance. In terms of security however, our proposed scheme is clearly advantageous since it provides non-repudiation, a security attribute that is not provided by standard TESLA [4].

4. THE STABILITY REGION AND INFORMATION CAPACITY OF ECDSA-BASED SIGNATURE SCHEMES

We need stability of the queues at the security and the MAC layer. By stability we will mean the standard *rate stability*, i.e., the output rate from a layer should be equal to the input rate. Clearly, this amounts to requiring that the blocking probability at the security layer is $P_B = 0$. Further, given the randomness introduced by the MAC layer, it is easy to see that $P_B = 0$ cannot be achieved by a finite buffer. Hence, we will be assuming an infinite buffer capacity available at the security layer and then seek the system parameters that would ensure rate stability. We will see how the stability condition and the fixed point approach is used to determine the end-to-end per-channel-use Information carrying capacity of the system under consideration.

The complex coupling between MAC and security layer implies that one cannot study stability of one layer without looking at the other layer. We will consider a simple slotted Aloha MAC which does not implement carrier sensing, i.e., time is slotted with packet transmission times and each backlogged MAC transmits with probability p in any slot. The approach of this section can be extended to account for carrier sensing using approach of [9], but since the fixed point nature of these approaches dictate numerical computation, one may not get complete feel of the various parameters affecting the stability performance. The expressions obtained in this section explicitly provide a glimpse of the interplay of various parameters from different layers in determining the stability performance of the system.

Let Π be the probability that the MAC layer of a node is backlogged in a single-cell setting with N nodes. Let λ be the rate of application→security traffic and Δ be the MAC layer slot length (packet transmission time). The security layer is assumed to implement an ECDSA-based digital signature scheme with t_s being the average signing/verification time. Assuming that the queue occupancy process of the MAC layers of various nodes are independent[7], we see that the MAC layer of a node essentially behaves like an M/M/1 queue (see Appendix for numerical justification) with arrival rate λ and service requirement $\frac{p}{\Delta}$. Hence, $\Pi = \frac{\lambda\Delta}{p}$, and the probability of collision at MAC layer is $P_C = (1 - (1 - \Pi p)^{N-1})$.

LEMMA 1. *Security and MAC layer are simultaneously stable if the following set of equations are consistent:*

$$\begin{aligned} \lambda &< \frac{p}{\Delta}, \\ \lambda t_s(1 + (N-1)(1-P_C)) &< 1, \\ P_C &= 1 - (1 - \lambda\Delta)^{N-1}. \end{aligned}$$

Outline of Proof The first condition assumes a stable security layer to get a condition for stable MAC. Second and third assume a stable MAC to get a condition for stability of security layer. The third equation assumes that the traffic into the MAC layer of any node is Poisson (of rate λ). • Let $\zeta = \lambda\Delta$ and $\mu = \frac{t_s}{\Delta}$, then we require $\zeta < p$ and $\zeta\mu(1 + (N-1)(1-\zeta)^{N-1}) < 1$. For a given value of p, the stability region

$$\mathcal{Z}_p = \{\zeta \leq p : \zeta\mu(1 + (N-1)(1-\zeta)^{N-1}) < 1\}$$

is not necessarily a convex connected set for all values of μ. The complex interplay between security layer and MAC layer parameters is clearly demonstrated by the fact that the system stability condition depends only on $\mu = \frac{t_s}{\Delta}$, the *relative timescales* of security and MAC layers.

The assumptions of the foregoing model are not binding and one can relax them to get the stability region $\mathcal{Z}_p$. The only reason for making these assumptions is to get tractable analysis.

Appendix B provides basic considerations in determining the information theoretic capacity of the system. The loss process $\{T_i\}$ given in appendix is required to be ergodic. Stable security and MAC layers imply ergodic packet loss process[8].

LEMMA 2. *Let* $\zeta^* = \sup\{\zeta < p : \zeta\mu(1 + (N-1)(1-\zeta)^{N-1}) < 1\}$. *In the stable regime where the conditions of Lemma 1 are satisfied, the per-unit-time information carrying capacity for a ECDSA-based single signature scheme (a given value of* t_s *and* Δ*) is*

$$C = \begin{cases} \frac{p}{\Delta}(1-p)^{N-1} & \text{if } \zeta^* \geq p \\ \frac{\zeta^*}{\Delta}(1-\zeta^*)^{N-1} & \text{otherwise,} \end{cases}.$$

Proof We need $\sup\{\lambda(1-\lambda\Delta)^{N-1} : \lambda\Delta < p \text{ and } \lambda t_s(1 + (N-1)(1-\lambda\Delta)^{N-1}) < 1\}$. Unconditional maximum of the objective function is obtained at $\lambda\Delta = \frac{1}{N}$, however since $\lambda\Delta < p < \frac{1}{N}$ and in this region the objective function is a monotone function of λ, we see that the maximum is achieved at $\lambda = \frac{p}{\Delta}$, i.e., the maximum possible load on the MAC. •

Remark: In practice one would like to adapt p to the number of nodes in vicinity, ensuring $p < \frac{1}{N}$. It turns out that

[7]This assumption is justified here since MAC is not implementing carrier sensing and by the broadcast nature of transmissions.

[8]Since we are considering infinite storage capacity at the security layer in this section, the only source of loss is the wireless collision losses. However, it is clear that the approach is also applicable to finite storage capacity at the security layer as well.

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2009.7643
http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7643

by using $p > \frac{1}{N}$, one can gain in terms of information carrying capacity and also in terms of the message generation rate λ that can be supported. This is because making MAC slightly inefficient could significantly help the security layer. We are exploring this tradeoff in our ongoing work.

4.1 Optimizing the Information Capacity Under a Broadcast Authentication Scheme

When designing the system for the best end-to-end per-unit-time Information transmission while keeping the security/application properties untouched, following the philosophy of [7], we have only three knobs to control: (a) *computation overhead*, (b) *communication overhead*, and (c) *storage overhead*; the other properties need to be kept intact. Thus, while maintaining the other cardinals fixed, we can tradeoff *computation overhead* with *communication overhead*. One-time signatures [7] can drastically cut down on the *computation overhead* but increase the *communication overhead*. Also, for such schemes, there is a one-to-one correspondence between communication and storage overhead. Since most of the load on the receiver comes from verification traffic, we will be concerned only with the verification times in this section.

To gain an understanding of the tradeoffs involved in tuning the parameters of a broadcast authentication scheme with an objective of achieving the information theoretic capacity, we select a scheme which offers tradeoff in the signing time and the signature size. Merkle-Winternitz One-time signatures [19] break a block of m bits into n blocks of k bits each, $nk = m$. The verification cost is, for a general probability distribution over the generated messages, $t_v = cn2^k t_h = c\frac{m}{k}2^k t_h$ for some distribution-dependent constant c and where t_h is the hash computation time. We do not get into the details of MW signatures here because of space constraints. The over-the-air signed message size is then $m + nH + M = m(1 + \frac{1}{k}) + M$ where H is the size of the output of hash function used and M is the MAC layer header. Sending the public key requires only a constant overhead. Clearly, increasing k would increase the verification time, but reduces the message size. One-time signatures, unlike TESLA, provide instantaneous verification.

Assume that the message size has an inverse relation with t_s, the security operation processing times[9], i.e., an increase in digital signature generation/verification processing speed comes at expense of extra per-message bytes (decreasing k). This leads to a decrease in the number of messages that can be stored in the security layer queue $B(b, t_s)$ where b is the byte-storage at the security layer, and also increases the per-message over-the-air time $\Delta(t_s)$. The probability of an erasure is then obtained using the fixed point approach of Appendix. The maximum rate (per-unit-time) at which information can be transmitted over this channel is then obtained using Theorem 1.

We now seek the optimizer by restricting ourselves to Markovian assumptions. Specifically, we assume a Poisson $\mathcal{I}$ and exponential distribution of the signature generation and verification times. We started with a given value of B and δ, and used two functional forms for the dependence $B(t_s)$ and $\Delta(t_s)$: Figure 4$\rightarrow B(t_s) = B * t_s$ and $\Delta(t_s) = \frac{\delta}{t_s}$, Figure 5$\rightarrow B(t_s) = B * \log(t_s + 1)$ and $\Delta(t_s) = \frac{\delta}{\log(t_s+1)}$. These functional forms are motivated by the MW-OTS scheme.

[9]One need not assume same signature generation and verification times, but is done here for simplicity and without loss of generality as significant fraction of load on security layer is from the to-be-verified stream.

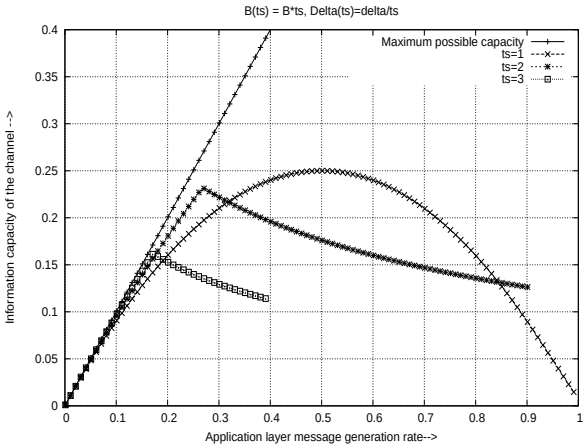

Figure 4: Per unit time transmission rates.

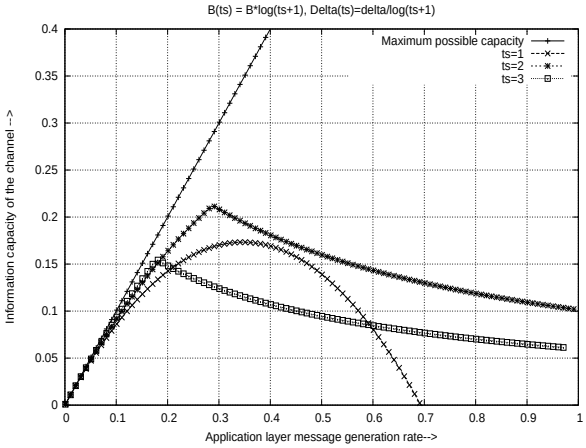

Figure 5: Per unit time transmission rates.

Key observations from the figures are

For a given $\lim_{t\rightarrow\infty} \frac{\mathcal{I}(t)}{t}$, there is no monotonicity in end-to-end drop probability as a function of t_s if one also takes into account the effect of changing t_s on effective buffering capacity at the security layer and the over-the-air transmission times. This implies that one cannot arbitrarily reduce the digital signature processing times to achieve good end-to-end performance.

The information capacity of the system depends crucially on the functional form of the various dependence of the underlying broadcast authentication scheme.

Minimizing end-to-end drop probability is *not* necessarily the way to achieve best *per-unit-time* information capacity from the secure V2V systems. It only helps implies best per-sample information transmission.

Capacity region is convex, i.e., there exists a threshold C^* achieved at the input rate of $\hat{\lambda}$ so that as long as the per-unit-time information rate of the sampled process is $H(\lambda) < C^*$, there exists a message generation rate and a coding scheme

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2009.7643
http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7643

that can ensure reliable transmission of this information. Interesting property however is that this message generation is not unique, i.e., one can achieve same end-to-end per-unit-time information transmission either with a smaller channel input rate (implying smaller drop probability) or with a high input rate.

[7] proposes a scheme for signing low entropy messages. However, as is evident from the foregoing discussion, for the sensing/tracking application, entropy depends on the sampling rate/transmission rate used. The quantification "low" should hence be used with caution. This also provides us with another possibility of tradeoff between sampling rate and coding efficiency.

We have not considered the information content of the timing process as in [20]; a subject of current study.

5. CONCLUSION AND FUTURE WORK

In this paper, a fixed point approach has been proposed to analyse the performance of a *generic* broadcast authentication scheme which allows for the possibility of *multiple digital signatures* being attached to messages. This kind of protocol enables us the flexibility of trading-off communication or computation overhead with the security strength so desired (based on the properties of the various signatures attached). We observe that a holistic approach to designing such systems is required. For instance, just attempting to optimize the performance of the security layer (by either using a fast broadcast authentication scheme, or using an optimal single-queue scheduling policy like SRPT) does not provide the expected gain; this is because of the dependence between the MAC layer and the security layer.

Further, our analysis on the channel capacity and stability of these systems reinforces the need for a *holistic* approach in studying the impact of a *broadcast authentication scheme*. For example, changing the signing time changes the required storage and also over-the-air size, so that the end-to-end drop probability does not necessarily reduce by speeding-up the signing time. At the same time, the information capacity of the system, being directly related to the end-to-end drop probability depends crucially on the functional form of the various dependence of the underlying *broadcast authentication scheme*.

The work presented in this paper is based on the notion of a *single cell*. The reason for this restriction is that there are essentially two source of randomness in periodic secure V2V application: a) the randomness introduced by the MAC layer, and b) that introduced by the mobility of vehicles. Having gained significant understanding of the system behavior with only one source of randomness (MAC), extending the model to include mobility of nodes is the subject of current investigations.

6. REFERENCES

[1] IEEE, "IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Nov. 1997. P802.11."

[2] C. Incorporated, "Crossbow technologies."

[3] M. Raya, "The security of vehicular ad hoc networks," *Proc. of the 3rd ACM workshop on Security of Ad hoc and Sensor networks*, 2005.

[4] A. Perrig, R. Canetti, J. Tygar, and D. Song, "The TESLA Broadcast Authentication Protocol," *RSA CryptoBytes*, vol. 5, no. 2, pp. 2–13, 2002.

[5] Varghese A., Iyer A., Kherani A., Shorey R., "Performance of TESLA for Secure V2V Safety Applications," *Under Submission*, 2008.

[6] Kherani A., Iyer A., Varghese A., Shorey R., "Performance of Broadcast Authentication for Secure V2V Safety Applications: A Holistic View," *WISARD Workshop*, 2008.

[7] M. Luk, A. Perrig, and B. Whillock, "Seven Cardinal Properties of Sensor Network Broadcast Authentication," *Proc. of the 4th ACM workshop on Security of ad hoc and sensor networks*, 2006.

[8] Crash Avoidance Metrics Partnership (CAMP), May 2005, USA, "Vehicles Safety Communications Consortium, *Vehicle Safety Communications Project - Final Report*."

[9] Rao A., Kherani A. A., Mahanti A., "Performance evaluation of 802.11 broadcasts for a single cell network with unsaturated nodes," *IFIP Networking*, pp. 836–847, 2008.

[10] IEEE Vehicular Technology Society, Jul 2006, "ITS Committee, *IEEE Trial-Use Standard for Wireless Access in Vehicular Environments- Security Services for Applications and Management Messages*."

[11] A. Menezes, M. Qu, D. Stinson, and Y. Wang, "Evaluation of Security Level of Cryptography: ECDSA Signature Scheme," *Evaluation report for Japanese government IPA CRYPTREC project*, 2001.

[12] D. Hankerson, S. Vanstone, and A. Menezes, *Guide to Elliptic Curve Cryptography*. Springer, 2004.

[13] A. Iyer, A. Kherani, A. Rao, and A. Karnik, "Secure V2V communications: Performance impact of computational overheads," *Computer Communications Workshops, INFOCOM. IEEE Conference on*, 2008.

[14] Karnik A., "Patent Details suppressed."

[15] R. Wolff, *Stochastic modeling and the theory of queues*. Prentice Hall.

[16] www.openssl.org, "OpenSSL."

[17] Liu, A. and Ning, P., "TinyECC: A Configurable Library for Elliptic Curve Cryptography in Wireless Sensor Networks," *Proc. of the 7th Intl. Conf. on Information Processing in Sensor Networks*, 2008.

[18] L. Schrage, "A proof of the optimality of the shortest remaining processing time discipline," *Operations Research*, vol. 16, no. 3, pp. 687–690, 1968.

[19] R. Merkle, "A Certified Digital Signature," *Proceedings of the 9th Annual International Cryptology Conference on Advances in Cryptology*, pp. 218–238, 1989.

[20] B. Prabhakar and R. Gallager, "Entropy and the timing capacity of discrete queues," *Information Theory, IEEE Transactions on*, vol. 49, no. 2, pp. 357–370, 2003.

[21] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *Selected Areas in Communications, IEEE Journal on*, vol. 18, no. 3, pp. 535–547, 2000.

[22] B. Bensaou, Y. Wang, and C. Ko, "Fair medium access in 802.11 based wireless ad-hoc networks," *Mobile and Ad Hoc Networking and Computing, MobiHOC. First Annual Workshop*, pp. 99–106, 2000.

[23] S. Diggavi and M. Grossglauser, "On information

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2009.7643
http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7643

transmission over a finite buffer channel," *Information Theory, IEEE Transactions on*, vol. 52, no. 3, pp. 1226–1237, 2006.

[24] C. Nair, B. Prabhakar, and D. Shah, "On entropy for mixtures of discrete and continuous variables," *Arxiv preprint cs.IT/0607075*, 2006.

[25] T. Cover and J. Thomas, *Elements of Information Theory*. Wiley-Interscience New York, 2006.

APPENDIX

A. ISOLATING THE STREAMS: WFQ AND MULTIPLE DIGITAL SIGNATURE SCHEME

The security layer implements weighted fair queuing to schedule the to-be-signed and to-be-verified packets. This approach breaks the system into many independent blocks so that each of them can be analyzed separately. A consistency check equalizing the flow across various layers then yields a fixed point that is taken to be the operating point of the system. This is along the lines of standard modeling literature on IEEE 802.11 [21].

Broadcast random access nature of MAC implies no exponential backoff or retransmission of collided packets, the extent of *capture* [22] is clearly negligible because the probability of accessing the channel does not change over successive transmission attempts. Large N and randomization introduced by the MAC thus indicates that one can model the MAC→security layer traffic (packet arrival process) as a Poisson process. This intuition is confirmed for the ECDSA-only scheme as given in Figure 6 which plots the distribution function of the interarrival times for MAC→security and security→MAC traffic; these times are observed to be exponentially distributed. This intuition also holds true for

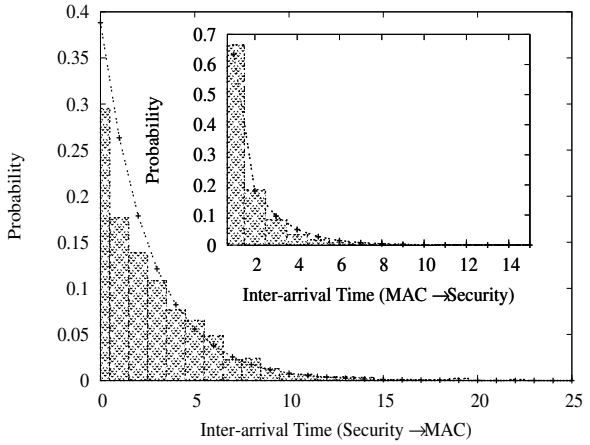

Figure 6: Distribution of inter-arrival times in the security→MAC and MAC→security traffic.

the multiple signature scheme (note that $K = 1$ corresponds to the ECDSA-only scheme), especially in the case of heavily loaded system. Note that this intuition is also valid for a general application→security traffic generation process as it is based on the MAC layer phenomenon.

For ease of presentation, and without loss of generality, we assume that the application→security traffic, being a negligible part of the total traffic into the security layer, is Poisson with mean rate $\lambda = \frac{1}{t_p}$. The MAC→security process is assumed to be Poisson of rate λ_{MS}, so that the aggregate traffic rate into the security layer is $\Lambda = \lambda + \lambda_{MS}$. In the symmetric system under consideration it is assumed that these quantities are same across all the nodes. For ease of presentation again, we assume $K = 2$ in the multiple digital signature scheme and assume deterministic service times of t_s, t_l and t_h to denote the time for signing, light-weight signature verification and heavy-weight signature verification, respectively; one an also use a different distribution for the service times, to account for the load on the processor from other non-V2V applications running in parallel. At the security layer of a node, after a service completion, if to-be-signed and to-be-verified packets are both present, a *to be signed* packet is selected with probability ρ and *to-be-verified* with probability $1-\rho$. Further, when a *to-be-verified* packet is selected, the security server performs a light-weight verification with probability $\frac{L}{L+1}$ and heavy-weight verification with probability $\frac{1}{L+1}$.

Let the *state* of a node's security queue at any time be the two-dimensional vector of the number of to-be-signed and to-be-verified messages present in the queue. Let X_n denote the state at the n^{th} service completion (a packet departure) instant. Under the Poisson assumption, X_n is a Markov chain. Let t_n be the time between n^{th} and $n-1^{th}$ service completions at the security layer of the node under consideration. Then the process $\{(X_n, t_n)\}$ forms a Markov Renewal Process (MRP) with natural definition of a renewal *cycle*. The blocking probability at the security layer is, using PASTA and straightforward application of renewal reward theorem [15],

$$P_B(\lambda, \lambda_{MS}) = \frac{1}{\lambda + \lambda_{MS}} \frac{\sum_x \pi_x E_x D}{\sum_x \pi_x E_x T},$$

where π_x denotes the stationary probability that a renewal cycle starts in state x and $E_x D$ ($E_x T$) denote the expected number of packets dropped (cycle length) in the renewal cycles starting in state x.

Let $P1(n_1, n_2)$ denote the probability that a *to-be-signed* packet is selected for security operation at the start of a renewal cycle when the state is (n_1, n_2); let $P2(n_1, n_2) = 1 - P1(n_1, n_2)$. Closed form expressions for these probabilities are straightforward. Then $E_{(n_1,n_2)}T = P1(n_1, n_2)t_s + P2(n_1, n_2)\left[Lt_l + t_h\right]/(L+1)$ and $E_{(n_1,n_2)}D = P1(n_1, n_2)$ $\sum_{a=B-(n_1+n_2)}^{\infty} \frac{r(\Lambda t_s)^a e^{-\Lambda t_s}}{a!} + P2(n_1, n_2) \sum_{a=B-(n_1+n_2)}^{\infty}$ $\frac{r\left[L(\Lambda t_l)^a e^{-\Lambda t_l} + (\Lambda t_h)^a e^{-\Lambda t_h}\right]}{a!(L+1)}$ where $r = a - B + n_1 + n_2$. The transition probability for the Markov chain $\{X_n\}$ is similarly obtained in closed form, and a numerical routine can be used to get π_x, the stationary distribution of the Markov chain $\{X_n\}$. Numerical computation of this system *for a given value of* λ_{MS} gives us the blocking probability $P_B(\lambda, \lambda_{MS})$. MAC→security traffic, λ_{MS}, is an aggregate of those messages coming out of the security layers of the remaining $N-1$ nodes that were received successfully by the MAC layer of the node under consideration. Assuming a stable system, the security→MAC traffic rate for any node is $\lambda_{SM} = \lambda(1 - P_B(\lambda, \lambda_{MS}))$. Assuming that $P_C(\lambda_{SM})$ is the collision probability at the MAC layer, $\lambda_{MS} = (N-1)\lambda(1 - P_B(\lambda, \lambda_{MS})\left[1 - P_C(\lambda_{SM})\right]$. Here $P_C(\cdot)$ is the collision prob-

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2009.7643
http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7643

ability at the MAC layer as a function of the message arrival rate into the MAC; various fixed point approaches are available to get $P_C(\cdot)$, for example [9]. We thus use a fixed point approach where we perform a consistency check of the traffic flow into and out of successive layers, i.e., $\lambda_{MS} = (N-1)\lambda_{SM}(1-P_C(\lambda_{SM}))$, $\lambda_{SM} = \lambda(1-P_B(\lambda,\lambda_{MS}))$. To do this, we start with a given λ and P_B, thus getting $P_C(\cdot)$, λ_{MS} and λ_{SM}, which then provides $P_B(\cdot,\cdot)$ itself, and the process continues till it converges. Note that we are assuming here that the system is stable, a subject of study in Section 4.

B. END-TO-END INFORMATION CAPACITY

A continuous time process $\{X(t)\}$ is sampled using an independent, stationary and ergodic, point process $\mathcal{S}$. In Sensor networks applications $\{X(t)\}$ could be an environmental variable like radioactive radiation, while in the V2V context it could be the actual location or velocity of the transmitting vehicle. The average per-sample information of the sampled process $\{X_i\}$ is

$$H_{\{X,\mathcal{S}\}} = \lim_{n\to\infty} \frac{H(X_1,\ldots,X_n)}{n} = \lim_{t\to\infty} \frac{H(X_1,\ldots,X_{\mathcal{S}(t)})}{\mathcal{S}(t)}.$$

Let the channel input sequence $\{C_i\}$, i.e., the actual messages generated by the application layer of a node, be sent out of the application layer according to a *marked* point process $(\mathcal{I},\mathcal{M})$, where $\mathcal{I}(t)$ is the number of inputs (application $\rightarrow$ security) to the channel by time t, and $\mathcal{M}_i$ is the mark associated with the i^{th} channel input. Let the sequence received at the receiver be $\{Y_i\}$. The i^{th} channel input observes a channel state Q_i, that takes values in a finite set. The i^{th} input to the channel is *erased* with probability $p(Q_i)$ so that $p(Y_i = C_i|Q_i) = 1-p(Q_i)$ and $p(Y_i = e|Q_i) = p(Q_i)$ where e represents the erasure symbol [23]. For a given input point process $\mathcal{I}$, let $T_i = 0$ if $Y_i = e$, and $T_i = 1$ otherwise.

The operational interpretation of this setup is that C_i are the actual to-be-signed message contents, and the corresponding marks $\mathcal{M}_i$ indicate their signature generation times, storage requirement, verification times etc. The sequence $\{\mathcal{M}_i\}$ could have complex dependence to account for schemes having correlation in signature/verification times of successive channel inputs, for example, in TESLA [4] and our multiple digital signature scheme. We will be assuming that the process $\{C_i\}$ and the marked point process $(\mathcal{I},\mathcal{M})$ are independent of each other. The process $\mathcal{I}$ will be allowed to depend on the sampling point process $\mathcal{S}$. The random variable $\mathbf{T}$ depends on $\mathcal{I},\mathcal{M}$.

THEOREM 1 ([23]). *The* per-channel-use *Information Capacity of the channel (security-MAC-security of a pair of nodes) under consideration is* $\bar{C}\lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n}T_i = \bar{C}p(\mathbf{T}=1)$*, i.e., the probability of an input not being erased; here* $\mathbf{T}$ *is a random variable with stationary distribution of the sequence* $\{T_i\}$ *and* $\bar{C}$ *is the number of bits per generated message* C_i*. Further, the* per-unit-time *information capacity of the channel is* $\bar{C}p(\mathbf{T}=1)\lim_{t\to\infty}\frac{\mathcal{I}(t)}{t}$.

In passing we remark that the sequence $\{C_i\}$ generated using appropriate codebook is assumed to have uniform distribution.

B.1 Information Content of a Continuous Time Process

If the per-unit-time information content (entropy rate, [24]) of the process $\{X(t)\}$ is less than the maximum per-unit-time information transmission capacity of the channel, the receiver can reliably reconstruct the actual process $\{X(t)\}$ when appropriate coding scheme is used (the contents $\{C_i\}$).

The entropy rate of the finite state process $\{X(t)\}$ is determined as follows: In a given interval of length T, the possible sources of randomness are: (a) $N(T)$, the number of transitions in the process $\{X(t)\}$ till time T, (b) $L_i, 1\le i\le N(T)$, the instants of the transitions, (c) $X(0)$ the process value at time 0, and (d) $X(L_i), 1\le i\le N(T)$, the process value just after the transitions. Then, the entropy rate of the process $\{X(t)\}$ is defined as
$H_X = \lim_{T\to\infty}\frac{1}{T}H(N(T), L^{N(T)}, X(0), X(L_1),\ldots,X(L_{N(T)}))$.

THEOREM 2. *For a finite state CTMC with generator matrix* Q*, the per-unit-time entropy rate is*

$$\frac{1}{\sum_i \frac{\pi(i)}{-Q_{ii}}}\left[H_{MC} + \sum_i \pi(i)(1-\log(-Q_{ii}))\right],$$

where $\pi(i)$ *is the stationary distribution of the embedded Markov chain (with transition probability* $E_{ij} = \frac{Q_{ij}}{-Q_{ii}}$*,* $j\neq i$*), and* H_{MC} *is the entropy rate of the embedded Markov chain.*

The proof of this theorem is similar to that of [24, Prop. 4.1], with the difference that they consider a uniformized Markov process, whereas we work directly with the original process. Theorem 2 indicates that the uniformization process is *not* "information preserving" because the definition of information rate (from [24]) also takes into account the number of events. Uniformization increases the number of *events*, and also ensures smaller variance in the states at these events, reducing the overall individual per-sample entropy.

In the applications where sensors are used to check threshold crossing of an environmental variable, $X(t)\in\{0,1\}$. Similarly in the example of V2V communication one can assume $X(t)\in\{0,1\}$ indicating $\{near, far\}$. Note that in the V2V context the information content may come from the coordinates of both, the sender and the receiver since the quantification near or far requires both the coordinates.

COROLLARY 1. *For the two-state Markov process* $\{X(t)\}$ *with generator matrix* $Q = [-a_0, a_0; a_1, -a_1]$*,*
$H_X = \left[\frac{a_0}{a_0+a_1}a_0(1-\log a_0) + \frac{a_1}{a_0+a_1}a_1(1-\log a_1)\right]$.

COROLLARY 2. *When* $a_0 = a_1 = a$*,* $H_X = a(1-\log a)$.

There is no information in the state process because of its deterministic alternating nature in a two-state process.

Remark Another definition of the per-unit-time entropy rate of the original process could be
$\lim_{t\to\infty}\frac{1}{t}\lim_{N\to\infty}\frac{1}{N}H(X(\frac{t}{N}), X(\frac{2t}{N}),\ldots,X(\frac{Nt}{N}))$. We can identify this to be uniformization with increasing rate. With the intuition from Theorem 2, this definition of information rate does not represent the information rate of the original process.

Remark An n-bit quantization of a continuous random variable (the inter-event times in our setting) with differential entropy H, has a entropy of $H+n$ bits [25, Theorem 9.3.1]. This is to be considered when comparing the entropy rate of $\{X(t)\}$ and the information capacity of the system.

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2009.7643
http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7643