Optimal Node Placement in Distributed Wireless Security Architectures

Fabio Martignon, Stefano Paris, Antonio Capone

Optimal Node Placement in Distributed Wireless Security Architectures

Fabio Martignon[1] and Stefano Paris[2] and Antonio Capone[2]

[1] Department of Information Technology and Mathematical Methods
University of Bergamo
fabio.martignon@unibg.it
[2] Department of Electronics and Information
Politecnico di Milano
{paris,capone}@elet.polimi.it

Abstract. Wireless mesh networks (WMNs) are currently accepted as a new communication paradigm for next-generation wireless networking. They consist of mesh routers and clients, where mesh routers are almost static and form the backbone of WMNs.
Several architectures have been proposed to distribute the authentication and authorization functions in the WMN backbone. In such distributed architectures, new mesh routers authenticate to a key management service (consisting of several servers, named *core nodes*), which can be implemented using threshold cryptography, and obtain a temporary key that is used both to prove their credentials to neighbor nodes and to encrypt all the traffic transmitted on wireless backbone links.
This paper analyzes the optimal placement of the core nodes that collaboratively implement the key management service in a distributed wireless security architecture. The core node placement is formulated as an optimization problem, which models closely the behavior of real wireless channels; the performance improvement achieved solving our model is then evaluated in terms of key distribution/authentication delay in several realistic network scenarios.
Numerical results show that our proposed model increases the responsiveness of distributed security architectures with a short computing time, thus representing a very effective tool to plan efficient and secure wireless networks.
Index Terms: - Wireless Mesh Networks, Security, Distributed Architecture, Optimal Node Placement.

1 Introduction

Wireless Mesh Networks (WMNs) have been accepted as a new communication paradigm able to provide a cost-effective means to deploy all-wireless network infrastructures [1]. Network nodes of WMNs, named mesh routers, provide access to mobile users like access points in wireless local area networks, and they relay information hop by hop like routers, using the wireless medium.

As the mesh networking technology has become popular, the research community has proposed innovative security solutions to meet the requirements of

WMNs. However, only few works consider the communication and computational overhead caused by the proposed protocols in their design.

In WMNs, two different security areas can be identified: one related to the *access* of user terminals which can be provided using standard techniques [2], and the other related to network devices in the *backbone* of the WMN.

Backbone security is a crucial issue. Mesh networks typically employ low-cost devices that cannot be protected against removal, tampering or replication. As a consequence, the research community has proposed several architectures to authenticate and authorize the mesh devices.

Centralized security solutions, like those proposed in [3, 4], can exhibit lower costs than distributed approaches; however, they are characterized by a single point of failure (the key management server), which can be exploited by an adversary to attack and subvert the whole network.

To overcome this problem, some preliminary distributed solutions have been proposed in the wireless ad hoc and mesh network research fields [5–7]. The essence of all distributed security architectures lies in the necessity for all mesh routers to transmit periodically to a subset of nodes (which we refer to as *core nodes*) the authentication request to be authorized to join the network. The authorization usually consists of some cryptographic information necessary to create the temporary key that all mesh routers use to encrypt the traffic transmitted over the wireless backbone.

The deployment of the core nodes is therefore a key element for the performance of all distributed wireless security architectures. In fact, each generic mesh router should be sufficiently close to a subset of core nodes, in order to collect in the shortest possible time their partial authorization responses necessary to obtain the entire cryptographic information.

In this paper, we provide a general framework to increase the responsiveness of all distributed wireless security architectures, by focusing on the optimal placement of core nodes.

In an effort to understand how this issue impacts the performance of WMNs, our work makes the following unique contributions. (1) First, we propose a novel and efficient Integer Linear Programming model that optimizes the core node positions, reducing the overall authentication delay in a WMN where a distributed security architecture is implemented. (2) Second, we use an effective link metric that, differently from existing works, models the features of commercial wireless cards. The metric consider both the available channel rate and the total delay that packet transmissions experience over wireless links. (3) Third, we evaluate the proposed model in several realistic network scenarios, comparing the responsiveness of the two architectures proposed in [4] and [7]. Numerical results demonstrate that our proposed model permits to improve consistently the WMN responsiveness. Furthermore, our model can be solved to the optimum in a short computing time, even for large-scale network topologies, thus representing an important tool to design secure and efficient distributed wireless mesh networks.

The paper is structured as follows: Section 2 discusses related work. Section 3 briefly presents two typical security architectures, one fully distributed, the other

centralized, which will be used in the Numerical Results section to gauge the performance of our proposed node placement model. Section 4 describes the Integer Linear Programming model that finds the optimal placement of core nodes. Section 5 discusses numerical results that show the effectiveness of our approach in a set of realistic network scenarios. Finally, conclusions and future research issues are presented in Section 6.

2 Related Work

Several works investigate the use of cryptographic techniques to achieve high fault tolerance against network partitioning. In [5] and [8], two different approaches are presented to allow specific coalitions of devices to act together as a single certification authority, whereas in [9] a hierarchical key management architecture is proposed to obtain an efficient establishment of distributed trust. Capkun et al. [10] propose a fully self-organized public-key management scheme that does not rely on any trusted authority to perform the authentication of other peer nodes. The public key management schemes proposed in [6] and [11] further enhance the security of distributed approaches by using proactive secret sharing and fast verifiable share redistribution techniques which permit to update periodically the secret shares.

Even though these distributed systems improve the network fault tolerance by removing the single point of failure introduced by centralized schemes, they are not very efficient in terms of computational or communication overhead. On the other hand, the centralized architecture proposed in [4] (MobiSEC), provides both access control for mesh users and routers with a negligible impact on the network performance. However, this latter solution is characterized by a single point of failure (the key management server), which can be exploited by an adversary to attack and subvert the whole network.

In summary, we underline that none of the above solutions considers the performance optimization issue in its design, while in this paper we provide a general framework to increase the responsiveness of all distributed security architectures.

The problem of finding the optimal places of network nodes that perform specific tasks in WMNs has been extensively studied in literature. In particular, this problem can be considered as a variation of the Capacitated Facility Location Problem (CFLP), which has been studied in the field of Operations Research [12]. Several works have extended the CFLP in order to model the constraints of the wireless environment and maximize the network throughput [13], or optimize the gateway placement while enabling an efficient reuse of the available resources [14]. Since CFLP is NP-hard, several heuristic algorithms have been proposed to efficiently solve such problems [15, 16].

To the best of our knowledge, this paper is the first that addresses the core node placement problem in WMNs protected by distributed security architectures. This is performed using an optimization tool, while simulation results confirm the performance improvement derived from applying our proposed node placement strategy.

3 Centralized versus Distributed Security Architectures

This section briefly introduces two security architectures, one fully centralized (named MobiSEC [4]), the other completely distributed (named DSA-Mesh [7]). Their performance will be compared in the Numerical Results section under the proposed node placement model. The goal is simply to illustrate the main differences between centralized and distributed security approaches in WMNs. We underline that our proposed model is general and can be applied to optimize the node placement of any distributed wireless security architecture.

MobiSEC and DSA-Mesh adopt a similar approach to protect the backbone of a WMN, that is, all mesh routers obtain the same temporary secret which is used both to prove their credentials to neighbor nodes and to encrypt all the traffic transmitted on the wireless backbone links.

In MobiSEC, backbone security is provided as follows: each new router that needs to connect to the mesh network first authenticates to the nearest mesh router exactly like a client node, gaining access to the WMN. Then it performs a second authentication, connecting to a Key Server able to provide the credentials to join the mesh backbone. Finally, the Key Server distributes the information needed to create the temporary key that all mesh routers use to encrypt the traffic transmitted over the wireless backbone.

On the other hand, DSA-Mesh is a completely distributed architecture, since it distributes the Key Server functionalities among a group of core nodes using threshold cryptography [7]. In the DSA-Mesh architecture, the Key Server consists of n special mesh routers (the *core* routers), which collaboratively generate the new session secret and provide it to the other backbone nodes (the *generic* mesh routers). The employment of threshold cryptography permits to reduce the overhead of the authentication and key management protocols, since it enables t out of n core mesh nodes to perform this operation jointly, whereas it is infeasible for at most $t-1$ nodes to do so, even by collusion. Throughout the paper, we will use the notation (n, t) to indicate such a threshold cryptographic system. Since we assume that a WMN contains t tamper-resistant nodes, n can be at most equal to $2t-1$ in order to make it impossible to compromise the t nodes necessary to recover the session secret. For this reason, and for maximizing at the same time the network reliability to node failures, the most reasonable setting, adopted in the following, is $n = 2t-1$.

A generic mesh router, after entering in the radio range of a mesh router already connected to the wireless backbone, broadcasts its first request to the entire network to obtain the secret used in the current session by the other routers that form the backbone, and the time when it was generated.

Each core node that receives the request from a generic node, after verifying the authenticity of the request, sends back the session secret and the timestamp encrypted with the public key of the generic node, and signs the message with its partial secret of the *key service* private key.

The generic node, after receiving at least t different responses, combines them and verifies the digital signature of the message using the *key service* public key. If the digital signature of the message is correct, then the generic node decrypts

the message and obtains the secret used by all mesh routers to create the key sequence of the current session. Finally, the generic node, based on the instant at which it joins the backbone, computes the key currently used to protect the wireless backbone and its remaining validity time.

Therefore, in the DSA-Mesh architecture the core nodes distribute the information needed to create the temporary key used by all mesh routers.

Figure 1 shows an example WMN with the message exchanges performed between generic and core nodes. A (5,3) threshold scheme is adopted, that is, there are $t = 3$ tamper resistant nodes out of $n = 5$ core nodes; black and white circles represent core and generic nodes, respectively.

For the sake of clarity, we draw only the messages necessary to compute the signature, which are represented by solid arrows for requests (from generic to core nodes), and by dashed arrows for core node responses.

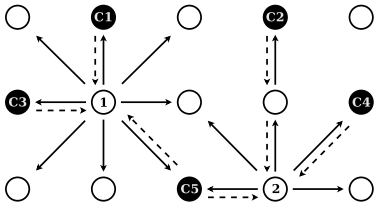


Fig. 1. Distributed security architecture: a (5,3) threshold scheme is adopted in this example WMN, where black and white circles represent, respectively, core and generic nodes. Solid lines represent requests, while dashed lines represent core node responses.

4 Optimal Core Node Placement

This section formulates the core node placement in distributed wireless security architectures as an optimization problem. In particular, an Integer Linear Programming (ILP) model is provided to solve the placement problem.

The input consists of a set of network node locations, N, the number t of messages needed to reconstruct the session secret (which is equal to the number of tamper-resistant nodes in the WMN), the total number $n = 2t - 1$ of core nodes, and finally a *distance* (or *cost*) d_{ij} between each pair of network nodes $i, j \in N$.

The problem consists in placing the n core nodes in order to minimize the maximum distance between each generic node and the farthest of the t closest core nodes. The rationale behind minimizing the *maximum* distance between generic and core nodes lies in the fact that only when t core node messages are collected, each generic node can discover the key used to encrypt data on the WMN backbone.

The setting of the distance function is a key element for an effective modeling of the behavior of a real WMN. For this reason, we propose to define it as the expected transmission delay, which depends both on the data rate (transmission time) and channel length (propagation time). More precisely, we compute the

path loss of each wireless link, and then we evaluate the achievable transmission rate according to that provided by the data sheet of the Wistron CM9 commercial wireless cards (based on Atheros chipset). Note that almost all wireless cards based on the same Atheros chipset are characterized by the same sensitivity.

In the next section we will compare our proposed metric with a simple scenario in which the wireless link costs are all set to 1, regardless of their data rate and length.

The received power (P_r) and the corresponding path loss (PL_{dB}) of a wireless link having length d can be computed according to the following equations:

$$\begin{aligned} P_r &= P_t \cdot \left(\frac{\lambda}{4\pi}\right)^2 \cdot \left(\frac{1}{d}\right)^\gamma = P_t \cdot \left(\frac{c}{4\pi f}\right)^2 \cdot \left(\frac{1}{d}\right)^\gamma \\ PL_{dB} &= 10 \cdot log_{10}\left(\frac{P_t}{P_r}\right) = 10\gamma \cdot log_{10}(d) + 20 \cdot log_{10}(f) - 32.45 \end{aligned} \tag{1}$$

where the transmission power P_t is equal to 0.1 W, while the frequency f is set to 5.18 MHz; γ is the path loss exponent.

The achievable transmission rate of the wireless link e, r_e, is evaluated comparing its path loss with those listed in Table 1.

Table 1. Achievable transmission rate as a function of the CM9 wireless card sensitivity

PL (dB)	88	87	85	83	80	75	73	71
Rate (Mbit/s)	6	9	12	18	24	36	48	54

The transmission time of the link, t_e, is then approximated simply dividing the message length, L, by the transmission rate: $t_e = \frac{L}{r_e}$. Note that in real WMN implementations, the rate of each link can be easily provided by the underlying MAC protocol.

The overall transmission time of the path P_{ij} connecting any two mesh routers i and j can therefore be evaluated as the sum of the transmission times of all the links that belong to P_{ij}:

$$d_{ij} = \sum_{e \in P_{ij}} t_e = L \cdot \sum_{e \in P_{ij}} 1/r_e. \tag{2}$$

We observe that, since in distributed security architectures the authentication messages are actually transmitted at the highest priority level, the queueing delays they experience are almost negligible. Hence, expression (2) represents a good approximation of the overall delay experienced by these messages.

Having defined link and path costs, we now introduce the decision variables used in our ILP model: y_i indicates which network nodes are selected as core nodes, whereas x_{ij} provides the assignment of generic nodes to core nodes. More precisely:

$$y_i = \begin{cases} 1 & \text{if generic mesh router } i \text{ is selected as core node} \\ 0 & \text{otherwise} \end{cases}$$

$$x_{ij} = \begin{cases} 1 & \text{if generic mesh router } j \text{ is assigned to core node } i \\ 0 & \text{otherwise} \end{cases}$$

Given the above definitions, the optimal core node placement problem can be stated as the follows:

$$\min(u) \tag{3}$$

$$s.t. \sum_{i \in N} x_{ij} = t \qquad \forall j \in N \tag{4}$$

$$\sum_{i \in N} y_i = n \tag{5}$$

$$x_{ij} \leq y_i \qquad \forall i, j \in N \tag{6}$$

$$u \geq x_{ij} d_{ij} \qquad \forall i, j \in N \tag{7}$$

$$x_{ij} \in \{0, 1\} \qquad \forall i, j \in N \tag{8}$$

$$y_i \in \{0, 1\} \qquad \forall i \in N \tag{9}$$

The objective function (3) minimizes the maximum distance (referred to as u) between each generic node and the farthest of the t closest core nodes. Constraints (4) ensure that each node $j \in N$ is assigned exactly to t core nodes, while constraints (5) ensure that exactly $n = 2t - 1$ mesh routers are selected as core nodes. Constraints (6) restrict generic node assignments only to selected core nodes (i.e., they ensure that whenever a node j is assigned to a core node i, this latter must necessarily have been selected as core node). Constraints (7) define the lower bound on the maximum distance between any node i and core node j, which is the quantity being minimized as objective function. Finally, constraints (8) and (9) ensure the integrality of the binary decision variables.

We observe that our problem is NP-hard, since it generalizes the t-neighbor n-center problem [17]. However, we were able to solve it to the optimum for realistic, large WMN scenarios using the CPLEX commercial solver; moreover, several polynomial time approximation algorithms that achieve a constant approximation factor have been proposed to solve different versions of this problem [17].

Finally, we note that our problem can be easily extended to take into account an incremental deployment of the WMN. In this regard, when topology changes occur (i.e., mesh routers join/leave the WMN) the network can be reconfigured considering the placement of already installed core nodes as fixed, while the position of eventual additional core routers is optimized adaptively. The investigation of the reconfiguration problem is left as future research issue.

5 Numerical Results

In this section we demonstrate the effectiveness of our proposed node placement model, measuring the performance improvement that derives from the optimal placement of core nodes in a fully distributed WMN security architecture, namely DSA-Mesh (reviewed in Section 3). A comparison with a centralized approach (MobiSEC) is also provided.

We simulated various WMN scenarios using Network Simulator (ns v.2). Packets are routed over shortest-paths, which are statically computed for all node

pairs; this is meant to reduce the overhead due to routing protocols, allowing us to evaluate more precisely the effect of our node placement model on the network performance.

We consider as performance figure the *delay* necessary to complete the periodic authentication protocol performed by the two security architectures, which provides an indication of the protocol responsiveness. In particular, we analyze the average and maximum delays experienced by all generic nodes to receive the response from the Key Server (in MobiSEC) or the last response from the core nodes (in DSA-Mesh). The optimal placement of the core nodes for each network scenario is obtained solving the ILP model described in Section 4 with the CPLEX solver.

The computational time we measured using a Pentium 4 with 3.0 Ghz and 2 GByte of RAM was always inferior to 20 seconds for network topologies comprising up to 40 nodes. We further tested our model with larger WMN scenarios, including up to 100 nodes, and the computational time never exceeded 10 minutes. Hence, the proposed model can be used to effectively design secure WMNs, and can be further envisaged to perform autonomic reconfiguration on highly reconfigurable network scenarios.

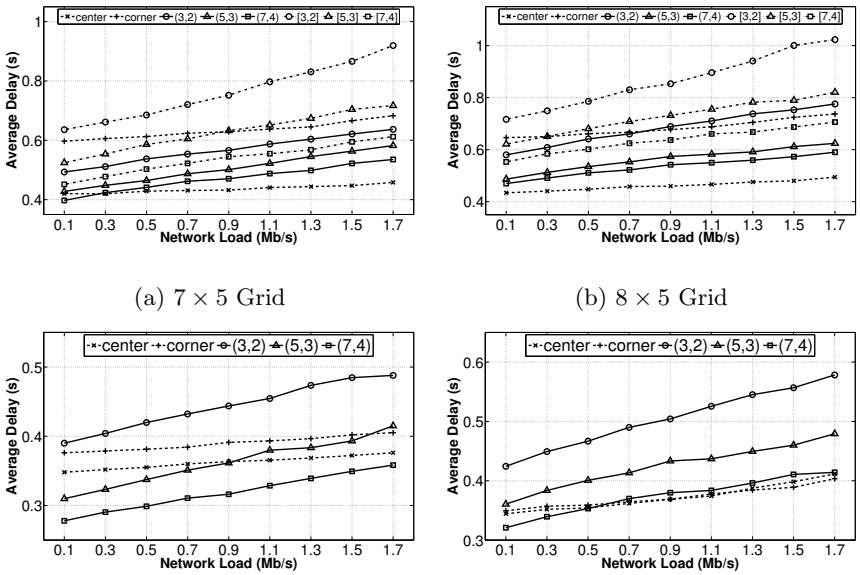
The aforementioned average and maximum delays were measured considering both Grid and Random network scenarios. More specifically, each Grid topology is composed of N nodes placed over a 2000 m $\times$ 2000 m area, with $N \in \{35, 40\}$. Random topologies are obtained by randomly scattering $N \in \{35, 40\}$ nodes over a 800 m $\times$ 800 m area.

To evaluate the behavior of the two architectures in realistic traffic conditions, we set up background UDP data transfers (with packet size equal to 1000 bytes) that involve disjoint network links.

The maximum transmission range of each wireless node is equal to 250 m; the carrier sensing range is 550 m when using the highest power level, equal to 0.1 W (these are ns v.2 default values). The maximum channel capacity is set to 54 Mbit/s, in accordance with commercial wireless card specifications (see Table 1). The reception threshold, the carrier sense and the capture thresholds are set to -64 dBm, -82 dBm and 10 dB, respectively. The path loss exponent, γ, is set to 2, and we leave as future issue the investigation of the model sensitivity to such parameter. All nodes use the same wireless channel since ns v.2 does not support natively multi-channel or multi-interface wireless nodes.

For each scenario we performed 10 independent measurements, achieving very narrow 0.95 confidence intervals, which we do not show for the sake of clarity. The simulation time on which we evaluated the performance was equal to 3000 seconds.

Figure 2 shows the *average delay* (for grid and random networks, respectively) measured by all generic nodes as a function of the network load imposed by the background UDP traffic. The lines identified by *"center"* and *"corner"* labels report the results obtained with the centralized security architecture (MobiSEC): the first line corresponds to a network configuration where the Key Server is installed at the center of the analyzed topology; on the other hand, the "corner"



(a) 7×5 Grid

(b) 8×5 Grid

(c) Random 35 nodes

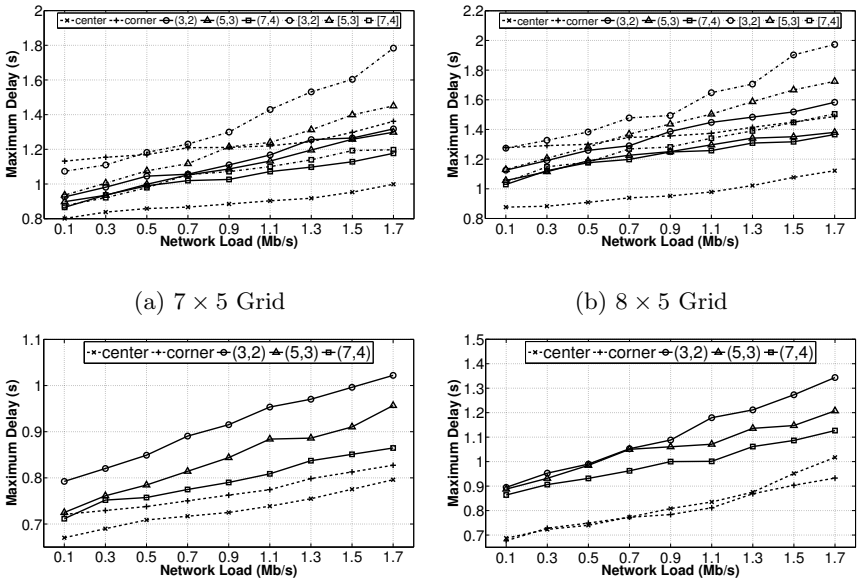
(d) Random 40 nodes

Fig. 2. Average Delay (seconds) measured in Grid and Random topologies with 35 and 40 nodes.

line reports the results obtained installing the Key Server at the topology border. The other curves in Figure 2 show the results obtained by the distributed security architecture (DSA-Mesh) using different (n, t) threshold schemes, viz. $(3, 2)$, $(5, 3)$ and $(7, 4)$. Solid lines correspond to the authentication delay measured when the core nodes are placed optimally, solving our ILP model (these curves are represented with the notation (n, t) in the legend), whereas dotted-dashed lines show this performance figure when the core nodes are simply installed at the topology corners (these curves are represented with the notation $[n, t]$). In this latter case, core nodes are placed in the corners of the grid network and, if $n > 4$, the remaining core nodes are placed randomly on the grid.

It can be observed that the average delay increases with increasing network load. At high traffic loads, the centralized architecture is more responsive than DSA-Mesh, when the Key Server is placed at the center of the network, since waiting for one authentication reply is less time-consuming than collecting t responses from the core nodes. However, when the Key Server is installed at the topology border (the "corner" curve) the centralized architecture exhibits higher delays, which are larger than those experienced by DSA-Mesh when the optimal placement is employed, regardless of the (n, t) scheme deployed. We further note that the optimal node placement permits to obtain consistent performance improvements, for any security scheme considered.

Figures 2(a) and 2(b) further show that the distributed architecture with seven core nodes placed in the topology corners (i.e., the $[7, 4]$ threshold scheme)



(a) 7×5 Grid

(b) 8×5 Grid

(c) Random 35 nodes

(d) Random 40 nodes

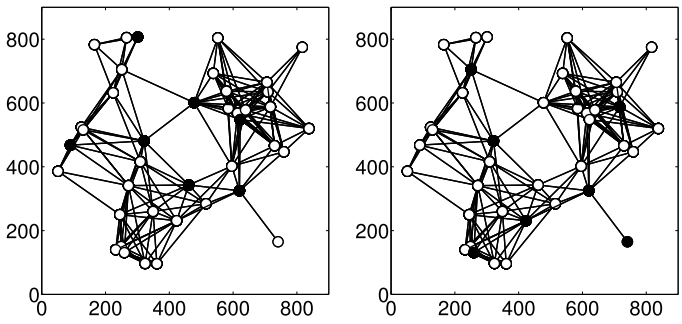
Fig. 3. Maximum Delay measured in Grid and Random topologies with 35 and 40 nodes.

exhibits in average a lower authentication delay than the centralized architecture with a single server (the "corner" curve). Hence, when a WMN operator cannot optimize the network design prior to its deployment, or when the WMN grows in an autonomous fashion, a distributed architecture with a high number of core nodes should be privileged with respect to a centralized system, since it provides better performance during the network operation.

Note that placing the Key Server in remote places, such as the corner node of a grid topology, can be the unique option in many network scenarios, where all the management services run on high-end machines.

In the Random topologies, the performance improvement of DSA-Mesh with respect to the centralized architecture is less evident than in the Grid scenarios, since the random network diameters are quite limited. In fact, the square area ($800\ m\ \times\ 800\ m$) over which the network nodes are distributed, bounds the distance of the worst path, and therefore the maximum value assumed by the authentication delay. As a consequence, the centralized architecture exhibits in all network scenarios similar performance. For sake of clarity, in these figures we did not show the results obtained placing core nodes at the topology edges, since they follow the same trends illustrated in the grid topologies of Figures 2(a) and 2(b).

To obtain a more complete comparison, we also measured the *maximum delay*, which provides an indication of the worst-case performance of the DSA-Mesh architecture. Figures 3 shows such performance metric for grid and random net-



(a) Rate metric (b) Hop Count metric

Fig. 4. Optimal placement of core nodes (the black circles) in a random topology with $N = 40$ nodes. Two different cost metrics are compared, namely the hop count and the overall transmission time (rate metric).

works composed of 35 and 40 nodes. The curves in these scenarios follow a trend similar to that obtained in the average case discussed above. More specifically, the optimal placement of the core nodes reduces considerably the maximum delay experienced by generic nodes, for any threshold scheme considered. On the other hand, the optimal node placement obtained using the model proposed in this paper permits to outperform a centralized architecture where the authentication and key distribution server is placed at the border of the network topology.

Finally, Figure 4 illustrates the optimal placement of the core nodes (the black circles) in a random topology scenario with 40 nodes and a (7,4) threshold scheme, comparing the metric introduced in Section 4 (the overall transmission time, referred to as Rate Metric) to a simple cost metric (the Hop Count, where the cost of each link is equal to 1). It can be observed that the Hop Count metric privileges paths with a low number of long wireless links (which, consequently, have low transmission rates), whereas the transmission time metric chooses routes with a high number of relatively short wireless links. As a consequence, the Hop Count metric tends to spread the core nodes over the network topology due to the limited distance among all node pairs, whereas the transmission time metric places the core nodes in areas highly crowded of network nodes.

Note that the different placements illustrated in Figure 4 lead to an average delay that is up to 30% higher for the Hop Count than for the Rate metric we propose in this paper.

6 Conclusion

This paper tackled the problem of determining the optimal placement of the core nodes that collaboratively implement the key management service in a distributed wireless security architecture. The core node placement has been

formulated as an optimization problem, modeling closely the behavior of real wireless channels; the performance improvement achieved solving our model has been evaluated in terms of key distribution/authentication delay (average and maximum) in several realistic network scenarios.

Numerical results show that our proposed modeling framework permits to increase both the average and the worst-case responsiveness in distributed architectures designed for WMNs with a short computing time, thus representing a very effective tool to design efficient and secure wireless networks.

References

1. I. F. Akyildiz, X. Wang, and W. Wang. Wireless mesh networks: a survey. *Elsevier Computer Networks*, 47(4):445–487, March 2005.
2. *IEEE Standard 802.11i. Medium Access Control (MAC) security enhancements, amendment 6.* IEEE Computer Society, 2004.
3. Y. Zhang and Y. Fang. Arsa: An attack-resilient security architecture for multihop wireless mesh networks. *IEEE Journal on Selected Areas in Communications*, 24(10):1916–1928, October 2006.
4. F. Martignon, S. Paris, and A. Capone. Design and Implementation of MobiSEC: a Complete Security Architecture for Wireless Mesh Networks. *Elsevier Computer Networks*, 53(12):2192–2207, August 2009.
5. S. Yi and R. Kravets. Moca: Mobile certificate authority for wireless ad hoc networks. *Annual PKI Research Workshop (PKI03)*, 2003.
6. J. Kim and S. Bahk. Meca: Distributed certification authority in wireless mesh networks. *IEEE CCNC*, pages 267–271, 2008.
7. F. Martignon, S. Paris, and A. Capone. DSA-Mesh: a Distributed Security Architecture for Wireless Mesh Networks. *Wiley Security and Communication Networks*, article in press, October 2009.
8. H. Luo, P. Zerfos, J. Kong, S. Lu, and L. Zhang. Self-securing ad hoc wireless networks. *ISCC*, pages 567–574, 2002.
9. G. Xu and L. Iftode. Locality driven key management architecture for mobile ad-hoc networks. *IEEE MASS*, pages 436–446, 2004.
10. S. Capkun, L. Buttyan, and J.-P. Hubaux. Self-organized public-key management for mobile ad hoc networks. *IEEE Trans. on Mobile Computing*, pages 52–64, 2003.
11. B. Wua, J. Wua, E. B. Fernandeza, M. Ilyasa, and S. Magliveras. Secure and efficient key management in mobile ad hoc networks. *IEEE IPDPS*, 2005.
12. R.M. Nauss. An improved algorithm for the capacitated facility location problem. *Journal of the Operational Research Society*, 29(12):1195–1201, 1978.
13. B. Aoun, R. Boutaba, Y. Iraqi, and G. Kenward. Gateway placement optimization in wireless mesh networks with QoS constraints. *IEEE Journal on Selected Areas in Communications*, 24(11):2127–2136, 2006.
14. V. Targon, B. Sansò, and A. Capone. The joint Gateway Placement and Spatial Reuse Problem in Wireless Mesh Networks. *Computer Networks*, 2009.
15. B. He, B. Xie, and D.P. Agrawal. Optimizing the Internet gateway deployment in a wireless mesh network. *IEEE MASS*, 2007.
16. E. Amaldi, A. Capone, M. Cesana, I. Filippini, and F. Malucelli. Optimization models and methods for planning wireless mesh networks. *Computer Networks*, 52(11):2159–2171, 2008.
17. S. Khuller, R. Pless, and Y.J. Sussmann. Fault tolerant k-center problems. *Theoretical Computer Science*, 242(1):237–246, 2000.