Decentralised Access Control in 802.11 Networks

Marco Domenico Aime[1], Antonio Lioy[1], and Gianluca Ramunno[1]

Politecnico di Torino, Dipartimento di Automatica e Informatica,
Corso Duca degli Abruzzi 24, 10129 Torino, Italy
{M.Aime, Lioy, Ramunno}@Polito.it

Abstract. The current WiFi access control framework descends from solutions conceived in the past for dial-up scenarios. A key difference between the two worlds is mobility: dial-up handles nomadic users, while modern wireless networks support continuous mobility through always-on personal devices. Not surprisingly, WiFi authentication does not exploit mobility in any way; on the contrary, mobility is perceived as a problem to be fixed by some fast-handoff solution. Though fast-handoff is indeed an open issue, mobility may even help to build security systems. The paper describes a decentralised access control framework for WiFi networks that exploits mobility to avoid a central authority to be always online.

1 Motivation

WiFi authentication and access control infrastructure, as defined in [1], relies on a central authority, the Authentication Server, to be always on-line as it is directly involved in each authentication attempt. A host of proposals have outlined changes to improve scalability and performance of this basic solution. [2] uses peer interaction among Access Points (APs) to move security contexts rather than creating new ones. Unfortunately, it is limited to APs within the same network segment and again requires the AP to interact with a remote entity (now another AP instead of the Authentication Server) for every authentication attempt. [3] brilliantly solves the network segment limitation and allows interactions with the Authentication Server to occur before the actual authentication attempts. Most interesting, in [4] the same authors propose a decentralised solution to let APs discriminate which authentication attempts they should expect. An AP learns which APs its users come from by tracking the source AP in each authentication attempt directly experimented: APs can thus foresee authentication attempts and proactively query the Authentication Server for proper credentials. The only limitation of [3] is the Authentication Server itself: though a central authority is a cornerstone in current network authentication architectures, it has some clear drawbacks. In particular, it is a single point of failure: when it falls, no authentication attempt can occur and the whole wireless network is stuck. This is perfectly acceptable when security is more important than availability, but it looks draconian in scenarios where availability plays a key role. Many modern applications of wireless networks do present this characteristic. For instance, in a museum where a wireless network delivers information only to paying visitors, the availability of the service is far more important than an eventual unauthorised access. Similar arguments apply to a road access control system

for vehicles: the cost of a false alarm (eventually triggering police intervention) is far more expensive than latency/failure in fraud detection. Nevertheless, in both examples efficient mechanisms are needed to revoke authorisation, for example when a customer definitely leaves the facility or the controlled area in general.

Recently, [5] has shown that mobility can be perceived as an aid to build security systems rather than a problem to be solved. We propose a decentralised access control solution that does not require a central authority to be always online and exploits node mobility to update authorisation status within the network. We recognise the main limit in that it requires advanced cryptographic techniques to be implemented both at client terminals and APs, and we will analyse the actual scope of this limit.

2 A decentralised access control framework

Here we investigate a novel framework where a central authority still exists but acts as a group manager rather than an authentication server. Its task is to admit and expel users from the group of authorised users, but it is not directly involved in each authentication attempt and access control decision. Fresh information on group membership is propagated by terminals as they roam within the network of APs. Recently admitted users will propagate their visa by their own, while membership revocation requires an ad-hoc transfer of information from the central authority to at least one AP. Revocation information can be either pushed by a remote network connection to a randomly chosen AP, or entrusted to a special terminal, or eventually delayed and given to a newly admitted terminal. In order to enable mutual authentication APs are equipped with the same class of credentials as the mobiles.

A straightforward way to implement our framework is a Public Key Infrastructure (PKI) based on digital certificates. The central authority would act as a Certification Authority (CA) and emit a certificate to every admitted user. Ownership of a valid certificate testifies group membership: APs may verify group admission by asking users for a certificate signed by the central authority. The overloading of public key certificates (PKC) with authorisation beyond identity information is a common practice even if more specialised techniques exist such as the attribute certificates (AC). However, revocation is the Achilles' heal of classic PKIs when deployed in fully decentralised scenarios. Revocation information is not propagated inside the certificates and should be retrieved by other means to correctly validate a certificate. In practice, a verifier must either query an online OCSP server or download an updated Certificate Revocation List (CRL) signed by the CA [6]. We deem both these solutions unsatisfactory because they involve the connection with an online authority that we aim to avoid. Note that CRLs could be propagated by terminals as for certificates, but there is no connection between admission and revocation information and thus no guarantee that they will be propagated with the same care. The relevance of this unsolved issue is clearly stated in [7].

As already observed, our task can be interpreted as a group membership problem, a well-established subject in security literature. In Sect. 3, we propose a simple extension to standard X.509 certificates able to satisfy our requirements. We then identify in the dynamic accumulator concept proposed in [8] an advanced cryptographic technique able to enhance our solution. Both solutions require fixed cryptographic burden for all

involved operations despite the current size of the group. In our framework this is a precondition to grant perfect scalability to large numbers of users. It also helps to estimate the requirements imposed on APs by cryptographic tools far more complex than current ones (even in latest security extensions [1], APs support only symmetric cryptography). The solutions rely on mobile terminals to propagate group membership information: to model it we referred to the literature about diffusion processes and epidemiological processes (see [9] for a broad spectrum review). In particular, in Sect. 4, we extend part of the analysis in [10] and adapt it to the peculiar requirements of our scenario. The involved dynamics are sensibly different since we want propagation to encompass the whole network while studies on viral spreading aim to limit the propagation of the infection. Moreover, we investigated the connection between terminal mobility patterns and the resulting propagation dynamics. Finally, in Sect. 5, we investigate how the system heavily depends on how information to be propagated is distributed within the mobile node population.

3 The protocol

We propose two mechanisms that can actually implement our framework, the former based on traditional PKIs and thus keener to current WiFi authentication system, the latter exploiting the advanced cryptographic techniques proposed in [11]. Both solutions entrust mobile terminals with the propagation of access control information as long as they travel within the AP network.

3.1 Basic solution

WiFi authentication infrastructure can already rely on digital certificates, though in a rather centralised fashion. The first step towards decentralisation is delegating the Authentication Server functionality to the APs. We argue that asymmetric cryptography's burden at APs is not a serious issue as it should be supported in any case to secure remote management. Nevertheless, it would be more problematic in environments experiencing fast mobility and strict authentication time constraints: in this case [3] may remain the best choice.

The Central Authority is in fact a Certification Authority and emits certificates to mobiles as they enter the community: schemes such as [12] can make this phase both secure and practical. Also APs are given certificates from the CA: they can be installed at deployment time through manual configuration or an imprinting mechanism as in [13]. Then, mobiles use their certificates to authenticate against APs through EAP-TLS or similar methods. Still, the main issue is revocation. We thus extend classic certificates to make revocation information easier to be spread by mobile nodes. The aim is twofold: (1) we eliminate the burden of transmitting huge membership information lists and (2) admission and revocation information are tightly joined.

Mobiles could propagate revocation information as CRLs. The CRL size would be a problem specially if revocation is frequent and/or certificate validity is long. For instance, in the museum example certificates could be revoked as customers exit the museum. Even worst, there is no guarantee that mobiles would propagate CRLs since there

is no connection between admission and revocation information. Thus, CRL emission should be frequent and APs should categorically refuse authentication if an updated CRL is not available: this imposes uncomfortable time constraints on the information spreading mechanism. Admission and revocation data may be linked simply by embedding CRLs within the certificates, but this is prevented by the size CRLs can grow to. We thus split revocation information within newly emitted certificates. Delta-CRLs [14] are a classic mechanism to limit the overhead due to CRL update. We extend this concept by having standard certificates to embed a subset of revocation data. This partial information is then spread by mobiles' movements and reconstructed at APs. The choice of a proper strategy to select which subset of information should be embedded in a particular certificate is addressed later in Sect. 5. Different embedding strategies may influence dramatically the security of the system. In fact, they determine the "information gap" probability, that is the probability that some revocation data is missing at APs. In this basic solution an information gap directly results in exposure to unauthorised access.

The above scheme fits scenarios where admissions are fairly frequent. For instance, it may work well in our museum example. We identify a key conceptual limit of this approach in that the nodes are obliged to spread fixed chunks of revocation data, but have no incentive to spread latest information: only recent certificates actually do valuable propagation job. An enhanced solution thus requires additional mechanisms able to push all members to look for fresh information and propagate it.

3.2 Enhanced solution

We further extend our proposal with the concept of dynamic accumulators. One-way accumulators are a novel cryptographic tool first introduced in [15]. A one-way accumulator allows to securely test the presence of a particular value within a set of values previously accumulated in a single fixed-size accumulator value. [8] extends the original construction to make the set of accumulated values to be dynamically changed.

We exploit a dynamic accumulator to build a compact representation of group membership. From [8], we retain the concept and implementation of a dynamic accumulator while renouncing to anonymous verification to avoid zero-knowledge proofs and their cryptographic burden. The Central Authority (let's identify it as CA) maintains a public key for a membership accumulator besides its usual public key. The accumulator public key consists in a RSA modulus $n = pq$ of length k, where p and q are safe primes ($p = 2p' + 1$ and $q = 2q' + 1$). During admission, the CA assigns to every mobile a prime e drown from a range $[A, B]$ where $2 < A < B < A^2 < n/4$. [1] The CA computes the new accumulator values as $z' = z^{e_a} \bmod n$, where z is the current accumulator value and e_a is the value assigned to the new member. Then the CA embeds $(e_a, u = z, z')$ within the terminal's certificate. When revoking a membership, the CA update the accumulator as $z' = z^{e_r^{-1} \bmod (p-1)(q-1)} \bmod n$, where z is the current accumulator value and e_r is the value inserted in the certificate being revoked.

To verify admission, an AP should both validate the mobile's certificate and check that the value e embedded within the certificate is still present in the latest accumulator value. To prove presence in the accumulator, a node associated to the prime e should

[1] Refer to [8] for a discussion on the choice of the range $[A, B]$.

show the witness u that satisfies $z = u^e \mod n$ where z is the latest accumulator value. Updated accumulator values z are spread by mobiles while filing their certificates.

The actual complexity in managing this scheme is updating the witness u. As [8] shows, a node should update its witness u for every change of the accumulator value. For every e_a added to the accumulator, the witness of every node must be updated as $u' = u^{e_a} \mod n$; while for every e_r removed from the accumulator, the new witness is $u' = u^b z^a$ where z is the new accumulator value, and a, b satisfy $ae + be_r = 1$ and are computed through the extended GCD algorithm. Hence, not only the fresh accumulator values but also the e added/removed from the accumulator should be propagated.

We argue that mobile-driven propagation may be exploited not only for accumulator values but also for accumulator change information. For instance, this can be achieved by having the CA to embed a subset of past changes (the e added/removed to/from the accumulator) in newly emitted certificate as done with revocation information in the basic solution of Sect. 3.1.

It's quite interesting to notice that a gap in the information being propagated has now quite different implications. As long as an AP knows the latest accumulator value it can safely prevent any unauthorised access. However, a legitimate terminal may not be able to prove its membership since it lacks data required to update its witness. Symmetrically, a terminal having updated credentials may not be able to authenticate an AP that has missed recent membership evolutions. These conditions are particularly scary in our reference scenarios where security must coexist with reliability. However, we notice that we have gained a lot of flexibility:

- *in policies*: nodes (APs or terminals) can flexibly choose between security and usability by accepting authentication attempts based on dated accumulator values
- *in resources*: nodes (APs or terminals) can tune the storage they reserve to accumulator history based on their policy
- *in fallbacks*: APs (and with more complex schemes also terminals) can fall back to expensive retrieval mechanisms just when needed and only for missing information chunks: alternatives are an online central directory or a peer-to-peer query system among APs

Related to the last point, note that the access to an online repository is going to be less frequent than in a classic CRL-based solution: once retrieved by a particular AP, missing information can then be propagated by terminals' movements.

Now, mobiles have additional incentive to propagate up-to-date information. In fact, a mobile will propagate the last known accumulator value to avoid storing multiple credential generations. This implies it will also tend to propagate membership changes that are needed to let APs update their own credentials. In practice, nodes will: (1) receive recent accumulator values from the APs they visit in the form of fresh certificates emitted by the CA, (2) update their own credentials, and (3) propagate fresh certificates containing the updated information.

The above schema is prone to further extensions. First, the terminal-driven propagation can be sided with a push mechanism among APs. At random intervals APs may send random chunks of information to a random selected peer (once again, information on AP community can be easily propagated by mobile terminals). A very low push-

ing probability can speed up propagation tremendously when mobility patterns are too much constrained: this is granted by the famous work of Watts and Strogatz [16].

Moreover, the anonymous credential system defined in [11] (the framework which dynamic accumulators where originally defined for) could further extend our framework and provide a key additional feature, namely untraceability of mobiles through anonymous authentication. In practice, the authentication process allows to verify the user authorisation without identifying her. If long-term terminal identifiers are hidden and node's movements are disguised, this ensures that the mobile presence, location and movements cannot be tracked within the covered area. Untraceability may be a key feature in public services covering large areas. Embracing the above anonymous credential system would require to drop standard certificates and would impose far higher cryptographic requirements to terminals and APs. However, the above construction would hold and terminal mobility could still be used to diffuse accumulator changes. An exhaustive analysis of the opportunities offered by an anonymous credential system and the relative performance impact are left for further investigation.

4 Terminal mobility and information propagation

Both our basic and enhanced solutions rely on terminal mobility to propagate information. Let us analyse their behaviour of our solutions in terms of information spreading.

We model the network of APs as a graph $G = (N, E)$ where N is the set of APs and E is the set of acquaintances. In other words, an edge $e_{j,i} \in E$ if $n_j, n_i \in N$, and a terminal can physically move from the AP n_j to the AP n_i. Then we refer to the viral spreading model presented in [10] and adapt it to our though different problem. This model aims to predict the dynamics of virus spreading in a computer network. We notice strong analogies with our propagation mechanism, where update information can only move from an aware AP to an unaware one thanks to the passage of a mobile between them. A main difference, that we will have to cope with, is that the probability of transmission cannot be assumed equal for all links as in [10], but heavily depends on the topology of the AP network and the terminal mobility patterns.

From the model described in [10], we retain some key quantities and overload them with different semantic:

$p_{i,t}$ – probability that the AP i has updated information at time t

$\beta_{j,i}$ – probability that updated information is propagated by any terminal from AP j to AP i

$\zeta_{i,t}$ – probability that AP i does not receive updated information from its neighbours at time t

Note that $\beta_{j,i}$ may now vary for each link. The quantity $\zeta_{i,t}$ is redefined as the probability that at time t an AP i has no updated information and will not receive it from any terminal coming from any of its neighbouring APs:

$$\zeta_{i,t} = \prod_{j:e_{j,i}\in E} (p_{j,t-1}(1-\beta_{j,i}) + (1-p_{j,t-1})) = \prod_{j:e_{j,i}\in E} (1-\beta_{j,i} * p_{j,t-1}) \quad (1)$$

We then use $\zeta_{i,t}$ to define $p_{i,t}$. In spite of the original model, in our case the "infected" status is irreversible: that is, once an AP has received a particular information chunk it can retain it indefinitely.[2] The quantity $p_{i,t}$ is thus computed as:

$$p_{i,t} = 1 - (1 - p_{i,t-1})\zeta_{i,t} \tag{2}$$

The probability $\beta_{j,i}$ is related to terminal mobility patterns. We model terminal mobility as a discrete Markov chain $M = (N, E^*)$ where N is the usual set of APs and E^* is the set E of links between APs weighted with the rate of terminals that transits along each link. We thus introduce two new quantities:

$a_{j,i}$ – probability that a terminal connected to AP j moves to AP i
π_i – probability that at a given time a given terminal is connected to AP i

Under ergodic conditions, using matrix notation we can compute $\boldsymbol{\Pi} = \boldsymbol{A}\boldsymbol{\Pi}$ as the principal eigenvector of the matrix $\boldsymbol{A}$, that is the eigenvector associated to the eigenvalue $\lambda = 1$. From theory, since each column of $\boldsymbol{A}$ adds up to one, at least one positive unitary eigenvalue exists, and for the ergodic assumption all other eigenvalues will be less than one.

Clearly, this mobility model is very simple. First, the model is discrete and thus the terminals are allowed to move only at discrete times. Second, it allows to model only constant numbers of terminals roaming within the network in a completely independent way. Third, the Markov assumption implies a memoryless behaviour of terminals, namely it is impossible to catch multiple highly-preferred directional paths along the network. Nevertheless, this model suffices to investigate relations between node mobility and information dissemination. In Sect. 4.1 we use it to analyse the behaviour of our access control framework against simple network topologies and highly guided mobility patterns. For instance, this may be the case in a wireless-enabled museum.

We can now define the probability $\beta^*_{j,i}$ that a given terminal propagates updated information from AP j to AP i as

$$\beta^*_{j,i} = a_{j,i}\pi_j \tag{3}$$

Assuming the same roaming pattern for all mobiles, we can finally compute the probability $\beta_{j,i}$ that some terminal propagates updated information from AP j to AP i:

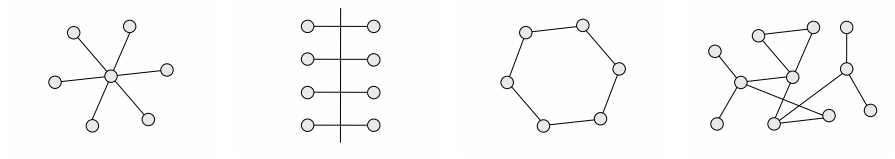
$$\beta_{j,i} = 1 - (1 - \beta^*_{j,i})^M \tag{4}$$

where M is the number of terminals present in the network, constant in time. Handling multiple roaming patterns requires to define a different matrix $\boldsymbol{A}$ per each pattern to determine a different β^* per each pattern.

4.1 Information propagation analysis

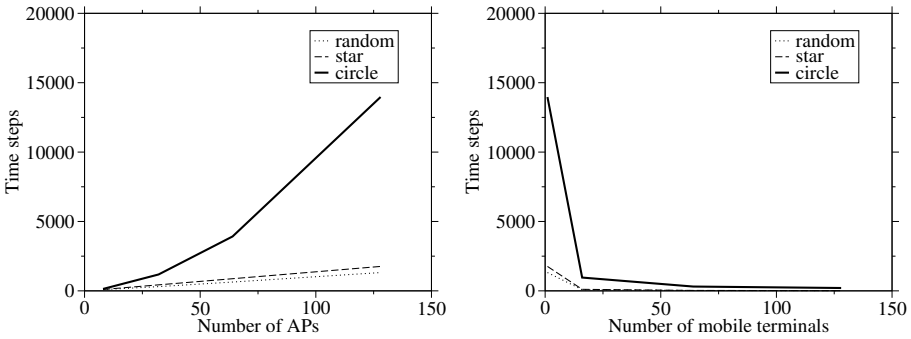
Rather than focusing on a particular AP topology, we chose to experiment our framework against a set of schematic topologies somehow related to typical architectonic structures. In particular, we selected the four topologies shown in Fig. 1:

[2] Actually, APs may purge obsolete information once the certificate it refers to is expired, but this lies outside the spreading analysis.



(a) Star (b) Ladder (c) Circle (d) Random

Fig. 1. Different AP topologies



(a) Propagation with a single terminal (b) Propagation in topologies of 128 APs

Fig. 2. Propagation in different AP topologies

- star – this represents an hall/square surrounded by a set of places; the square is a common building as well as city architectural module since ancient Greece
- ladder – represents a corridor/street with a sequence of places on both its sides; this is a module made famous by Roman cities
- circle – in our simple mobility model, this is the best representative of a guided corridor/street; this may model a museum as well as an highway
- random mesh – this is mainly used for comparison but it may model for example a large exposition ambient

Figure 2(a) shows how the number of discrete time steps t needed to have $p_{i,t} > 0.99$ for all APs changes based on the number of APs. Figure 2(b) shows instead how t changes based on the number of mobile terminals in different topologies of 128 APs. As expected, propagation may perform poorly when mobility paths are particularly constrained (as in the circle topology). However, as the number of mobiles grows the probability of a jump between two given APs rapidly increases and so does the propagation speed.

5 Information embedding strategy

Both the solutions discussed in Sect. 3 rely on mobile terminals to carry some chunks of information, either revocation data under the certificate-based solution or accumulator updates within the enhanced solution. However, In Sect. 3 we have put off the definition of the actual embedding strategy, that is the choice of what subset of information has to be included in each newly emitted certificate for propagation. To complete the analysis of our proposal, we present an initial investigation on the tremendous effects embedding strategies may have on its global behaviour.

Following the idea behind Delta-CRLs, a sliding window selection mechanism may be used. In other words, a newly emitted certificate includes all and only the changes occurred from the immediately previous certificate. In this case, the probability P_{gap} that some block of information gets permanently lost can be computed as:

$$P_{gap}(T) = 1 - (1 - P_m)^T \tag{5}$$

where P_m is the probability that a single certificate gets lost and T is the number of emitted certificates. P_m highly depends on the specific scenario: for instance, it is effected by the probability that users subscribe to the service but do not use it, and the threat of sabotage attempts. We argue that a careful analysis of proper values of P_m is crucial and leave it to future investigation. Nevertheless, from (5) it is evident that P_{gap} will rapidly approach 1 even for low values of P_m. The problem is that a single missing certificate is sufficient to create a permanent gap in the information being propagated.

To overcome the poor performance of the sliding window approach, we propose to randomly choose the subset of information to be embedded. To limit the size of embedded information, the probability that a specific information gets embedded is decreased as its freshness. Assuming membership change events are fully ordered,[3] we define the probability P_e that the information chunk at ordinal number t gets embedded in a new certificate emitted at time T as

$$P_e(t, T) = \frac{1}{(T - t)^\alpha} \tag{6}$$

The parameter α determines the size of the embedded information in a single certificate, as well as the expected number of certificates that a particular information is embedded in. Table 1 shows the expected number of information chunks embedded in a certificate with different values of α and different sizes T of information history. Note that the behaviour with T approaching infinity is not relevant since old information sooner or later can be dropped because of certificate expiration.

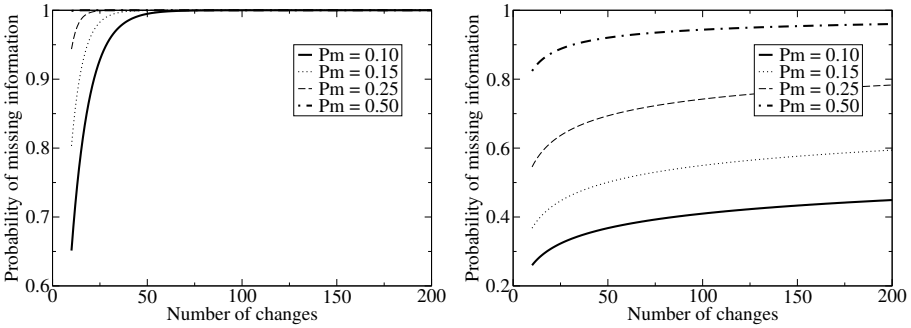
To analyse the performance of our probabilistic embedding strategy, we start computing the probability that the t-th information is not lost at time T when supposing that all the successive certificates are not lost:

$$P_{presence}(t, T) = 1 - (P_m * \prod_{2<s<(T-t)} (1 - P_e(s, T))) \tag{7}$$

[3] Full orderability is obviously guaranteed by allowing a single entity, the CA, to modify the set.

Table 1. Expected number of embedded information chunks

	$T = 10$	$T = 100$	$T = 1000$	$T = 10000$
$\alpha = 0.9$	2.68	4.28	5.57	6.60
$\alpha = 1$	2.93	5.19	7.49	9.79
$\alpha = 1.1$	3.22	6.43	10.52	15.69



(a) Sliding window

(b) Probabilistic embedding

Fig. 3. Comparison of different embedding strategies

The above equation states that the t-th information chunk is present if the t-th certificate is not lost or if it has been embedded in some of the successive certificates. Now we can compute P_{gap} at time T as:

$$P_{gap}(T) = 1 - ((1 - P_m) * \prod_{2<t<T} (P_{presence}(t, T))) \tag{8}$$

This states that we face a gap when not all the information is somehow present, either in its native certificate or embedded in successive ones.

Figures 3(a) and 3(b) show how the two discussed strategies behave with different values of the probability P_m that a single certificate gets lost: as expected, with the sliding window the gap probability rapidly tends to one, while the probabilistic embedding leaves additional space to recovering.

6 Final remarks

The analysis presented here suggests further investigation. First, a more realistic mobility model could help to better understand how our approach fits in real environments. In particular, it could prove interesting to understand when the integration of terminal-based propagation with push mechanisms by APs may be useful and the achievable efficiency. Our probabilistic embedding strategy should be tested against coalition of adversaries trying to prevent or manipulate the information spreading. We argue that a

detailed comparison among different strategies could help to measure the actual robustness of our solution. As already observed, a major extension is related to the integration of an anonymous credential system. This could boost the value of our construction in environments where privacy is a concern. A careful performance analysis of the specific credential system is a key step towards this opportunity. Finally, we argue that an implementation of the specific mechanisms we have described could help to gain additional insight in the whole system behaviour. Actually, this step is unavoidable to understand whether fully decentralised authentication frameworks may challenge traditional ones in future wireless networks.

References

1. IEEE: Std 802.11i/d7.0, part 11: Wireless medium access control (MAC) and physical layer (PHY) specifications: Medium access control (MAC) security enhancements (2003)
2. IEEE: P802.11f/d5, recommended practice for multi-vendor access point interoperability via an inter-access point protocol across distribution systems supporting IEEE 802.11 operation (2003)
3. Mishra, A., Shin, M., Arbaugh, W.A.: Proactive key distribution to support fast and secure roaming. Submission to IEEE 802.11 Working Group 802.11-03/084r0 (2003)
4. Mishra, A., Shin, M., Arbaugh, W.A.: Pro-active key distribution using neighbor graphs. Technical report, Department of Computer Science, University of Maryland College Park (MD, USA) (2003)
5. Capkun, S., Hubaux, J.P., Buttyan, L.: Mobility helps security in ad hoc networks. In: Proc. of the 4th ACM international symposium on Mobile Ad Hoc Networking & Computing (MobiHoc). (2003) 46–56
6. Wohlmacher, P.: Digital certificates: a survey of revocation methods. In: Proc. of the 2000 ACM workshops on Multimedia. (2000) 111–114
7. Rivest, R.L.: Can we eliminate certificate revocations lists? In: Proc. of Financial Cryptography (FC). (1998) 178–183
8. Camenisch, J., Lysyanskaya, A.: Dynamic accumulators and application to efficient revocation of anonymous credentials. In: Crypto # 2002. (2002) 61–76
9. Newman, M.E.J.: The structure and function of complex networks. In: SIAM Review. Volume 45(2). (2003) 167–256
10. Wang, Y., Chakrabarti, D., Wang, C., Faloutsos, C.: Epidemic spreading in real networks: An eigenvalue viewpoint. In: 22nd Symposium on Reliable Distributed Systems (SRDS). (2003) 25–34
11. Camenisch, J., Lysyanskaya, A.: An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In: EuroCr # 2001. (2001) 93–117
12. Balfanz, D., Smetters, D.K., Stewart, P., Wong, H.C.: Talking to strangers: Authentication in ad-hoc wireless networks. In: Proc. of Network and Distributed System Security Symposium (NDSS), San Diego (CA, USA) (2002)
13. Stajano, F., Anderson, R.: The resurrecting duckling: Security issues for ad-hoc wireless networks. In: Proc. of the 7th International Workshop on Security Protocols, Cambridge (UK) (2000) 172–194
14. Cooper, D.A.: A more efficient use of delta-CRLs. In: IEEE Symposium on Security and Privacy (S&P). (2000) 190–202
15. Benaloh, J., de Mare, M.: One-way accumulators: A decentralized alternative to digital signatures. In: EuroCr # 93. (1994) 274–285
16. Watts, D., Strogatz, S.: Collective dynamics of 'small-world' networks. Nature **393** (1998) 440–442