Fully Distributed Cooperative Spectrum Sensing for Cognitive Radio Networks

Carles Garrigues
Estudios de Informática, Multimedia y Telecomunicación
Universitat Oberta de Catalunya
Email: cgarrigueso@uoc.edu

Helena Rifà-Pous
Estudios de Informática, Multimedia y Telecomunicación
Universitat Oberta de Catalunya
Email: hrifa@uoc.edu

Guillermo Navarro-Arribas
DEIC, Departamento de Ingenieria de la Información y de las Comunicaciones
Universitat Autònoma de Barcelona
Email: guillermo.navarro@uab.cat

***Abstract*—Cognitive radio networks (CRN) sense spectrum occupancy and manage themselves to operate in unused bands without disturbing licensed users. The detection capability of a radio system can be enhanced if the sensing process is performed jointly by a group of nodes so that the effects of wireless fading and shadowing can be minimized. However, taking a collaborative approach poses new security threats to the system as nodes can report false sensing data to force a wrong decision. Providing security to the sensing process is also complex, as it usually involves introducing limitations to the CRN applications. The most common limitation is the need for a static trusted node that is able to authenticate and merge the reports of all CRN nodes. This paper overcomes this limitation by presenting a protocol that is suitable for fully distributed scenarios, where there is no static trusted node.**

I. Introduction

Spectrum is an essential resource for the provision of mobile services. In order to control and delimit its use, governmental agencies set up regulatory policies. Unfortunately, such policies have led to a deficiency of spectrum as only few frequency bands are left unlicensed, and these are used for the majority of new emerging wireless applications. Besides, studies conducted by the Spectrum Policy Task Force show that most of the licensed spectrum is largely under-utilized [1]. Thus, a large portion of the assigned spectrum is only used sporadically, and this has led to an unnecessary shortage of spectrum.

A promising way to alleviate the spectrum shortage problem is adopting a spectrum sharing paradigm in which frequency bands are used opportunistically. In a spectrum sharing scenario, those who own the license to use the spectrum are referred to as primary users, and those who access the spectrum opportunistically are referred to as secondary users. Secondary users must not interfere with primary ones, who always have usage priority.

The enabling technology for opportunistic sharing is Cognitive Radio (CR) [2]. A CR is a system that senses its electromagnetic environment and can dynamically and autonomously adjust its operating parameters to access the spectrum.

CR terminals form self-organizing networks capable to detect vacant spectrum bands that can be used without harmful interference with primary nodes. Once a vacant band is found, secondary nodes coordinate themselves in order to share the available spectrum.

Performing reliable spectrum sensing is a difficult task. This is mainly due to the fact that wireless channels can suffer fading, and this can result in nodes failing to detect a primary transmitter. This is known as the hidden node problem. As the most important challenge for a CR is to identify the presence of primary nodes, secondary nodes must be significantly more sensitive in detecting primary transmissions than primary receivers.

In order to reduce the sensitivity requirements of individual CRs, recent studies propose performing distributed spectrum sensing (DSS)[3]. In DSS, multiple secondary nodes cooperate and share their local sensing results, which are then merged together to reach a final decision. These protocols assume that reports from secondary nodes can be effectively authenticated. As a result, malicious nodes can be detected -their reports repeatedly differ from the final decision- and their contributions discarded. However, the mechanisms proposed to date to authenticate the observations sent by secondary nodes are based on the introduction of requirements that usually make the security solution hardly applicable to real scenarios.

The most common restriction imposed by authentication protocols is the need for a static trusted node. This trusted node is mainly used to:

- access a Public Key Infrastructure that allows verifying the validity of public key certificates,
- merge the sensing results shared by the secondary nodes and make a joint decision, which is trusted by all the nodes

Obviously, the trusted node simplifies the protection of the spectrum sensing protocol. However, unfortunately, this kind of entity is not present in all CRN scenarios. On the contrary, CR networks are characterised by:

- being created in an ad-hoc fashion by nodes that have no a priori knowledge of each other,
- being very dynamic, in the sense that there is no node that is always present,
- having limited access to Internet and, thus, to a Public Key Infrastructure.

Therefore, in this paper we present a protocol that allows

conducting the cooperative spectrum sensing in a fully distributed way. This means that our protocol allows authenticating the sensing results and making a joint decision without the requirement of a static trusted node. The protocol has been designed, additionally, to minimise the used bandwidth, since the opportunistic access to the spectrum requires a very efficient use of the network.

The rest of the paper is structured as follows. Section II introduces the related work on cooperative spectrum sensing. Section III presents the main characteristics of the security framework proposed. Sections IV, V and VI describe the three main parts of the cooperative sensing protocol. Section VII provides a discussion on the main aspects of the protocol. Section VIII concludes the paper and points out future directions.

II. Background

Cooperative sensing is based on merging the local observations of multiple secondary nodes. Since local spectrum sensing results are subject to multipath and/or shadowing fading, the cooperation among CRs is fundamental to achieve a reliable decision.

Several data fusion schemes have been proposed to merge the sensing data observed by every secondary node. Among the proposed methods, the most typical one is based on applying the "k out of N" rule. This rule determines that the channel is occupied if at least k of the N secondary nodes have detected the primary signal. As avoiding interference with primary nodes is a top priority, the most common value of k is 1.

Many other methods have been proposed for merging the sensing data. In [4], authors review the main cooperative sensing protocols that assume the existence of malicious nodes in the network and try to nullify their effects. As it can be seen, several proposals are intended to increase the security of the cooperative sensing process by authenticating the sensing results contributed by each node [5], [6], [7]. Indeed, including authenticating mechanisms in the cooperative sensing protocols is fundamental, as it allows building reputation systems in which the identity claimed by each node can be verified, and so the reliability of each contribution can be weighed correctly according to the reputation of the node.

The problem, as is usually the case, is that the inclusion of security mechanisms in the protocols leads to the introduction of restrictions in the design of the network and its applications. Thus, secure cooperative sensing protocols proposed so far are based on the assumption that the CRN contains a node that can play the role of a fusion centre. The fusion centre is conceived as a well-known and static entity in the network that manages the spectrum of a certain area and connects to the Internet and to any Certification Authority (CA), if required. The role of the fusion centre is assumed by a secondary base-station or a spectrum broker.

The problem is that this trusted central node is not necessarily found in many CRN scenarios, because the network is completely dynamic and is created in an ad-hoc manner. Additionally, from the security point of view, this central node becomes a single point of failure, which can easily compromise the security of the whole network if it is successfully attacked.

In this paper, thus, we face the challenge of designing a cooperative sensing protocol that allows CRN nodes to sense the spectrum and decide whether the channel is free or occupied in a fully distributed way, without depending on a secure and trusted fusion centre.

Before describing our protocol in detail in sections IV, V and VI, the following section presents the security framework that we have assumed in order to define our protocol.

III. Security framework

One of the key goals of the protocol design is to develop an efficient solution suitable for constrained devices. Therefore, the cryptography involved in our proposal is based on simple hash functions and symmetric keys.

The use of symmetric keys is essential to implement the authentication of the sensing data provided by the secondary nodes. However, the main challenge of symmetric key systems is how to distribute and manage the keys among the authorized nodes. Different lightweight processing solutions have been proposed in the scope of sensor networks, which pre-distribute or dynamically generate the secret keys using probabilistic approaches (see a review in [8]). However, such schemes are impractical for CRNs due to the particular features that differentiate a CRN from a traditional sensor network, namely:

1) The topology of CRNs is continuously changing. Some sensor networks are dynamic in the sense that they allow addition and deletion of sensor nodes after deployment to extend the network or replace failing and unreliable nodes without physical contact; however, the dynamism of CRN goes further: nodes are mobile and join and leave a particular community in short periods of time.
2) The number of network members is several orders of magnitude larger than that of sensor networks. The number of connected members in a particular moment is similar to a sensor network, but in an open CRN network, the number of potential nodes is unlimited and so, key management must be highly scalable. Moreover, it must allow the addition of new nodes in the system in the course of time as opposed to admitting them all at once at system start-up.
3) In a CRN, the channels can only be used for limited periods of time (while primary nodes are not active). Thus, the time available for data transmission must be maximized and the security protocols must be designed in such a way that they introduce the minimum possible overhead.

With these challenging operational requirements, the use of sensor network designed schemes for key distribution becomes too complex in CRNs.

As we already mentioned, the main goal of our proposal is to enable the secure authentication of nodes' sensing reports,

and to allow nodes to take a joint decision on whether the channel is free or occupied.

In order to achieve these goals, we have defined a protocol that allows both hard-decision and soft-decision techniques. Thus, nodes can exchange all the information of their local observations, or they can share only their individual 1-bit decisions (free or occupied).

In order to identify network nodes, we take advantage of the fact that CRNs are suitable for more powerful devices than sensor networks are, so we take a public key infrastructure (PKI) approach to initialize the network. However, as this process is costly, it is executed only once, when a new node joins the cognitive radio network. From that moment on, the messages are signed using efficient Hash Message Authentication Code (HMAC) functions.

HMAC functions provide message authenticity and integrity by calculating a hash of two inputs: the target message and a secret key. In our protocol, we use hash chains to produce the one-time secret keys. Hash Chains, first proposed by Lamport [9], are versatile low-cost constructions that are used extensively in various cryptographic systems.

We also take advantage of the fact that messages exchanged between nodes are sent through a wireless environment, which means that all nodes can receive the messages as if they were sent to a broadcast address.

The design of our proposed protocol also requires the acceptance of some assumptions, which we describe below:

- The exchange of messages between the secondary nodes is carried out through a common control channel. The mechanism used to implement this control channel is out of the scope of this paper.
- Every network node has a public key certificate, which the node has obtained before entering the cognitive radio network. Nodes also have the public key of the Certification Authority that generates the public key certificates.
- The cognitive radio network is not permanently disconnected from the Internet. This means that the network usually has one or more nodes with Internet connection, and thus they have access to a Public Key Infrastructure. The nodes that have Internet connection do not necessarily have to be the same ones all the time.
- Every node uses the same protocol to merge the sensing results of other nodes and update the corresponding reputations.

With these assumptions, we have defined a protocol that is based on three main procedures: election of a coordinator node, node registry, and cooperative spectrum sensing. In the following sections, we describe each one of these procedures in detail.

IV. Electing a coordinator node

As we have already mentioned, our protocol is intended to allow nodes of a CRN to decide if a channel is free or occupied in a secure and fully distributed way, without depending on any static trusted node. In order to make this possible, however, our protocol is based on the use of nodes that centralise and coordinate the authentication of nodes and the joint decision-making. Nonetheless, nodes taking this role do not necessarily have to be trusted by the other nodes, they do not need to be known in advance, and they can vary over time.

In our proposal, unlike centralised protocols, all network nodes can verify the authenticity of the other nodes' messages, merge the different results, calculate the final sensing decision, and update the reputation assigned to the other nodes.

The fact that all nodes can merge the sensing results and calculate the reputation of the other nodes allows verifying the reliability of the coordinator node. Thus, a coordinator whose decisions are repeatedly different from those of the majority will be rapidly considered badly, and this will result in the replacement of this coordinator.

As we can see, additionally, the fact that all nodes have all the information needed to authenticate and calculate the reputations of the other nodes allows coordinator replacements to be quick, and therefore feasible. Otherwise, the information regarding identities, reputations, signing keys, etc. would have to be transmitted from one coordinator to the next every time the coordinator was changed. Besides, as all nodes share this information, the coordinator role can be easily assumed by any node.

In order to elect the coordinator node, as we have previously mentioned, we assume that the CRN is usually connected to the Internet. This means that there is usually some node that has Internet connection and, thus, can connect to a Validation Authority (VA). Only nodes with Internet connection can assume the coordinator role. The election process, which is based on the clusterhead election algorithm for ad-hoc networks presented in [10], is as follows:

1) nodes with Internet connection announce their availablity to play the coordinator role.
2) from these candidates, nodes select the one with the highest reputation value from their reputation table.
3) nodes send their vote in a signed message.
4) the winner is decided based on simple majority. The previous coordinator sends a broadcast message to announce the winner.
5) If nodes disagree with the previous coordinator's message, they send a reply with its choice of candidate. If at least 50% of the nodes support this last candidature, the elected coordinator is the last one.

Once the new coordinator is elected, it obtains a signed confirmation of the validity of its public key certificate from a Validation Authority (VA). This confirmation is broadcasted so that the other nodes can verify the messages signed by this new coordinator securely from this moment on.

Additionally, in order for the newly elected nodes to be trusted throughout the different sensing sessions, this election process is repeated periodically. The exact periodicity to use will not be the same for all possible CRN scenarios. It can depend on the specific parameters or characteristics of the network, such as the average node trustworthiness, or the network's degree of dynamism, etc. Moreover, this periodicity

can be extended over time if the coordinator node repeatedly proves that it can be trusted.

To avoid unnecessary processes of coordinator replacement, before starting a new election process, nodes vote if the current coordinator is not trustworthy any more and it is necessary to replace it. In order to decide their vote, nodes use the reputation that the coordinator has built over the different sensing sessions, as we will see in section VI.

It is also worth noting that, if the coordinator loses its Internet connection, it must request a coordinator replacement, as it can no longer verify the identity of new network nodes by connecting to a Certification Authority.

V. Node registry

In order to register to the CRN, a new node has to perform the following steps:

1) The new node Q selects a random number R_Q and prepares a hash chain of length N, where N is selected according to the memory availability of Q. The values of this hash chain will be used as the signing keys of the HMAC functions:

$$V_Q[N-1] = R_Q$$
$$V_Q[N-2] = hash(R_Q)$$
$$V_Q[N-3] = hash(hash(R_Q)) = hash(V_Q[N-2])$$
$$\cdots$$
$$V_Q[0] = hash(V_Q[1])$$

Thus, $V_Q[0]$ is the first signing key used by Q, $V_Q[1]$ is the second, and so on. Obviously, as $V_Q[i-1] = hash(V_Q[i])$, by knowing $V_Q[i]$ we can calculate $V_Q[i-1]$ easily. However, by knowing $V_Q[i-1]$, it is theoretically impossible to compute $V_Q[i]$.

2) Q sends a broadcast message requesting to register to the CRN
3) The coordinator node replies requesting the authentication of Q. This reply is signed with the coordinator's private key.
4) Q sends the following authentication information:
   - Q's public key certificate
   - The top value of Q's hash chain ($V_Q[0]$) signed with Q's private key.
5) The coordinator connects to a VA to obtain and broadcast the signed confirmation of the validity of Q's public key certificate. This allows the other nodes to verify the messages signed by Q from this moment on.
6) The other nodes of the CRN store $V_Q[0]$ as the first signing key to be used by Q. Thus, all those nodes that have been able to listen to the previous steps 1 to 5 will be able to verify the authenticity of the messages signed by Q, using an HMAC function and the hash chain values $V_Q[0]$ to $V_Q[N-1]$.

VI. Cooperative Spectrum Sensing

This section presents the procedure followed by cognitive radio nodes to authenticate the sensing reports of every node and calculate a common sensing decision. The protocol prevents nodes from illegitimately claiming false identities and from injecting fake sensing data. Thus, this protocol aims at withstanding the following attacks:

- Altering the final sensing decision. A node could increment her weight in the data fusion process by forging several identities and making a contribution for each of them. With enough forged identities, a node might be able to completely alter the aggregate reading.
- Deceiving the reputation system. By using a different identity each time, a node might report false sensing data repeatedly and avoid earning a bad reputation.
- Obtaining resources unfairly. A node could use many identities to obtain more than her fair share of resources (e.g. bandwidth).

This part of our proposed protocol is the most important one, as it allows:

- sharing the signing keys, reputations and other node information securely, rapidly and efficiently.
- calculating the reputation of the coordinator node, so that the CRN nodes can decide jointly if this node cannot be trusted any more and has to be replaced by another one.

Thus, as we will see, this part of the protocol allows new nodes to obtain the reputations and signing keys of all the other nodes. This information, as we have seen in the previous section, has not been transmitted to the new nodes during the registration process, thus saving a lot of network bandwidth.

The protocol is based on the fact that each node s uses a reputation table built as follows:

$$ReputationTable_s = [\ [Id_1, Reputation_s[1], V_1[i]], \cdots, [Id_S, Reputation_s[S], V_S[j]]\ ]$$

where S is the total number of nodes in the network at a given moment, Id_i is the identifier of node i, and $\{i \cdots j\}$ is the set of indexes to the hash chain values of each node. Thus, as we can see, this table contains the list of reputations and signing keys of all CRN nodes.

The steps required to perform the cooperative spectrum sensing are as follows:

1) The coordinator node announces which channels have to be sensed for the current sensing session.
2) Each node s senses the spectrum and reports the following information:

$$Result_s = HMAC_{V_s[i]}\ (Id_s, SensingResult_s, Reputation_s[s])$$

where

- $HMAC_{V_s[i]}$ is the result of applying an HMAC function using the hash chain value $V_s[i]$ as the signing key. The information is also sent in cleartext, so as to allow nodes to obtain the data and verify the HMAC signature.
- $SensingResult_s$ is the sensing result of node s
- $Reputation_s[s]$ is the reputation that node s has calculated for itself throughout the different sensing sessions.

3) The coordinator node, once the time needed by all nodes to share their sensing results has elapsed, requests all nodes to send the signing key ($V_s[i]$) used to build their sensing result in the previous step.
   At this point, we assume that the exchange of messages is synchronised in such a way that it is possible to detect that a message is invalid if it has not been sent at the correct time.
4) Each node s sends the signing key used in step 2: $V_s[i]$
5) With the information provided by the sensing results sent in step 2, each new node Q creates its $ReputationTable_Q$. As we have seen in the registration process, no information about the reputation or the signing keys of other nodes is provided to Q when it joins the network.
6) Each node s builds a list of results to ignore:

$$IgnoreList_s = [\ [Id_{w_1}, Reputation_s[w_1]], \cdots, [\ [Id_{w_i}, Reputation_s[w_i]], \cdots, [\ [Id_{w_Z}, Reputation_s[w_Z]]\ ]$$

   where Z is the number of results to ignore, and nodes $\{Id_{w_1}, \cdots, Id_{w_Z}\}$ are those that:
   - the HMAC signature verification has failed, or
   - the reputation reported by the node is different from that stored by node s, or
   - there is no evidence of the node having registered to the network, or
   - no sensing result has been received,

   and $Reputation_s[w_i]$ is the reputation stored by s for node w_i, or $null$ if w_i is not in the reputation table.
7) Each node s calculates the final sensing decision from the sensing results received. In order to weigh up each node's contribution according to its reputation, s uses the weighted fusion approach presented by Chen *et al.* in [11].
8) From the final sensing decision made in the previous step, each node s updates the reputations of the other nodes and stores them in $ReputationTable_s$. The algorithm used to update these reputations is again the one presented by Chen *et al.* in [11].
9) The coordinator broadcasts the list of results to ignore ($IgnoreList_c$) and its final sensing decision.
10) Each new node Q corrects its $ReputationTable_Q$ with the information provided by $IgnoreList_c$.
11) Each node s compares its final sensing decision and its list of nodes to ignore ($IgnoreList_s$) with the corresponding information from the coordinator node. From this comparison:
    a) s updates the coordinator's reputation. Again, the algorithm used to update the coordinator's reputation is the one presented by Chen *et al.* in [11]. In this case, however, this algorithm is applied taking into account that having a different final sensing decision decreases the reputation a lot more than having individual differences in the list of results to ignore.
    b) s completes its reputation table, including the information of those nodes that were not present in $ReputationTable_s$ and were not included in $IgnoreList_c$.

After performing the steps above, the final decision announced by the coordinator in step 9 is the one followed by all the secondary nodes to decide whether the channel is free or occupied. However, as we have seen in steps 7 and 8, every node computes its own final decision and updates its own reputation table. Thus, nodes can compare their decision and their reputation table with the coordinator's. This comparison performed in step 11 allows nodes to decide whether the coordinator has to be replaced, which is decided by vote as we have seen in section IV.

VII. Discussion

A remarkable aspect of the proposed protocol is that it allows introducing new nodes in the CRN without the transmission of all node reputations and signing keys. Thus, we are avoiding the sending of an excessive amount of information over the network, mainly when the number of nodes is high.

The protocol also allows a node to authenticate the sensing results of the other nodes, even when it has no access to a Public Key Infrastructure. This is due to the confirmation of the validity of new nodes' public key certificates, which the coordinator obtains from the Validation Authority and broadcasts to the other nodes.

The information shared by nodes is validated by the coordinator indirectly. This means that new nodes can verify that the reputations and the signing keys reported by the other nodes are valid if the coordinator does not include them in its $IgnoreList_c$. Additionally, nodes can update their reputation tables with the information of the other nodes when they detect that the reputation given by the majority to the coordinator is significantly different from their own.

The protocol uses a small amount of bandwidth due to the use of HMAC functions for message authentication. Once the node has announced its public key and first signing key, all subsequent messages are signed using lightweight HMAC functions. This reduces the amount of transmitted information significantly. The HMAC keys can be generated and checked with efficient mechanisms for fast chain traversal [12] and for economic setup and verification [13]: a one-way chain with N elements only requires log(N) storage and log(N) computation to access an element.

From the security point of view, the proposed system is robust against Sybil attacks, in which a user illegitimately claims multiple identities, and the injection of false sensing reports. Sybil attacks are prevented using certificates generated by a Certification Authority. If a node does not own a valid certificate, it is not authorized in the CR network and cannot send sensing reports to the other nodes. On the other hand, the injection of false sensing reports is avoided using verifiable HMAC signed reports.

The protocol does not use any static fusion centre or trusted node, thus avoiding the problem of having a single point of

failure, which might compromise the security of the whole network if it was successfully attacked. Besides, this makes the protocol suitable for a wider range of CRN applications in which this kind of entity does not exist.

VIII. Conclusions

Cooperative sensing protocols are vulnerable to malicious attacks that can result in erroneous decisions: the failure to recognize primary node signals and thus provoke inconvenient interferences; the mistake to consider a channel is occupied and cannot be used for CR users; or simply making an unfair distribution of the encountered free spectrum.

In this paper, we have presented a protocol to perform cooperative spectrum sensing securely. The protocol allows authenticating the sensing reports shared by secondary nodes, which can then make a reliable joint decision on whether the channel is free or occupied. The protocol does not require the use of static trusted nodes, which are not found in usual CRN scenarios and are a dangerous single point of failure. It only requires some nodes (not necessarily the same ones all the time) to have access to a Validation Authority.

The protocol is based on the election of coordinator nodes which centralise the process of sharing the sensing results, authenticating these results and making the final decision. These coordinators are evaluated periodically by the other CRN nodes, in such a way that they can be replaced if their decisions start to differ from those of the majority.

As part of our future research, we plan to simulate the protocol and perform tests to verify that the amount of information transmitted over the network is not excessive.

Acknowledgements

This work is partially supported by the Spanish Ministry of Science and Innovation and the FEDER funds under the grants TSI-020100-2009-374 SAT2, TSI2007-65406-C03-02, TSI2007-65406-C03-03 E- AEGIS, CONSOLIDER CSD2007-00004 ARES, TIN2010-15764 and TIN2011-27076-C03-03. It is also supported by the European Commission under the grant FP7/2007-2013 Data without Boundaries (grant agreement number 262608)

References

[1] Federal Communications Commission. Spectrum policy task force report. Technical report, ET Docket No. 02-135, 2002.

[2] J. Mitola III and G.Q. Maguire Jr. Cognitive radio: making software radios more personal. *IEEE personal communications*, 6(4):13–18, 1999.

[3] S.M. Mishra, A. Sahai, and R.W. Brodersen. Cooperative sensing among cognitive radios. In *IEEE International Conference on Communications*, pages 1658–1663. IEEE Computer Society, 2006.

[4] H. Rifà-Pous, M. Jiménez Blasco, and C. Garrigues. Review of Robust Cooperative Spectrum Sensing Techniques for Cognitive Radio Networks. *Wireless Personal Communications*, pages 1–24, 2011. 10.1007/s11277-011-0372-x.

[5] P. Kaligineedi, M. Khabbazian, and V.K. Bhargava. Secure cooperative sensing techniques for cognitive radio systems. In *IEEE International Conference on Communications (ICC)*, pages 3406–3410, May 2008.

[6] R. Chen, J. Park, Y.T. Hou, and J.H. Reed. Toward secure distributed spectrum sensing in cognitive radio networks. *Communications Magazine, IEEE*, 46(4):50–55, April 2008.

[7] G. Jakimoski and K. P. Subbalakshmi. Towards secure spectrum decision. In *IEEE International Conference on Communications*, pages 2759–2763, Piscataway, NJ, USA, 2009. IEEE Press.

[8] Y. Xiao, V.K. Rayi, B. Sun, X. Du, F. Hu, and M. Galloway. A survey of key management schemes in wireless sensor networks. *Computer Communications*, 30(11-12):2314 – 2341, 2007. Special issue on security on wireless ad hoc and sensor networks.

[9] L. Lamport. Password authentication with insecure communication. *Commun. ACM*, 24(11):770–772, 1981.

[10] H. Rifà-Pous and J. Herrera-Joancomartí. A fair and secure cluster formation process for ad hoc networks. *Wireless Personal Communications*, 56(3):625–636, February 2011.

[11] R. Chen, J.-M. Park, and K. Bian. Robust distributed spectrum sensing in cognitive radio networks. In *INFOCOM. The 27th Conference on Computer Communications*, pages 1876–1884. IEEE Computer Society, 2008.

[12] Y. Sella. On the computation-storage trade-offs of hash chain traversal. In *Financial Cryptography*, volume 2742 of *LNCS*, pages 270–285, 2003.

[13] M. Fischlin. Fast verification of hash chains. In *The Cryptographers' Track at the RSA Conference (CT-RSA)*, volume 2964 of *LNCS*, pages 339–352, 2004.