On Device Authentication in Wireless Networks: Present Issues and Future Challenges

Georgios Kambourakis and Stefanos Gritzalis

Laboratory of Information and Communication Systems Security
Department of Information and Communication Systems Engineering
University of the Aegean, Karlovassi, GR-83200 Samos, Greece
{gkamb,sgritz}@aegean.gr

Abstract. Whilst device authentication must be considered as a cardinal security issue, complementary and of equal importance to user authentication, in today's wireless networks, only a few papers address it patchily. This paper identifies and analyses possible major solutions towards solving the device authentication problem. We discuss key issues and future challenges that characterize each solution examining its pros and cons. We also offer a short qualitative comparative analysis for the device authentication schemes in question, examining its applicability for both infrastructure and ad-hoc deployments.

Keywords: Device authentication, 802.1X, TCG, Wireless security.

1 Introduction and Problem Statement

Today, networks face many security risks, whether wired or wireless. One of the most common is unauthorized network access by an unknown device that connects to a network. From the one hand, wired devices like routers and switches are considered to be "locked-in-the-rack" and therefore under the supervision of an administrator. In contrast, one of the most important problems in today's deployment of infrastructure IEEE 802.11 wireless LANs, Mobile Ad-hoc Networks (MANET) and Wireless Sensor Networks (WSN) is that of the rogue device problem. In many installations, anything plugged in is given access to the network. Devices can almost immediately begin broadcasting data and reading information, regardless of what they are or come from. These systems can be difficult to scan, patch, or control. Furthermore, an unauthorized device is difficult to identify, locate and repel, when on the move, in an emergency situation. For instance, this refers to the situation in which an insider connects an unauthorized IEEE 802.11 device, say an Access Point (AP), to the corporate LAN, thus creating a security hole in the company network. Whether this sort of attack is most common to infrastructure IEEE 802.11 networks, similar problems may easily arise to MANETs, WSNs and even Radio Frequency Identification (RFID) tags where a rogue or even a compromised or cloned device can be fatal for the overall network trustworthiness. For instance, at present, most RFID devices promiscuously broadcast their static identifier with no explicit authentication procedure. This gives the opportunity to attackers to passively scan identifying data

C. Lambrinoudakis, G. Pernul, A M. Tjoa (Eds.): TrustBus 2007, LNCS 4657, pp. 135–144, 2007.

performing a *skimming attack*. Additionally, skimmed data may be used to fabricate cloned tags, thus giving more opportunities to attackers. In a *swapping attack*, for example, the adversary fabricates cloned tags, seals them inside a decoy container and quickly swaps the fake container with the original. Having the ability to clone a tag and prepare the decoy in advance, the adversary is able to carry out the physical swap very quickly. Furthermore, it is well known that erratic behaviours in sensors networks seeking physical access to sensor devices are difficult to be repelled due to the anonymous and (semi)uncontrolled terrain in most cases. At best, physical access to a certain sensor enables the aggressor to obtain sensor's secret keys. According to [1] a competent attacker equipped with a laptop is able to retrieve sensor keys in less than a minute given that he/she has physical access to it. Once these keys are compromised the attacker has access to the communications of the whole network.

In all cases, the heart of the problem is the lack of any mutual device-to-device authentication procedure or mechanism when a certain device attempts to join the network. Also, there are many cases where an identified device may not be allowed on the network; for example, if it was reported as stolen, the metadata in the device identity or policy store would indicate that it should not be allowed. Device authentication mechanisms enable an organization to manage both users and devices, thus it is considered as a second layer of authentication, ensuring that only specific authorized devices operated by authorized users can access the organization's network. Separately, neither one can have access. This means that even in case passwords, credentials or tokens are stolen or compromised, the network will still be well protected as long as the authorized device is not employed. It also assures that private data stored across network resources are never exposed because unauthorized devices cannot access the network, even when operated by an authorized person. Moreover, in case of infrastructure devices (e.g. Access Points, switches, etc) and other hardware that is not operated directly by humans, (like sensors) device authentication can guarantee to a great deal that a device is genuine and has not been somehow compromised. Therefore, device authentication effectively enforces network access control policies in a proactive manner, that is, before they connect to the network.

Currently, the most usual practice to protect against unauthorized access is to perform device identification by maintaining a list of MAC addresses that are allowed to access the network. However, today, this solution is considered ineffective as the majority of end-user devices allow the user to configure its MAC address at will. As a result, an insider can modify the MAC address of his rogue AP to match an existing authorized device and connect to the network without detection. In this paper we survey all major potential solutions and trends to the device authentication issue and examine its pros and cons. Each option is further analyzed and compared with the others based on some indicative qualitative criteria giving a comprehensive view about its applicability and robustness in terms of security. The remainder of the paper is structured as follows: the next section identifies and analyses possible solutions to the device authentication problem so far. Section 3 gives a qualitative analysis for the device authentication schemes in question. Finally, Section 4 offers some concluding thoughts and future directions of this work.

2 Identification of Possible Solutions

2.1 The IEEE 802.1X Framework

With the advent of the IEEE 802.11i specification [2] the 802.1X [3] framework provides various Extensible Authentication Protocol (EAP)-based and certificate-oriented mechanisms that can be employed both for user as well as for device authentication. Towards this direction every device must afford a device certificate bound to it to be able to prove its identity prior acquiring an IP address and joining the network. The uniqueness of each network device can be determined by a combination of its hardware and software characteristics. For example, hardware parameters may be the device's serial number, hard disk or other components serial codes and manufacturer identities, MAC address, processor type, memory capacity, etc, while as software parameters may use a hash of some driver codes, start/end memory address of software portions stored in ROM and other similar attributes. A careful choice of this kind of characteristics is enough to uniquely identify each network device even those of the same model and type. Note however that these attributes must be static in the long run as they comprise the identity of each particular device.

Once a collection of such parameters has been decided, e.g. by the network operator, a hash of the concatenated sequence (charact_1‖ charac_2 ‖…‖ charact_n) is calculated to serve as the mid or long-term identity of the device. As a result, a device certificate must bind a combination, say a hash of various physical properties of the device (MAC address, serial number, driver versions, etc), to a private key in the form of a X.509 certificate. After that, device-to-device authentication can be effectively exercised utilizing EAP methods (EAP-TLS, EAP-TTLS, PEAP, etc), before any user authentication takes place. It is stressed that the private key of the device must be stored securely in the device in the form of a tamper resistant memory, therefore not accessible by human users or applications. By this scheme, the authentication server can utilize the same identity certificate that is always used when being authenticated by other network nodes.

However, at least for IEEE 802.11 infrastructure mode, 802.1X-based device authentication mandates several modifications concerning the current communication procedures between the AP and the authentication (usually RADIUS) server. Specifically, all APs must act as supplicants when booting-up (before acquiring an IP address) to be able to be authenticated as devices to the corresponding RADIUS server. Moreover, all network devices, including APs, must support e.g. EAP-TLS protocol functionality to support certificate based authentication at the data link layer. In addition, a well-defined and scalable (re)keying mechanism between the AP and the authentication server to encrypt the traffic between them must be somehow automated and not rely on administrators to configure it manually. This is especially true for remote network devices. Currently however, no standard automated session key derivation procedure between an AP and the authentication server exists. Furthermore, to thwart clever attackers any solution applied must support periodic re-authentication at regular intervals, thus ensuring session freshness. Additionally, periodic session validation may presume the derivation of a session key between the involved devices during initial device authentication phase. After that, it is not possible to substitute a legitimate device, since the rogue one does not know the

current session parameters, including the key. Apart from all previously discussed issues the 802.1X approach: (a) cannot straightforwardly be accommodated to ad-hoc network configurations as it requires infrastructure mode, (b) mandates some sort of Public Key Infrastructure (PKI) and some rather sophisticated and maybe costly hardware and software components to be implemented, (c) in most cases requires expensive public key operations and protocols, that lightweight mobile devices is difficult to afford. Therefore, it is only appropriate for medium to large organizations rather than for Small Office/Home Office (SOHO) environments, MANETs, or WSNs. Concluding this subsection, we can say that 802.1X-oriented device authentication, if refined and standardized sometime in the future, can provide a promising avenue towards solving the device authentication problem.

2.2 The IEEE 802.16 Case

When Device authentication through corresponding device (manufacturer) certificates is already part of the IEEE 802.16 standard, namely the Privacy Key Management (PKM) protocol [4]. The PKM RSA authentication protocol employs X.509 digital certificates and the RSA public key encryption algorithm that binds public RSA encryption keys to MAC addresses of MSs. Under this context, a Base Station (BS) authenticates a client Mobile Station (MS) during the initial authorization exchange. Each MS must incorporate a unique X.509 digital certificate issued by the MS's manufacturer. The digital certificate among other contains the MS's Public Key and serial number and the MS's MAC address. When requesting an Authorization Key (AK), an MS presents its X.509 certificate to the BS. Upon reception, the BS verifies the MS's certificate, and then uses the public key that it contains to encrypt an AK, which then sends back to the corresponding MS. Under this scheme MAC spoofing attacks can be effectively repelled considering that only the legitimate MS device has the matching private key to decrypt AK and join the network. Briefly, the specification mandates that all MSs using RSA authentication shall have factory-installed RSA private/public key pairs or provide an internal algorithm to generate such key pairs dynamically. All MSs with factory-installed RSA key pairs shall also have factory-installed X.509 certificates. All MSs that rely on internal algorithms to generate an RSA key pair must offer a mechanism for installing a manufacturer-issued X.509 certificate after key generation. For mutual authentication each BS is also equipped with a digital certificate that binds its hardware characteristics with the corresponding public key as described in [4].

Note that the newest PKM version 2 protocol specification [4] supports 802.1X/EAP authentication too. This is of course a movement towards providing a unified 802.11/802.16 authentication framework, but in our case device authentication services to heterogeneous 802.11/802.16 contexts may also be applied as discussed earlier in the previous subsection. Generally, the PKM's authentication protocol establishes a shared secret (AK) between the MS and the BS. The shared secret is then used to secure subsequent PKM exchanges of temporary keys. PKM also supports periodic re-authentication / re-authorization and key refresh. Although, the 802.16 approach is effective as far as the device authentication problem is concerned, it suffers from the same problems discussed in Section 2.1.

2.3 The Trusted Computing Solution

A different hardware oriented solution towards solving the device authentication problem has been examined in the means of trusted computing. Considering this option a number of hardware and software manufacturers have cooperated forming the non-profit Trusted Computing Group (TCG). The main aim of TCG is to develop trusted platforms by utilizing Trusted Platform Module (TPM) chips and novel hardware architectures. The TPM chip [5], also referred to as the "Fritz chip", is responsible for a number of basic functions including integrity measurement, integrity storage and integrity reporting of all critical events occurring in the trusted platform. This chip can be either embedded in a smartcard or dongle soldered onto the motherboard or will be integrated in the main processor. The latter approach offers better security because the data is not transferred on motherboard buses between the TPM and the CPU. Very recently [6], TCG formed the Mobile Phone Work Group focusing on the adoption of TCG concepts for mobile devices. This work group will enhance TCG as needed to address specific features of mobile devices like their connectivity and limited capability.

The specification defined by the TCG [7] states that Trusted Platforms (TPs) are computing platforms that add to themselves the property of trust. In other words, they provide proper mechanisms to verify, in a secure way, that the data yielded by them is not tampered with. When a manipulation is performed, a security discrepancy is detected and reported to the user who will decide whether or not to trust the data provided by the TPs. More specifically, on booting up, the TPM takes over inspecting the integrity of boot ROM, then loading and executing it, and finally, verifying the overall system's state. It then verifies the first portion of the operating system, loads and executes it, and again attests the system's state. This procedure repeats several times for all protected software modules which in the end are loaded and become available to the system upon booting up. Moreover, the TCG-enabled system preserves and maintains a list of approved hardware and software components. For each of them, the system must confirm whether it is approved and not revoked and whether it is digitally signed in case of software. Meanwhile, e.g. in case that some components have been upgraded and therefore the system's configuration has changed, it must go online to be recertified. In this context, trusted computing can contribute a great deal to the vision of the "self authenticated, self protecting network" where every wireless or wired network entity that contains a TPM is self and cross authenticated before entering the network. As a result, rogue components either hardware or software can be repelled from joining the network. Nevertheless, currently the level of security provided by TPM modules highly depends on the details of design and implementation, which are not clear yet for almost all trusted computing manufacturers. Moreover, the TCG specifications has to cover some distance until it reaches a mature state and proved to be secure and trustworthy enough (not simply trusted) in the long run [8,9,12].

2.4 Other Approaches

In this subsection we shall briefly survey other research works dealing diametrically or partly with device authentication.

In NIST report 7206 [10] the authors employ smart cards to support user and mobile devices authentication. They state that smart card authentication is perhaps the best-known example of a proof by possession mechanism when compared to other more traditional categories of authentication, including proof by knowledge (e.g. passwords) and proof by property (e.g. fingerprints). Towards this direction the report provides an overview of two novel types of smart card that use standard interfaces supported by most handheld devices. Without doubt, when used for user authentication, smart cards can improve the security of a device and provide additional security services too. Device authentication can also be seconded considering that it is generally more difficult to operate a rogue (compromised or stolen legitimate) device without the proper smart card. On the other hand, cloning an existing device and its matching smart card is not exactly an easy task for the attacker to accomplish. On the contrary though, standard size smart cards are generally not suitable for handheld devices due to the relatively large size of the card, the need for a proper card reader, and the difficulty and cumbersomeness of embedding a reader to the device. Putting aside these obstacles, by e.g. utilizing interfaces found today in most smart card readers (as in the aforementioned report), smart card authentication may prove very profitable. Some difficulties remain however including the increased acquisition and administrative cost for the users and the organizations themselves and the fact that this solution is not suitable for small wireless devices like sensors and RFIDs.

In another work [11] that partly deals with device authentication the authors examine location-based access control mechanisms. They propose a new protocol for location verification, called the Echo protocol and they prove its security. Location verification enables location-based access control. This means that a person carrying a specific device can be granted access to particular resources only if his/her location has been confirmed by employing a corresponding protocol. Naturally, when this approach is combined with physical security e.g. who's entering the building, then location verification can be used to allow wireless access to all those inside. It is true that location-based access control has several pros. Among others, it is natural for various applications. While one simple security policy might permit wireless access of only the printers installed in the office you are in, on the other hand might force that a wireless device must cease operating if it is detected operating outside the company building or being moved to another room. By this means, stolen, compromised or rogue devices not operated in certain premises, where they are supposed to operate according to the current policy, will be proved useless to malevolent individuals. Though, while location-based access control in human terms is straightforward, e.g. turning on the TV set in a particular room needs to have a physical presence in the room, achieving the same kind of guarantee with wireless networks, is not so easy. Location-based access control policies on networks and information resources by extension, requires a method to perform location verification, where an entity's location is securely verified to meet certain criteria: e.g. being inside a particular room. In practice, while this approach may be effective if implemented properly (guarantee in-region verification for a high rate of legitimate location claims), requires significant administrative costs in terms of configuring and maintaining proper and strict policies for every network entity involved. On the top of that, as with 802.1X, location-based access control adapts better with infrastructure

wireless networks having some sort of administrative authority to define policies rather than ad-hoc pervasive mode and nomadic computing.

A different approach that examines the feasibility of identifying wireless nodes in a network by measuring distinctive electromagnetic characteristics or "signatures" of Wireless Local Area Network (WLAN) cards is presented in [13,14]. There the authors focus and perform preliminary experiments with IEEE 802.11 compatible cards but their conclusions can be applicable to other wireless technologies as well. Their idea originates from the remark that the physical layer of 802.11 wireless communications cannot effectively protect the identities of the communication endpoints. Specifically, any electromagnetic signal transmitted over the air can be passively or actively monitored, captured and analyzed at will by any properly equipped adversary located within the wireless device's transmission range. This physical layer "vulnerability" is also under investigation by several researchers in the context of the so called template attacks. Therefore, users' anonymity and privacy can be in danger if their device can be uniquely identified, through the measurement of distinctive radio-frequency electrical characteristics or electromagnetic signatures that it emits. The attacker's aim in this case is to correctly relate a received electromagnetic emission with a specific transmitter (device). At frequencies, such as 2.4 GHz or 5.2 GHz, used in 802.11 networks even minor component variations in a transmitting circuit may result to a significant effect on the emitted signal. Given that we are able to detect and record distinctive electromagnetic signatures, a wireless device and its user can not only be monitored, but when combined with visual identification, can also be identified. Due to these qualities, devices' electromagnetic emissions are worth being further investigated in the context of effective device identification / authentication. Rogue, compromised and even cloned devices can be differentiated from the legitimate ones through their electromagnetic signature that they emit. However, this must be proven so, not only in sporadic experiments, but also in large scale, where many types and access technologies of wireless devices are employed. On the other hand, device authentication based on this scheme may be practical in corporate networks - by constructing beforehand a database of all authorized devices' electromagnetic signatures (metadata describing the asset) and putting it in a corresponding authentication server - but seems rather unpractical for ad-hoc deployments.

The last one but lightweight category of solutions has been proposed in [15] and redefined later in under a three party (proxy assisted) setting[1]. The authors analyze a particular human-to-computer authentication protocol designed by Hopper and Blum (HB), and demonstrate by using RFID tags that it is practical for authenticating low-cost pervasive devices as well. The outcome of their work is a new symmetric authentication protocol, namely HB+ that is appropriate to securely identify and authenticate wireless devices with limited power and processing capabilities. The motivation here is that low-end RFID tags and other similar pervasive devices share many limitations with human beings. For instance, just like people, RFID tags can neither remember long passwords nor keep long calculations in their working memory. In this context, well-studied human authentication and identification

[1] We selected these works among others in the literature [17] as the most representative for low-end, low-cost wireless devices.

protocols utilized for proving human's identity to a machine, can also be applied in low-cost wireless devices. It is true that securing low-end wireless devices is a challenging issue because of their limited resources and small physical form. Towards this direction the HB+ and other analogous protocols [17] can contribute to the problem of secure device authentication. Nevertheless, while theoretically the HB+ protocol is secure against both passive and active aggressors and should be realizable for implementation in current RFID tags, a number of open questions remain before the HB+ can see practical realization [15]. Moreover, do not neglect that HB+ and alike protocols proposed both for RFIDs and sensors devices lean against symmetric secrets stored inside the device, which in turn can be entirely revealed through active or physical attacks, such as electron microscope probing as discussed in [16].

3 Discussion

Currently, there exist several software-based ways to safeguard mobile devices Virtual Private Networks (VPNs), firewalls, upper layer data encryption software, device management solutions, to name just a few. These types of solutions typically protect the data and or operating systems of the devices from attacks, but cannot guarantee the integrity and authenticity of the hardware platform on which they are running. For example, while SIM or UICC are employed in the wireless cellular networks to authenticate users, they cannot ensure the computing platform on the mobile equipment is trustworthy too. Also, many applications of cryptographic identification protocols are vulnerable against adversaries who perform real time active attacks. For instance, when identifying a physical device like a wireless AP, common identification schemes can be by-passed by faithfully relaying all messages between the communicating participants. This attack is well known in the literature as mafia fraud. Furthermore, this sort of solutions does not contribute much in protecting the unique identity of a handheld device such as a mobile phone. When intercepted, these identities can be further utilised to install rogue network components in absence of effective access control mechanisms. However, device authentication is a hard problem to deal with, as it involves some sort of bootstrapping trust between the access control mechanism and the stranger device or between several stranger devices in ad-hoc mode. This becomes even more complicated considering (a) the heterogeneity of the wireless access technologies that currently exist and (b) the diversity of network providers reflected in their security policies. In the previous section we investigated several device authentication schemes and discussed its pros and cons. Generally, schemes based on symmetric cryptography have obvious performance advantages over public-key cryptography; they fit much better to low-end wireless devices and ad-hoc modes, but usually suffer from complex key management. They also mandate some sort of trust in the entire network as a device moves from one wireless domain to another. Admittedly, schemes based on public-key technology offer less computation for more communication rounds, but are still too costly to be practical for at least non-infrastructure wireless networks that involve low-power computing devices.

Table 1. Device authentication schemes comparison (Mod.=Moderate, P=Partly, NA=Not Applicable, S=Symmetric, A= Asymmetric, Inf.=Infrastructure)

Scheme Description	Inf. /Ad-Hoc	S/A key	Effectiveness/ Robustness	Scal-ability	Prac-ticability	Heterog. Env.
IEEE 802.1X	Inf.	A	High	High	Mod.	Partly
IEEE 802.16	Inf.	A	High	High	Mod.	Partly
Trusted Comp.	Mainly Inf.	Both	High	High	Mod.	Mostly
Smart Cards	Mainly Inf.	Both	High	Mod.	Fair	Partly
Location-based Access Control	Mainly Inf.	NA	Mod.	Mod.	Mod.	Partly
Electromagnetic Signatures	Mainly Inf.	NA	Mod.	Mod.	Mod.	Partly
HB+ and other similar protocols	Both	S	Mod.	Fair	High	Partly

Table 1 depicts an aggregate comparative view of all the anticipated schemes considering six basic criteria: (a) supports infrastructure and/or ad-hoc deployments, that is, centralized and/or distributed, (b) requires symmetric or/and asymmetric key technology, (c) effectiveness and robustness in terms of security, (d) scalability, (e) practicability to implement, (f) supports heterogeneity in terms of access technologies and trust relations between network providers. As a general remark it seems that the trade-offs between security robustness and lightness in terms of processing power and accompanying infrastructures and between ad-hoc and infrastructure modes are not easy to fulfil. More specifically, the trusted computing approach and the 802.1X authentication framework seem to be the most promising solutions towards solving the device authentication problem. On the downside, these options are rather impractical for nomadic users and ad-hoc deployments, due to the PKI and Authorization, Authentication, Accounting (AAA) entities that they mandate and the associated cost that goes with them. The IEEE 802.16 solution although based on 802.1X principles is more or less custom-tailored to Wi-Max networks. All the other approaches are very interesting still, they have to prove their effectiveness in terms of security robustness, scalability, key administration and ease of materialisation. In our opinion one global universal solution is at present difficult to form. It is better to orientate ourselves in choosing one of the aforementioned schemes, according to our particular needs and interest or alternatively develop a custom-made hybrid solution.

4 Conclusions and Future Work

In this work we define device authentication (or identification) as the entity authentication in which the objective is to identify and further authenticate a physical device possibly at a specific location. In this paper a constructive analysis of the current potential solutions and trends to the device authentication issue have been given. Each scheme was briefly presented and some comments including implementation problems and research challenges have been provided. Finally, a comparison of the schemes was conducted based on several criteria. As a statement of direction, we are currently working on expanding this work by proposing a new optimized hybrid device

authentication method, which exploits the advantages of the presented mechanisms, while at the same time minimizes the drawbacks pointed out throughout this paper. Another important issue worthy of investigation is how to preserve privacy, that is, logically disassociate the user from the device that they operate; in other words how to correctly identify a device without disclosing user's private information, thus preserving anonymity, context privacy, location identity, etc.

References

1. Hartung, C., Balasalle, J., Han, R.: Node Compromise in Sensor Networks: The Need for Secure Systems, T.R. CU-CS-990-05, Department of C.S. Univ. of Colorado (January 2005)
2. IEEE Std. 802.11i-2004, Amendment to IEEE Std. 802.11, 1999 Edition, Amendment 6: Medium Access Control (MAC) Security Enhancements, Part 11, IEEE Press, Los Alamitos (June 2004)
3. IEEE 802.1X-2004 IEEE Standards for Local and metropolitan area networks - Port-Based Network Access Control (December 2004)
4. IEEE P802.16e/Draft12, IEEE Standard for Local and metropolitan area networks, Amendment for Physical and Medium Access Control Layers (published October 2005)
5. TCG, TPM Main Part 1 Design Principles Spec. Version 1.2 Revision 85 (February 2005)
6. TCG Mobile Trusted Module Spec., version 0.9, Revision 1, DRAFT (September 2006)
7. TCG, Spec. Architecture Overview, Specification Revision 1.2 (April 2004)
8. Bruschi, D., Cavallaro, L., Lanzi, A., Monga, M.: Attacking a Trusted Computing Platform, Improving the Security of the TCG Specification, Tech. Report RT (June 2005)
9. Hendricks, J., van Doorn, L.: Secure Bootstrap is Not Enough: Shoring up the Trusted Computing Base. In: Proc. of the 11th ACM SIGOPS, September 2004, ACM, New York (2004)
10. Jansen, W., Gavrila, S., Séveillac, C., Korolev, V.: Smart Cards and Mobile Device Authentication: An Overview and Implementation, NIST, NISTIR 7206 (July 2005)
11. Sastry, N., Shankar, U., Wagner, D.: Secure Verification of Location Claims. In: ACM WiSE'03, September 19, 2003, California, USA, pp. 1–10 (2003)
12. Zheng, Y., He, D., Yu, W., Tang, X.: Trusted Computing-Based Security Architecture For 4G Mobile Networks. In: Proc. of PDCAT '05, pp. 251–255 (2005)
13. Remley, K., et al.: Electromagnetic Signatures of WLAN Cards and Network Security. In: IEEE Int'l Symposium on Signal Processing and Information Technology, pp. 484–488 (2005)
14. Henrici, D., Muller, P.: Hash-based Enhancement of Location Privacy for Radio-Frequency Identification Devices using Varying Ident. In: IEEE PerCom, pp. 149–153, 04
15. Juels, A., Weis, A.: Authenticating Pervasive Devices with Human Protocols. In: Boyd, C. (ed.) ASIACRYPT 2001. LNCS, vol. 2248, pp. 149–153. Springer, Heidelberg (2001)
16. Anderson, R., Kuhn, M.: Low Cost Attacks on Tamper Resistant Devices. In: Christianson, B., Lomas, M. (eds.) Security Protocols. LNCS, vol. 1361, pp. 125–136. Springer, Heidelberg (1997)
17. Wen, H.-A., et al.: Provably secure authenticated key exchange protocols for low power computing clients. Computers & Security 25, 106–113 (2006)