Erwin Aitenbichler · Andreas Heinemann

Proximity-Based Authentication for Windows Domains

Received: June 08, 2007 / Accepted: June 30, 2007

Abstract Over the past few years, several authentication methods based on location-limited channels have been presented in research literature. We extend this notion to location-aware, zero-interaction authentication, present an efficient protocol implementation, and describe the integration of our authentication system into a state-of-the-art enterprise network solution.

1 Introduction

Ubiquitous Computing means that we will be surrounded by many computing devices at our future workplaces and that computers will increasingly be used in a freely moving context. To date, switching between computers requires users to perform manual authentication procedures. With more computers, this task will become more and more tedious. Hence, there is a demand for novel authentication concepts which do not require manual user interactions and still provide a good security. In this paper we describe a system for zero-interaction authentication for Microsoft Windows Server domains.

In our system, users are equipped with personal devices, i.e., small wearable computers. Regarding the location of a user in relation to a computer, we distinguish between three zones (Fig. 1). Once the user approaches the computer in the environment and gets into line-of-sight distance, she is automatically authenticated. When she moves out of line-of-sight distance, but stays in proximity, the computer is locked. Thus, it remains reserved for the user. When the user moves out of the proximity zone, she is automatically logged out. Thus, the device is freed and can now be accessed by other users.

Erwin Aitenbichler, Andreas Heinemann
Telecooperation Group
Department of Computer Science
Technical University Darmstadt
Hochschulstr. 10
D-64289 Darmstadt, Germany
Tel.: +49-6151-16-4557
Fax.: +49-6151-16-3052
E-mail: (erwin,aheine)@tk.informatik.tu-darmstadt.de

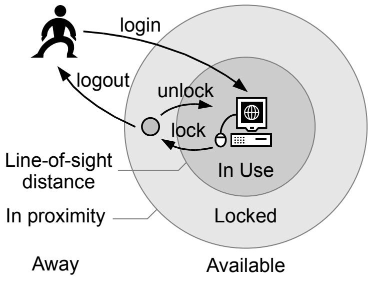


Fig. 1 Relation of user location and computer state

Our system mainly targets mobile work scenarios, where the worker requires her hands and eyes free to perform the task at hand. She interacts with the computer system through a wearable computer using speech. For example, we investigated the use of such a system in a car repair shop [17]. However, a pure speech-based interface may be inefficient when the user needs to browse drawings or large documents. In such a case, a nearby wall display, that is controlled by speech commands, may serve as an additional device for visual information processing.

In this paper we describe a system for the automatic authentication between the user's wearable and arbitrary devices in the environment. This is more convenient for the user compared to typing in passwords, etc. and under certain circumstances, users may not be able to use their hands for interaction. For example, consider a surgeon who needs to keep his hands clean, a mechanic with oil-smudged hands, or a firefighter wearing thick gloves.

In such mobile work scenarios, the wearable computer is required for speech-based interaction anyway. Thus, integrating an authentication system in addition does not put any additional burden on the user. The authors believe that this approach will also be useful for everyday use in offices or homes, as soon as the necessary hardware can be built sufficiently small and is less cumbersome to wear than the current prototype.

A Windows domain provides the backbone infrastructure of our authentication solution. Active Directory allows the central management of users, computers, and access policies in an enterprise. In addition, we use the Public Key Infrastructure (PKI) functionality to implement certificate-based secure authentication. In such an environment, a user can easily access all resources in the same domain, given that she owns the necessary access rights. Because of the described infrastructure requirements, our system mainly targets corporate settings.

The contribution of this paper is threefold. First, we will introduce the notion of a user's personal device, called Minimal Entity (ME). Second, we present an efficient authentication protocol for out-of-band channels. Because out-of-band channels often only support low bitrates, a compact protocol lowers delays, thereby facilitating a fluent interaction. Third, our prototype implementation has been fully integrated with Microsoft Windows Server domains and the industry-strength authentication framework Mica [18] by usd.de ag [1].

This paper is organized as follows. We will first describe related work in section 2. The user's personal device is introduced in section 3. Next, we will present our authentication protocol in section 4. In section 5 we describe the implementation of our system. Finally, the paper is concluded in section 6.

2 Related Work

Our work uses infrared as a *location-limited channel* [5] in order to bootstrap a secure communication between the ME device and a Windows PC. This out-of-band communication reduces the risk of attacks. A similar usage of infrared is described by Spahic et. al. [19]. The authors describe an IrDA-based solution that can be used to point to a device for establishing a secure connection. For that, the first phase uses an infrared link for communication. Two devices agree on a key by using the Diffie-Hellman algorithm. In the second phase, the devices switch over to a radio-based communication that is secured by the generated key. A similar approach makes use of a laser beam [12] to exchange a key. This method allows for more precision over a longer distance.

The Diffie-Hellman algorithm allows to establish a secure communication link between *any* two devices. In contrast to that, we want to ensure that users only pass login credentials and sensitive information to trusted devices and that devices can only be accessed by trusted users. The Windows domain allows the central management of all those trust relationships and provides a PKI. Therefore, we secure communication based on asymmetric cryptography, instead of using Diffie-Hellman, which would add an unnecessary additional layer of encryption.

In addition, we combine the out-of-band IR link with a location tracking system in order to establish different zones around a device: *authenticated and in use*, *authenticated and locked*, and *available*. The ME is a wearable computer that operates autonomously. This gives us two advantages: First, the ME's infrared transceiver is mounted on a headset worn by the user. This leaves the user's hands and eyes free for operating keyboards, screens, or any kind of physical work. Second, the ME can automatically make decisions based on policies. Thus, the user does not have to perform any manual steps during authentication procedures in the regular case.

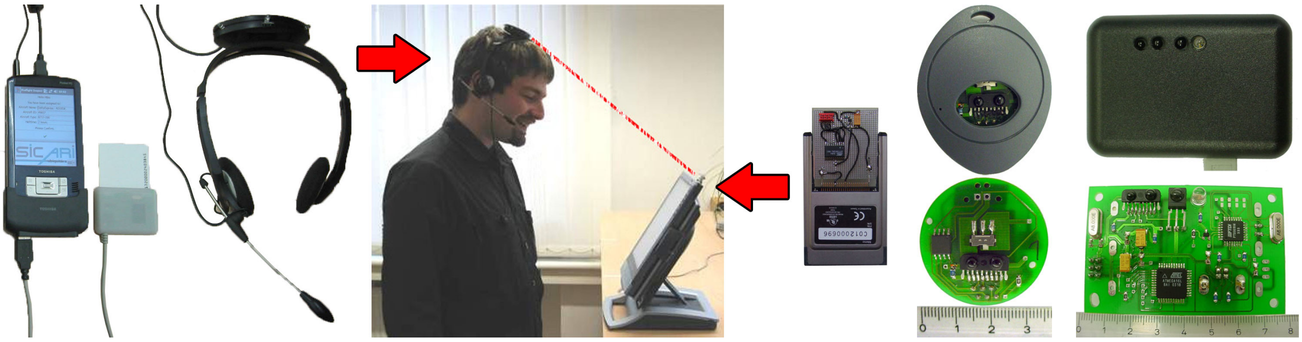
Besides infrared, there are other suitable out-of-band channels, for example, dynamically generated *2D barcodes* [8,14], *audio* [10], *biometric data* [6], and *ultrasound* [7,13]. For a detailed comparison of these out-of-band channels with respect to their security, convenience, and hardware requirements see [14].

3 Hardware: The User's Personal Device

The Mundo project [2] is concerned with general architectures, middleware, common services, and software development processes for ubiquitous computing. The central element of the device architecture is the user's personal device, called Minimal Entity, or short ME.

ME devices are considered as the representation of their users in the digital world. The personal ME is the only entity always involved in the user's activities. It is a small wearable computer with a "minimal" functionality. However, to still be able to interact with ubiquitous computing environments in a sensible way, the term "minimal" is associated with a set of specific requirements regarding *size*, *identity*, *security*, *interaction*, *context awareness*, and *networking*. Users can augment their personal computing space through *association* with any number of so-called *User aSsociable* (US) devices. The user is expected to wear the ME device almost at all times, while all other entities are optional.

In terms of pure functionality, i.e., holding the user's digital identity, such a ME device could potentially be constructed in a very simple manner; for example based on an RFID chip or similar [9,20]. However, there are two significant problems when using solely RFID tags for user identification. First, users cannot enable or disable the tag. It should only work when it is worn by the legitimate user and not for a malicious user who has stolen the tag. Second, the tag does not decide in an intelligent way to whom the user's identity should be disclosed and to whom not. It reveals the user's identity to everyone and thereby compromises the user's privacy. Furthermore, tags can be read from great distances (20 meters) with suitable equipment [11]. Some RFIDs, like the ones integrated in paper passports, protect the information stored on the tag by requiring authentication by the reader. But then, such a tag is only readable by a certain system and cannot be used as a general-purpose authentication token. Even if RFIDs implement a content protection, they can in many cases be tracked, e.g., by observing their collision-avoidance behavior [4].



(a) Talking Assistant – User with TA and Tablet PC – PC Card Tag (b) Ovo Tag (c) Room Receiver

Fig. 2 Hardware

When the user puts on the device, the keychain stored on the ME has to be unlocked first by a personal identification procedure. For example, this could be implemented by means of a fingerprint sensor, voice and speaker recognition, or by entering a PIN. Once the user has successfully authenticated, the keychain can be decrypted. It stores various credentials used to authenticate with other parties based on username-password pairs, certificates, Mica credentials, etc. Finally, the device is able to detect with a sensor when the user takes it off again and locks the smartcard automatically.

The notion of requiring users to carry personal, trusted devices permits solving a number of important issues:

- The device helps the user to preserve her privacy by deciding in an intelligent way whether the identity should be disclosed to third parties or not.
- The ME device introduces a two-stage authentication mechanism. When the user puts the ME on, she has to authenticate once. Later on, the ME automatically authenticates the user with other computing devices in proximity. This scheme drastically increases user convenience, while preserving security and privacy.
- Because the user can trust her personal device, cryptographic keys can be stored on the device and it can be used as a secure terminal for secure transactions.
- The ME is the optimal place for hosting certain coordination services. It manages the user's personal computing environment by keeping track of associated devices, delivering credentials, and distributing encryption keys.

The Talking Assistant (TA) [3] is a prototype for a ME. The current version comprises three parts: a Windows CE based PDA, a headset with sensors, and a smartcard reader. The smartcard is used to protect the ME keychain. In the current prototype version, the user wears the headset on her head and wears the PDA on her belt. Because the headset only contains a few sensors, it is no more cumbersome to wear than a standard headset without any sensors. All the sensors are housed in the small box, attached to the top of the headset and connected to the PDA through a cable as shown in Fig. 2(a). The functionality of the device is as follows.

The TA supports audio streaming over the network, local speech recognition and synthesis. This functionality can be used to implement speech-based user interfaces.

A wear sensor mounted at the bottom of the sensor box on the headset measures the distance between headset and user's head, which allows to determine if the headset is worn.

In addition, the headset supports infrared communication with a *short-range* and a *long-range mode*. These two different modes are implemented by using specific combinations of hardware, modulations, and protocols.

3.1 Relative Positioning

The headset is also equipped with a *short-range* infrared transceiver. This interface has two uses. First, the TA can determine its location by receiving signals sent by IR tags without requiring any infrastructure. Second, the interface provides a location-limited channel that is well-suited for authentication protocols.

Tags are small IR emitters that can be attached to both fixed and mobile devices. We have built tags in various form factors. Fig. 2(a) shows a PC card tag that can be inserted into a Tablet PC or notebook. The tag shown in Fig. 2(b) can be attached to a stationary PC or monitor. Each tag has a unique tag identifier which it broadcasts periodically.

The communication range was intentionally limited to the space in front of the headset and a distance of about 1.5 meters, by picking the appropriate hardware components and modulation. The directional characteristic of the IR transceiver is shown in Fig. 3. Consequently, transmissions from tags can only be received by nearby receivers within direct line of sight and the reception of a tag ID conveys the context that the receiver is close to and facing the tagged object.

3.2 Absolute Positioning

Eight infrared emitter diodes cover all 360 degrees around the headset and are used for the *long range mode*. The TA sends out its ID at fixed intervals. These signals are

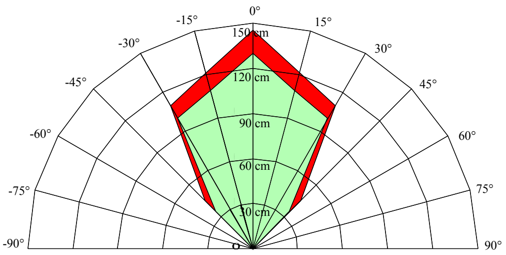


Fig. 3 Short Range IR Directional Characteristics

received by a so-called *room receiver* (Fig. 2(c)). Room receivers are firmly installed in a room, listen for TA broadcasts, and relay them to the local network.

This system implements a presence detection: When the signal of a certain TA is received, it must be in the same room. The range of the system is limited to about 10 meters and works with diffuse reflections as well, i.e., it does not require a direct line of sight between TA and receiver. This way it is possible to track users at room granularity.

The TA also supports 3D tracking and sensing the user's head orientation in all three rotational axes. However, these features are not used in the present application described here.

4 Authentication

The short range link used for the communication between tags and the TA can also be used as a *location-limited channel* for cryptographic authentication protocols. Ideally, this channel is only shared by the TA and the device to authenticate with.

We assume that two different wireless communication interfaces are used. The first interface is based on infrared and provides a location-limited channel for bootstrapping secure communications. The second is a conventional radio frequency-based interface used for the actual data transmission, because a radio link is more efficient and reliable than an infrared link.

The protocol for secure association establishes a secure wireless link between Minimal Entity (ME) and User aSsociable (US) device. The algorithm uses the following cryptographic operations.

$\mathbf{E}_K(M)$	Encrypt message M with key K.
$\mathbf{D}_K(M)$	Decrypt message M with key K.
$\mathbf{H}(M)$	One-way hash (or message digest) of M.
$\mathbf{S}(M)$	Signature for M: $\mathbf{S}(M) = \mathbf{E}_P(\mathbf{H}(M))$ with P being the private key of the signer.

The association process involves the following steps.

1. The US sends an *advertise message* to the ME over the location-limited channel. With the advertise message, an US indicates that it is available for association. The message contains the network address A_{US}, concatenated with the certificate C_{US} of the US, and a timestamp T. Finally, the whole message is signed using the US's private key.
   $A_{US}, C_{US}, T, \mathbf{S}_{US}(A_{US}, C_{US}, T)$
2. The ME verifies C_{US} and extracts the public key K_{US} from the US's certificate C_{US}.
3. The user is asked if she wants to associate her ME with the US device.
4. The ME sends an *association request message* to the US. First, the ME generates a random session key K_S for symmetric cryptography and encrypts it with the public key of the US. The ME's network address A_{ME}, its certificate C_{ME}, and the timestamp T provided by the US are encrypted with K_S. Finally, the whole message is signed using the ME's private key.
   $\mathbf{E}_{K_{US}}(K_S), \mathbf{E}_{K_S}(A_{ME}, C_{ME}, T), \mathbf{S}_{ME}(\mathbf{E}_{K_{US}}(K_S),$
   $\mathbf{E}_{K_S}(A_{ME}, C_{ME}, T))$
5. The US decrypts the session key using its private key P_{US}. Because nobody else has access to P_{US}, a possible eavesdropper cannot reconstruct K_S.
   $K_S = \mathbf{D}_{P_{US}}(\mathbf{E}_{K_{US}}(K_S))$
6. The US verifies the timestamp T. It must closely match the value of T sent in step 1.
7. The US verifies C_{ME} and extracts the ME's public key K_{ME}.
8. The US verifies the signature of the message to make sure that the sender of the message has knowledge of the ME's private key P_{ME}. This step is performed to circumvent *man-in-the-middle* attacks. The US verifies that
   $\mathbf{H}(M) = \mathbf{D}_{K_{ME}}(\mathbf{S}_{ME}(M))$
   with
   $M = \mathbf{E}_{K_{US}}(K_S), \mathbf{E}_{K_S}(A_{ME}, C_{ME}, T)$
9. All communication between ME and US can now be encrypted using K_S. The US sends a reply message to the ME indicating that the association was successful. The association can be terminated explicitly by sending a disassociate request or the duration can be controlled by means of leases.

An issue of this protocol is that the advertise message sent over the IR interface is relatively long (about 680 bytes), because it contains the certificate of the US. The amount of data that needs to be passed over the location-limited channel can be reduced to about 40 bytes by adding the following two steps at the beginning:

1. In the first step, only the address of the US and the digest of the advertise message are passed over the restricted channel.
   $A_{US}, H(A_{US}, C_{US}, T)$
2. The ME sends a request for the full advertise message to the US, along with its address.
   $A_{ME}, H(A_{US}, C_{US}, T)$
3. The US verifies that the received digest matches the actual digest of the advertise message. Thus, the US only reveals its identity if the correct digest is provided. Next, the US sends the full advertisement message to the ME over the shared channel.

Using a digest of the advertise message allows the ME to verify that it is really communicating with the correct US. A malicious US listening on the shared channel could simply answer all association requests. The properties of the described protocol are very similar to the two-way protocol of the ISO Authentication Framework [16].

5 Software Implementation

The software architecture is shown in Fig. 4.

Windows Server Domains use the Kerberos V5 protocol for mutual authentication between clients, such as users, computers, or services, and servers [16,15]. Active Directory (AD) allows the central management of users and devices in an enterprise and with its replication and caching mechanisms, it offers a scalable and efficient solution for authentication and authorization in large networks. Virtually all services offered by Windows servers support Kerberos authentication. Because Microsoft's implementation supports the IETF standard RFC 1510 and Active Directory can be accessed via LDAP, also many non-Windows-based services can use domains for authentication.

Kerberos is based on the assumption that initial transactions between clients and servers take place on an open network, in which an unauthorized user can pose as either a client or a server and intercept or tamper with communication. When a client requests a logon, it sends an initial authentication request to the Kerberos Key Distribution Center (KDC). If username+password are provided, then the password hash is used as symmetric key for the encryption of parts of the request and the reply message. Consequently, a user providing the wrong password will not be able to decrypt the reply message from the KDC. When the user logs in with a smartcard, then the request is encrypted with the private key. The KDC encrypts the reply with the user's public key, which the user can only decrypt with her matching private key.

In order to integrate own authentication methods, it is only necessary to implement the initial authentication step with the KDC accordingly. From that point on, Kerberos works independent of the authentication method.

Mica [18] is an authentication framework for Windows server domains, allowing operators to implement flexible authentication schemes. It consists of server and client software. The Mica server [18] is run on a domain controller and the client replaces the Gina (Graphical Identification aNd Authentication) login component on computers. The client has a modular architecture and can access plugin libraries to interface with additional authentication hardware, e.g., RFID readers, fingerprint readers, or infrared interfaces.

System administrators can now centrally define authentication policies for groups of computers using a proprietary script language. When a client starts the login process, the corresponding login script is started on the server. Certain commands can request credentials from the client which are then verified on the server. The Mica server stores credentials for additional authentication methods also in the Active Directory (AD).

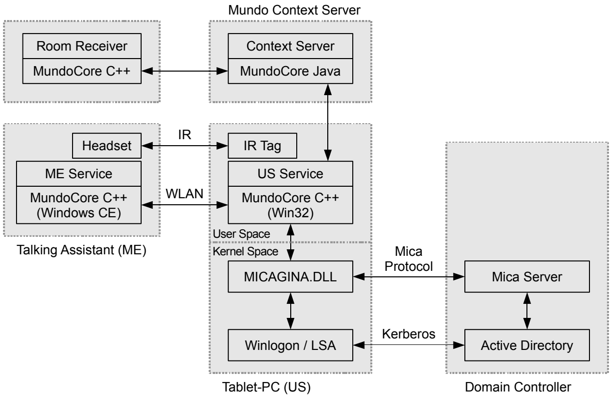


Fig. 4 Software Architecture

This architecture offers a great flexibility in defining authentication schemes. The scheme implemented depends on the criticality of a computer and the available reader/scanner hardware. For example, for less critical systems a company could use the same RFID cards they use for opening doors in the building for authentication. More critical systems could require users to provide RFID+password to verify possession and knowledge.

When a login attempt is evaluated successful, then Mica generates a random symmetric user key. This key is written into AD on the server side, and sent to MicaGina, which forwards the key to WinLogon on the client side.

MundoCore [2] is a communication middleware that was specifically designed for the requirements in ubiquitous computing. With its modular architecture it can be run also on resource-constrained devices. In the context of this application, especially its ability for spontaneous networking is important. We use WLAN as main communication network between ME and US. The infrared link is only used for authentication purposes.

Our **Context Server** gathers information from the sensors embedded in smart environments and is responsible for transforming these sensor readings into information that is meaningful to applications. To do so, it transforms data from various location systems into a common representation, maintains a geometric world model and records events for later queries. Application can issue queries to the context server or subscribe to context changes. In this application, the US uses the context server to detect when the user leaves the room.

The **US Service** installed on the Tablet PC communicates with MicaGina in kernel space. Our authentication protocol is carried out between ME and US service.

The **ME Service** installed on the TA detects nearby devices and decides whether to authenticate with a device or not. A policy defines if the ME should always authenticate, never authenticate, or ask the user. In the latter case, the user can use a graphical or speech-based interface to decide. The necessary credentials for the login are stored on the ME. ME and US share a private infrared link that allows them to detect each others proximity and to exchange secrets.

5.1 Application

A typical use of our system involves the following steps:

1. The user puts the TA on and has to authenticate with the device. Users can use smartcard and PIN for identification.
2. When the user approaches the Tablet PC (Fig. 2(a)), he is asked if an association should be performed.
3. The user can speak "yes" to accept, or "no" to decline. Users can also define to accept all logins automatically in a policy.
4. The authentication protocol is performed as described in section 4. After the association, the TA sends a login request to the PC to start an interactive user session.
5. When the TA stops receiving IR messages from the PC for several seconds, it sends a lock request. The PC then locks the windows session.
6. When the TA receives beacons from the PC again, it sends an unlock request to the computer.
7. The PC performs a logout if one of the following situations occurs:
   - The user performs a manual logout.
   - If the computer is stationary, the system performs an automatic logout when the user leaves the room. To detect this event, the US service on the PC issues a subscription to the context server.
   - The Tablet PC remains locked for several minutes.
   - The network connection between TA and PC breaks.
8. When the user puts off the TA, the smartcard is locked and the keychain becomes inaccessible.

6 Conclusion

We have described a zero-interaction, location-aware authentication system for Windows Server domains. The system is based on location-limited channels for secure authentication and uses location tracking technology to establish different zones around computers. The notion of putting more intelligence into the user's identity token looks promising to improve usability, security, and better protect the user's privacy.

Our experiments indicate that infrared is a very suitable technology to implement location-limited channels, since IR does not pass through walls and the properties of the IR channel (range, directional characteristics, energy requirements, etc.) can be tailored by picking the appropriate hardware components and protocols. We plan to make more extensive user tests in our future work.

Acknowledgements The authors want to thank Jan Ötting and Henno Buschmann from usd.ag for extending the Mica framework to support the TA.

References

1. usd.de ag: usd Homepage. www.usd.de
2. Aitenbichler, E.: System Support for Ubiquitous Computing. Ph.D. thesis, TU Darmstadt University of Technology (2006)
3. Aitenbichler, E., Kangasharju, J., Mühlhäuser, M.: Talking Assistant Headset: A Smart Digital Identity for Ubiquitous Computing. In: Advances in Pervasive Computing, pp. 279–284. Austrian Computer Society (2004)
4. Avoine, G., Oechslin, P.: RFID Traceability: A Multilayer Problem. In: The 9th Intl. Conf. on Financial Cryptography (FC'05), pp. 125–139. Roseau, Dominica (2005)
5. Balfanz, D., Smetters, D., Stewart, P., Wong, H.: Talking to strangers: Authentication in adhoc wireless networks. Symposium on Network and Distributed Systems Security (NDSS '02), San Diego, California (2002)
6. Buhan, I.R., Doumen, J.M., Hartel, P.H., Veldhuis, R.N.J.: Feeling is believing: a location limited channel based on grip pattern biometrics and cryptanalysis. Technical Report TR-CTIT-06-29, Enschede (2006)
7. Capkun, S., Cagalj, M.: Integrity regions: authentication through presence in wireless networks. In: WiSe '06: Proceedings of the 5th ACM workshop on Wireless security, pp. 1–10. ACM Press, New York, NY, USA (2006)
8. Claycomb, W.R., Shin, D.: Secure Real World Interaction Using Mobile Devices. In: Proc. PERMID / Pervasive 2006 (2006)
9. Corner, M.D., Noble, B.D.: Zero-interaction authentication. In: Conference on Mobile Computing and Networking (MobiCom), pp. 1–11. ACM Press (2002). URL citeseer.ist.psu.edu/corner02zerointeraction.html
10. Coulouris, G., Dollimore, J., Kindberg, T.: Distributed Systems: Concepts and Design. Addison-Wesley Longman (2005)
11. Grunwald, L.: Entschärft - Black Hat: Zurückhaltende Berichterstattung. iX - Magazin für professionelle Informationstechnik - Heise Verlag p. 18 (2005)
12. Kindberg, T., Zhang, K.: Secure spontaneous device association. In: UbiComp 2003, vol. 2864, pp. 124–131. Springer (2003)
13. Mayrhofer, R., Gellersen, H.: On the Security of Ultrasound as Out-of-band Channel. In: 3rd Intl. Workshop on Security in Systems and Networks (SSN2007). IEEE Computer Society Press (2007)
14. McCune, J.M., Perrig, A., Reiter, M.K.: Seeing-Is-Believing: Using Camera Phones for Human-Verifiable Authentication. In: IEEE Symposium on Security and Privacy, pp. 110–124. IEEE Computer Society (2005)
15. Microsoft: Windows XP Professional Resource Kit: Understanding Logon and Authentication (2005). URL http://technet.microsoft.com/en-us/library/bb457114.aspx
16. Schneier, B.: Applied Cryptography. Wiley (1995)
17. Schnelle, D., Aitenbichler, E., Kangasharju, J., Mühlhäuser, M.: Talking Assistant - Car Repair Shop Demo. In: Ubicomp'04 Video Track (2004)
18. SicAri: Subproject Applications. http://www.sicari.de/index.php?id=13&L=1 (2007)
19. Spahic, A., Kreutzer, M., Kähmer, M., Chandratilleke, S.: Pre-Authentication using Infrared. In: Privacy, Security and Trust within the Context of Pervasive Computing, *Kluwer*, vol. 780, pp. 105–112 (2005)
20. Want, R., Pering, T., Danneels, G., Kumar, M., Sundar, M., Light, J.: The Personal Server: Changing the Way We Think about Ubiquitous Computing. In: Ubicomp'02, *LNCS*, vol. 2498, pp. 194–209 (2002)