Sensor-Enhanced Authentication Token for Dynamic Identity Management

Mark Smith, Mat Hans
Mobile and Media Systems Laboratory
HP Laboratories Palo Alto
HPL-2004-226
December 8, 2004*

E-mail: {msmith,mhans}@hpl.hp.com

sensor, authentication, identity management, security communication protocol

Identity management as a component in applications is increasing in importance. Many applications need to manage user identities in a dynamic way because threats or conditions under which the application is to be used can change rapidly. We describe a new system for dynamic identity management that can accommodate changes in authentication requirements based on context. Central to this system is a user neutral, context aware token that is worn by a person seeking access to resources or assets. The token device stores information related to the wearer's security permissions including biometric data. A unique feature of this system is the use of user attachment sensors that detect if the token has been removed from the wearer's possession, and return it to its neutral state. Using this token, we explore security in context aware and dynamic systems, and give details from a prototype system.

* Internal Accession Date Only
Approved for External Publication

Sensor-Enhanced Authentication Token for Dynamic Identity Management

Mark Smith, Mat Hans
Hewlett-Packard Laboratories
{msmith,mhans}@hpl.hp.com

Identity management as a component in applications is increasing in importance. Many applications need to manage user identities in a dynamic way because threats or conditions under which the application is to be used can change rapidly. We describe a new system for dynamic identity management that can accommodate changes in authentication requirements based on context. Central to this new system is a user neutral, context aware token that is worn by a person seeking access to resources or assets. Such resources use intrinsic secure authentication points (SAPs) to communicate with the tokens. The token device stores information related to the wearer's security permissions including biometric data. This information is loaded during an initialization step after the token is attached to the person. At this time the person is authenticated, resulting in a system that does not permanently bind a token to a particular wearer. The token is equipped with a heterogeneous sensor set that provides continuous context data, including user attachment and proximity information. A unique feature of this system is the use of user attachment sensors that detect if the token has been removed from the wearer's possession, and return it to its neutral state. The proximity sensors allow a SAP to detect if the wearer is attempting to gain access to an asset. The token's data processing system has sufficient computing power to allow it to execute an elliptic curve based secure communication protocol over which all transactions with secure authentication points wirelessly take place. Using this token, we explore security in context aware and dynamic systems, and give details from a prototype system. The system has received considerable interest from healthcare providers, commercial aviation and military security sectors.

1. Introduction

Networked applications often utilize some form of access control to assure that unauthorized individuals do not gain access to confidential information or restricted resources. Similar technologies are also used in secure sites to authorize access to various parts of a building or other physical assets. The secure access protocols often require a user to memorize multiple passwords or to carry assigned electronic identification cards. Although the authenticity of the password or ID card can be verified by the querying system, the system cannot necessarily identify that the person typing the password or presenting the card is the real owner. In principle an asset may be equipped with hardware that also allows it to authenticate the person presenting the card or password. If the person is identified directly using a biometric method, then an identification card is not needed. Although biometric devices exist, providing reliable and robust hardware at all possible assets would be prohibitively expensive. Additionally, such deployment may not accommodate dynamic changes in security requirements of the protected asset.

The solution presented here provides a system to perform dynamic security using a sensor based token that interacts with Secure Authentication Points (SAPs) that are attached to or built into protected assets. The sensor based tokens uniquely solves the following problems. Identity tokens are not permanently assigned to a user, but instead reside in a neutral state and are bound to any user on demand. This results in a token that cannot be lost and used by another individual.

The token is bound to a user potentially using strong biometrics, and then represents the user as a form of biometric cache whenever the user presents himself to an asset. At the time the token is bound to the user, other data may be stored on the token as well, such as specific assets that may be used, and conditions under which they may be used, such as a specific time and place, accompanied by another person, actively performing a specific task or other requirements. This means that the token can authenticate the user to an asset based on the context by which the asset is to be used. This also addresses the problem of dynamic security in that as the security requirements of an asset changes, the data bound to the token can dynamically change with it. The result is an authentication system that adapts to both the user, the asset and any context requirements that must be followed to satisfy security needs. One contextual parameter is if the token is separated from the user, in which case it returns to its neutral state. Sensors that detect separation of the token from the user's body as opposed to detecting if an attachment device, such as a strap or clip, has been opened may also be used. This results in users not being able to lose their authentication token. If a user needs a new token, any token may be taken and rebound to him. A diverse collection of sensors and communication devices allows the token to interact with a large variety of SAPs.

2. A Token Based Solution

2.1. Related Work

Electronic tags used for identification are relatively ubiquitous, and have been reported extensively in the literature [WAN1]. For human ID management most existing tags are statically and permanently assigned to a user, and because of this are exposed to typical risks and costs associated with loss or theft. These tags use relatively simple methods of identifying the user, such as bar codes, magnetic stripes or passive RFID. Other more complex tags, tokens and supporting infrastructure have been reported that associate a user with location information to enable an application [HAR,WAN2]. These systems are more advanced in that they can be altered and in some cases have user interfaces intended for interactive use in an application. Related tokens exist in the commercial space that not only identify a user, but also interface directly with security protocols such as those in computer operating systems. These systems statically assign the tokens to single users, and are constructed to protect a single type of asset. An example of such a system can be found in [ENS].

Our contribution is to incorporate context into the use and security model. Context implies a knowledge of identity, location and under what conditions an action or activity is taking place. By using context, our identity management token can be used in systems that can command, control and protect a heterogeneous collection of assets. Different levels of security from none to extreme can be invoked depending upon the context in which the token is being used. Using context also contributes to the use model in that security parameters can be sensed and supplied automatically. This frees users from having to interact with the tag and security system allowing them to focus on the use of the asset the system is protecting. From an engineering point of view, this is done by exploiting several sensor and communication types on the token platform, and using them to comply with the various security demands of the assets using an appropriate communication and encryption model. The security demands can change at any instant depending on a number of context conditions, such as the identity and number of individuals, location and orientation, time, target asset, procedure, and other simultaneously or previously occurring events. This give rise to

the idea of dynamic security, meaning that the parameters to satisfy a security requirement can change in real time. Because the tokens are context aware, they are able to provide the information for the changing security requirements.

2.2. A Sensor-Enhanced Authentication Token

A person seeking access to an asset is represented by a sensor-enhanced authentication token. The token includes a data processor having memory, a transceiver, and sensors, including attachment and proximity sensors. The token allows a user to be authenticated once and authorized many times during the rest of the day. In a workplace scenario, a user will pick up a token on arrival to work. The token as picked up is in a neutral state, but will sense when the user possesses it, for example by attaching it to his clothing. At this point the token only knows that a person possesses it, but not yet who. The identity of the wearer is next established and bound to the token using any desired method, such as a walk-up kiosk for biometric authentication or using a combination of biometric measurements and passwords. After successful authentication, security clearance data, e.g. X.509 certificates, device access codes such as PINs and other data such as schedules, restrictions or privileges are uploaded into the token's memory. These certificates are then used over a variable time period for authorization when interacting with assets that are secured by the system, such as information terminals, rooms and equipment. Each of these assets have associated with it a SAP which provides a means of communicating with the token. The SAP itself is specialized to the needs to the asset it is protecting, for example a door might just have a Bluetooth radio to communicate with the token, but a data terminal might use proximity data from the token in the form of an IR sensor to determine if someone is actually positioned in front of it. An example of a data terminal using IR both to detect user presence and orientation, and for data communication is shown in Figure 1.

The token is equipped with a user attachment sensor. Upon detecting the removal of the token from the wearer, this sensor causes information stored in memory to be expunged, returning the token to the neutral state. Sensors used to detect token removal include smart clips, strain gauges and motion sensors. In addition, sensors that directly measure the token's proximity to a human body are currently under investigation to provide further robustness. If a wearer finds that they have inadvertently removed or misplaced their token, all they need do is find another one, bind their identity to it, and continue with their activities. It also means that someone finding the misplaced token will only find a neutral token, which they in turn may wear and bind their identity to. Any token can be worn and bound to by any user.

To resist eavesdropping and be resilient to malicious users, a robust security communication protocol was designed. For this, a temporary, one-time secret key is created to encrypt communications between a token and a SAP. The elliptic curve cryptography version of the Diffie-Hellman protocol is used to generate this secret key. Token sensor noise is used as a seed for random number generation. The security algorithms have been tuned to run efficiently on the processing hardware available on the token, and can perform encrypted communication in real time.

2.3. State Machine for Security

The authentication tokens implement a state machine that allows a consistent interface to be presented to a diversity of smart access points. The state machine accommodates the sensing, communicating and power requirements of the tokens, and at the same time allows different

Figure 1: SAP Infrared Transceiver Located on Data Terminal Display

access points to use the token in a way consistent with the requirements of the protected resource. Figure 2 shows the state machine implemented in the tokens. It has three major subsections or phases called Activation, Discovery and Communication, which reflect the major function of the device in the dynamic security environment.

The activation phase deals with the progression from a neutral state to one where the token has established a binding to a user who now possesses it. State 1, the neutral state, is when the token is not associated with any user. It enters this state from power up, detection of any kind of failure, or when the sensors indicate that no user is in possession of the token. When the sensors on the token determine that a potential user has taken possession of it, it enters state 2. In this state no security parameters or user information is present in the token. The token obtains this data by discovering and connecting to an authentication resource. The authentication resource is another smart access point connected to a user verification device, such as a networked biometric sensor. When this is discovered, an encrypted connection with the token is made, the identity of the user established and this information along with dynamic security parameters, permissions, access codes and other data are downloaded to the token over the encrypted connection. At this point the token enters state 3 in which it is active and bound to a specific user.

The discovery phase deals with how the token is discovered by the secure authentication points associated with the resources the user needs to exploit. Power management is an integral part of the system. It is accomplished by requiring the secure authentication points to beacon to the tokens rather than the opposite. Beaconing can occur continuously, be initiated by the user opportunistically, or occur as a consequence of sensed context. Examples of these could be turning on a protected device or the user positioned in a certain location or orientation. Before discovery the battery powered tokens are placed in a low power state until interrupted into an active state by the signal sent from the SAP. While in this lower powered state, shown as state 4, the token only needs to manage SAP discovery and the sensors that verify possession. Once the opportunity to connect to a secure authentication point is found as in state 5, the token proceeds to the communication phase.

The communication phase is used to set up a secure communication link to the SAP, exchange across it required data, and maintain the link if required by the device or service connected to the SAP. These are shown as states 6 and 7 respectively. In some cases the device or service will only

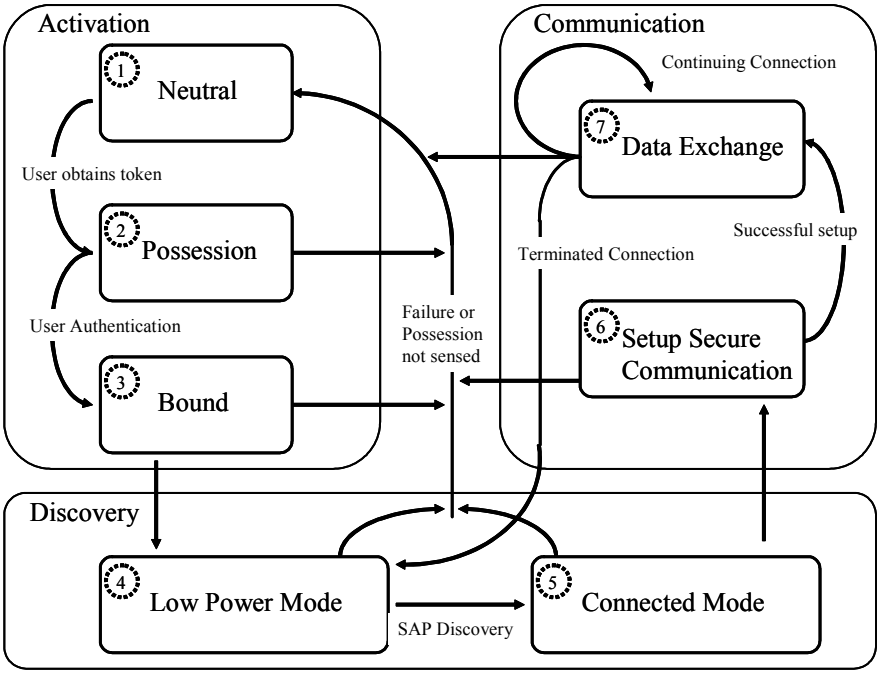

Figure 2: Token State Machine Activation, Discovery, and Communication Phases. The 7 states are indicated within dashed circles.

require a single authorization transaction, for example a door. When all the conditions required by the door to open are met, the door opens and the secure connection between the SAP and the token is torn down and the token returns to state 4. Other resources may require continuous authorization transactions while in use, for example a terminal showing classified data. In this case the link is maintained while the resource is in use.

From the SAP point of view, the token responds to a collection of commands that allow it to react to requests to provide or store data. The nature of the SAPs and devices they protect can be very diverse, and can exploit the properties of dynamic security by changing their authorization or admittance requirements based on context, such as time of day, location, number and identity of people requesting admission and intended use. These can change at any time reflecting the dynamic nature of the security needs.

2.4. Security Communication Protocol

To resist eavesdropping and be resilient to malicious users, a robust security communication protocol was designed and implemented. A temporary, one-time secret key is created to encrypt communications between a token and a SAP. The elliptic curve cryptography version of the Diffie-Hellman protocol is used to generate this secret key [BSS]. When a token wants to communicate with a SAP, both token and SAP generate a random number, for example r_1 for the token and r_2 for the SAP. An aggregate of noise from the token's sensors is used as a seed for random number generation. Both the token and the SAP share the knowledge of a point P on the elliptic curve used. The token then transmits the product P^r_1 to the SAP, and the SAP transmits P^r_2 to the token. At that point, both sides may compute the key P^r_1^r_2, which is almost

impossible to construct in a practical amount of time with only the knowledge of P^r_1 and P^r_2. This key P^r_1^r_2 is then used to encrypt data transmitted over the communication link, e.g. using DES. Note that the keys can be generated whenever the application requires it, for example every few minutes or at each SAP, and that by using different random numbers every time a new communication link is created, the risk from a replay attack is eliminated. Additionally, if point P is a shared secret between the tokens and all SAPs, then this protocol is also robust to eavesdroppers. Distribution of this data point P to tokens may be done at activation time in state 2.

As noted above, the system can utilize any type of transceiver to construct the communication link between the token and the relevant terminals. In the prototype, infrared sensors are used because such sensors require relatively low power and provide additional security over other types of communication links such as RF links. An eavesdropper can monitor an RF signal from a location outside of the cubical or room in which the token and terminal are operating. In contrast, light-based systems require the eavesdropper to have a clear line of sight to both the terminal and token transceivers.

2.5. Possession Sensors

The token uses sensors to determine if the user still possesses it. If the token detects that its current owner no longer possesses it, it reverts to the neutral state 1 by expunging all stored user and security information. Any form of sensor that provides a signal when the token is removed from the user can be utilized as the possession sensor. For example, the token can be attached to the wearer's clothing via a clip that can detect its opening. Wristbands, neck straps and attachment pins for clothing can all be used in similar ways. A weakness of these approaches is that such sensors infer possession by measuring the state of a clip, strap or pin, without determining if the token is really still in the user's possession, for example unclipped from clothing and placed in a pocket. To improve on this the token may implement some form of biometric measurement to assert that it is still on or next to the authorized individual's body. Sensors that detect body heat or pulse can be utilized for this function. Such sensors may be included in wristwatch-like prototypes in which the sensor is pressed against the wrist of the user when the token is worn. If the token is removed, the temperature will decrease or the pulse signal will be lost. A combination of several sensors can also be utilized for detecting the removal of the token from the wearer to provide increased robustness. The token prototype used in this study has several sensors that can be used in this way. This approach is unique in that these sensors detect if the token is still actually possessed by the user by associating possession with proximity or direct association with the user's body rather than detecting if an attachment mechanism such as a strap or clip is still intact.

2.6. Hardware Platform

The token prototype has been built on top of a research platform called the SmartBadge that is used to evaluate media, sensor, and power management technology in wearable and mobile systems. It is based on one of a series of context aware wearable platforms called the SmartBadges that have been developed by Hewlett Packard Laboratories, the Swedish Royal Institute of Technology, and the University of Wollongong [BMS]. The SmartBadge series of context aware platforms are being used in a number of sensor evaluation studies and in algorithm development for biometrics, agents and services that will exploit location and environment data provided by the on board sensors. The technology in the SmartBadge may be incorporated into a

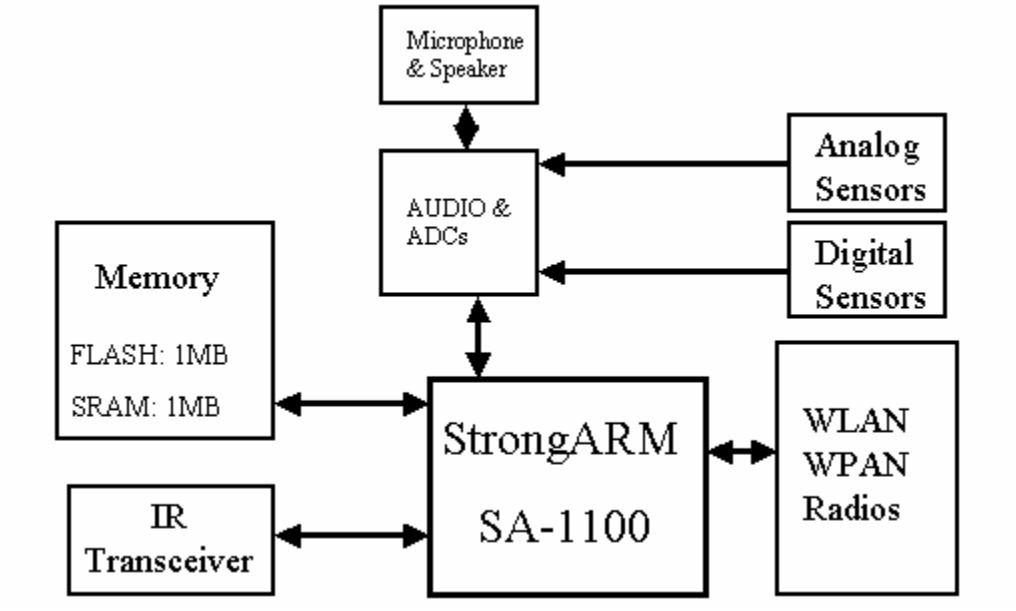

Figure 3: SmartBadge Block Diagram.

variety of wearable platforms for communication, information and entertainment applications as well as in larger mobile and fixed platforms such as automobiles and buildings [HS]. Schematic diagrams of all the SmartBadge platforms are available in the public domain and may be used as a basis for one's own design [MAG].

Figure 3 shows the organization of the SmartBadge used as the token prototype. It is a 64 mm × 115 mm card that integrates a microcontroller and memory system with a rich collection of sensors and communication capabilities. An Intel SA-1100 StrongARM processor is used which provides a good balance of cost, I/O capability, system power requirements and performance. The performance is sufficient to support the token's security requirements.

The sensor set intrinsic to the badge includes front and back temperature and humidity sensors, audio sensing, light level and acceleration in 3 axis. Having temperature and humidity sensors on both sides allow wearable applications to measure these parameters on both the side facing the wearer, and the side facing away. There are also digital and analog channels available to be deployed off of the token. This capability is useful for other sensors such as biometric possession devices and I/O devices such as cameras, button pads and displays.

Wireless communication is provided by two means: the first is an infrared interface supporting bit rates up to 4 Mbit/s; the second is provided by PCMCIA slots and allows plugging commercially available radios for networking and communication. It is possible to accommodate two wireless devices, for example local area and personal area connectivity at the same time. In addition to communication, the infrared and RF capabilities provide a means to determine location and orientation.

Low power consumption and the ability to aggressively manage power usage was a central design goal. The current SmartBadge platform typically operates under 0.40 Watts without the radio card at processor clock rates of about 200 MHz, and 1.0 Watt with an 802.11B radio card (PCMCIA Orinoco gold card running at 3.3V). Two alkaline AA batteries provide power for the

prototype token. The entire token unit with batteries and radio weighs 8 oz. A photo of the prototype token with an added smart card reader is shown in Figure 4.

Figure 4: Prototype Dynamic Security Token.

3. Prototype System and Status

The prototype authentication token system was designed to address the needs of a major Health Care Provider (HCP). The HCP was putting in place mechanisms to comply with new US HIPPA (Healthcare Information Privacy and Portability Act) laws. This involved a design for new secured and accountable electronic medical records. They also wanted to add secure access and accountability tracking to other assets in their clinics, such as examination rooms and diagnostic equipment. In addition to satisfying the HIPPA laws, the HCP realized that analyzing long-term records, procedures and results would lead to a deeper understanding of best medical practices. On the business side, operations in the clinics would be tuned.

The prototype token system consists of the sensor enabled tokens, a biometric kiosk, and PCs used to show patient records. The PCs also act as SAPs for the electronic medical record system and include IR sensors to interact with the tokens and an API to allow token data to unlock the PC and bring up patient data. This API interacts with both the medical information database and with the normal password and security layer of Microsoft Windows© operating system running on the PC. The token state machine is integrated with the application, written in C, running on the token. Both SAP and token communicate serially using the IrDA infrared protocol.

The HCP was especially concerned that access to assets were to occur automatically, and that physicians not be burdened with any overhead connected with the system. To accommodate this the prototype system allows for the following actions. A physician entering the clinic would take

a token out of a basket, and attach it to her clothing. Binding the physicians ID to the token is done using a commercial iris scan device, which sends to the token the identity of the wearer following the encryption scheme used by the token. Physicians then entering a clinical space can present themselves to the PC information terminal to obtain patent information. By using infrared communication, orientation and proximity to the terminal is established. Patient records are brought up following the successful login to the PC by the token, which implies that the correct physician for that patient is oriented facing the display. Login takes place and patient records obtained without any other action by the physician, which means that even if the physician were to be performing a sterile procedure she does not need to touch anything. When the physician walks away from the terminal, it locks and blanks the display. When another physician presents himself to the display, the first physician is logged out, and the next logged in using his token. At the end of the day, the physician returns the token to the basket to be recharged. Because loss of token possession is sensed, the user data in the token is expunged. Future architectures will extend the security token use to the rest of the clinic. We are currently building on top of this token solution, and are exploring security in context aware and dynamic systems, in other areas such as airports and university campuses.

4. Future Work and Conclusion

In summary, the key benefits of our solution are as follows:

- The sensor equipped token uses context measurement and secure communication to allow it to satisfy parameters that allow a user to access a protected asset. These parameters can change dynamically, and the token can respond appropriately.
- Guarantees increased security because of enhanced ease of use. The SAP technology is transparent to the user so the user can concentrate on the task at hand.
- Accelerates user authorization. The user is granted access to the SAP through transactions by the token. The speed is only limited by the speed of the processor and communication.
- Nothing to misplace or lose. If a user loses her token, she can get another one and authenticate herself while sensors on the lost token determine that it is no longer in her possession, forcing the deletion of all on-board security data.
- Traceability. Auditing and accountability are made possible since the token and SAP may log all transactions and communicate that log on regular basis back to a central system.
- Deployment in Heterogeneous Infrastructure. The security clearance data stored on the token is independent of the security communication protocol and so it may be deployed in any secure site, for example traditional cryptographic systems based on public/private key or symmetric key or X.509 certificates.

The study with the HCP also revealed numerous areas where improvements and additions can be made, such as user personalization, diverse methods of user verification and biometrics, more smart space interaction and new application areas. This has given rise to the future dynamic security token shown as a drawing in Figure 5. User personalization is desirable as a feedback mechanism to let the token wearer know that their identity has been correctly identified, and that they are ready to proceed. In the prototype token a synthesized voice served this purpose by telling the user their token was activated. Because of the context aware capabilities of the token, user feedback of proper functioning can be far richer and include not only the token's user but also other entities making up the current context. For example by including a small display on the

token, it can show different pictures, animations or other data based on the context it is currently in, for example showing images and video to pediatric patents based on the token being aware of who the physician's current patient is. A related improvement is in how user authentication is done. Previously, it was assumed that the authentication of the user is performed by equipment attached to an administrative kiosk. However, authentication hardware may also be part of the token. For example, the token may include a biometric sensor such as a fingerprint scanner or a combination of devices to perform a multi-sensor verification, such as fingerprint and voice. Other ID management devices such as a smart card can be used. In this case the token can be thought of as a way to enhance the smart card with context aware and communication features. Figure 4 shows a smart card used in this way with the prototype token.

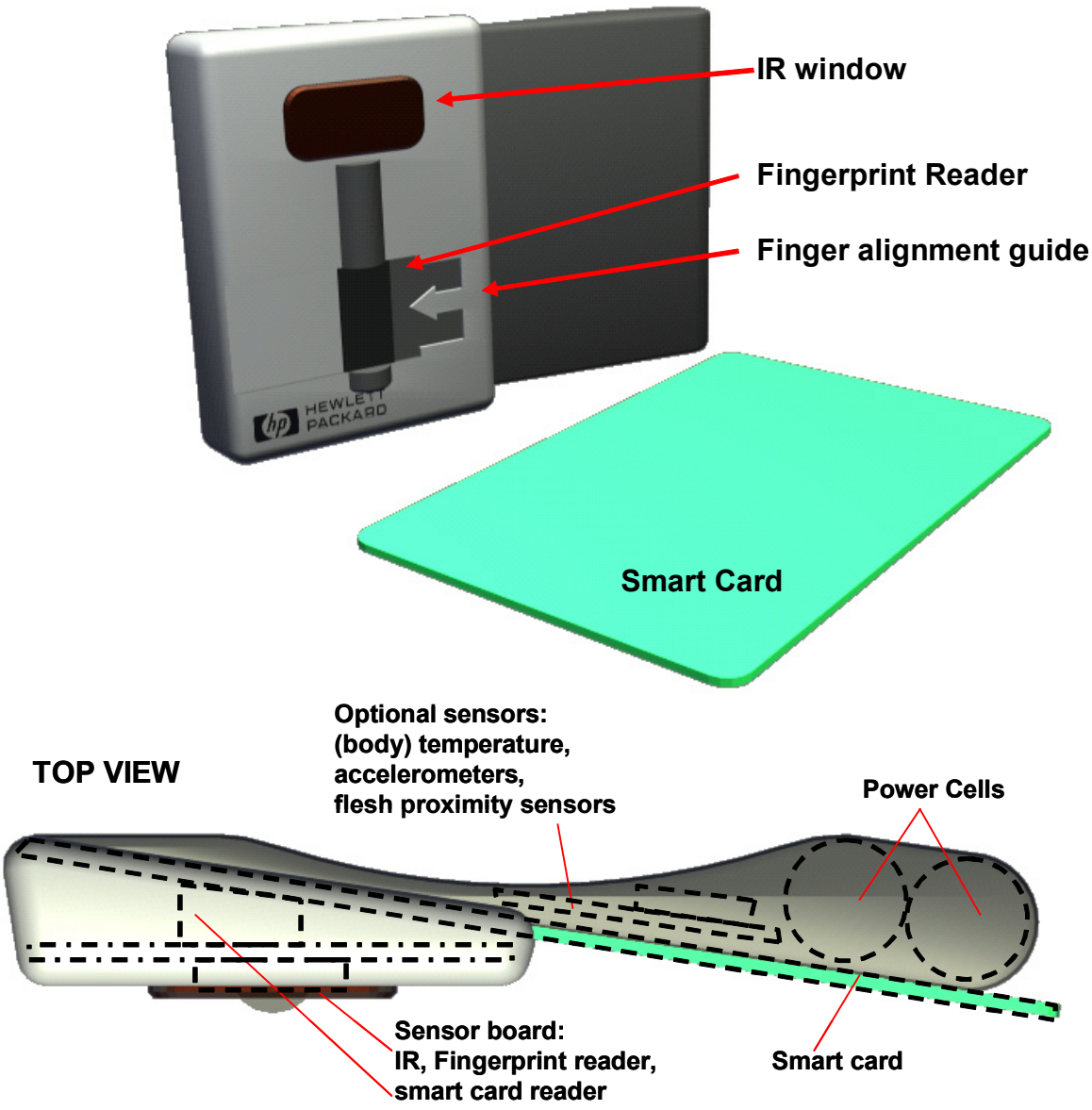

Figure 5: BadgeKey Concept Design.

Extending the functionality of the system to more of the clinical space is also attractive, in effect creating a much larger smart space in which to interact. In the current prototype, only the identity of the wearer and the location and orientation were established to access data. A future system will include other context data such as who the patient is, who and how many other people are in the room. In addition to asset authorization usage, other functions such as verification of a completed patient visit can also be determined. This can be done by using location and schedule data to check that all required clinical resources have been visited by the patient.

Wireless communication diversity has become an area of research due to the varying needs of different assets. With this has come issues of connection latencies and associated power management problems inherent to some radio protocols such as Bluetooth. In these cases, a multi sensor platform is reasonable as the communication method can be matched to the information transfer and latency requirements. Standards would help with this, allowing vendors to be able to offer potentially protected objects and devices that don't need to be retrofitted at a later time and can be deployed into new dynamically secured environments such as the general enterprise space, commercial aviation and defense.

5. Acknowledgments

The authors would like to acknowledge their colleagues John Ankcorn, Ian Blake, Vinay Deolalikar, Mehrban Jam, Ian Robinson, Gary Sasaki, and Gadiel Seroussi for their contribution and discussions.

6. References

[BMS] P. Beadle, G. Maguire., M. Smith, *Using Location and Environment Awareness in Mobile Communications*, Proceedings IEEE ICICS, Singapore 1997.

[BSS] I. Blake, G. Serrousi, N. Smart, *Elliptic Curves in Cryptography*, Cambridge University Press, 1999.

[ENS] Ensure Technologies Xyloc System, http://www.ensuretech.com

[HAR] A. Harter, A. Hopper, *A Distributed Location System for the Active Office*, IEEE Network, Vol. 8, No. 1, January/February 1994.

[HS] M. Hans, M. Smith, *A Wearable Networked MP3 Player and 'Turntable' for Collaborative Scratching*, Proceedings of the IEEE International Symposium on Wearable Computing, October 2003.

[MAG] G. Maguire, SmartBadge Version 4, http://www.it.kth.se/~maguire/badge4.html

[RS] R. Rivest, A. Shamir, *How to Expose an Eavesdropper*, Communications of the ACM, April 1984.

[WAN1] R. Want, K.P. Fishkin, A. Gujar, B. L. Harrison *Bridging Physical and Virtual Worlds with Electronic Tags*, ACM Conference on Human Factors in Computing Systems, CHI 99, pp 370-377, 1999.

[WAN2] R. Want, B.R. Schilit, N.I. Adams, R. gold, K. Petersen, D. Goldberg, J.R. Ellis, M. Weiser, *An Overview of the ParcTab Ubiquitous Computing Experiment*, IEEE Personal Communications, Vol. 2, No. 6, December 1995.

7. Appendix: Security Analysis

We analyze the security of the following communication protocol between a token and a SAP based on elliptic curve encryption on an elliptic curve E over a finite field F_q having a predetermined point P.

7.1. Communication Protocol

The communication protocol between a token and a SAP proceeds in the following steps.

- Step 1. The token generates a random number r_t and computes P^r_t
- Step 2. The SAP generates a random number r_s and computes P^r_s
- Step 3. The token sends the SAP P^r_t
- Step 4. The SAP sends the SAP P^r_s
- Step 5. Both the token and the SAP compute P^r_s^r_t = P^r_t^r_s which is a point on the elliptic curve. This common point is then used to determine the session key.

7.2. Threat Model

There are two scenarios:

- Scenario 1: External node introduced into the system: Here we assume that while the SAP and the token both have knowledge of a predetermined specific point P on the Elliptic curve, while the malicious external node does not have knowledge of P. The malicious node eavesdrops upon the conversation between the token and the SAP. Since the protocol demands that the SAP and token both generate a random number each (call it r_s and r_t, respectively) and compute P^r_s and P^r_t respectively, the malicious node has access to P^r_s and P^r_t, but not to P, r_s or r_t.
- Scenario 2: An internal node has been compromised. In other words, a node that has knowledge of P has been compromised and is now maliciously operating within the system. In this scenario, the malicious node has knowledge of P, P^r_s and P^r_t, but not to r_s or r_t.

Clearly, scenario 2 is the more worrisome one from the security point of view since in this case the malicious node has the additional knowledge of P that is not available to the malicious node in Scenario 1. Therefore we will analyze only Scenario 2 and will show that the protocol is secure under the standard assumptions.

7.3. Threat Analysis

Eavesdropping

In order to compute r_s and r_t from the knowledge of P, P^r_s and P^r_t, the malicious node would have to solve the discrete log problem on elliptic curves, which is thought to be hard provided the number of points #E(F_q) on the elliptic curve E over the finite field F_q satisfies the following:

- Condition 1: The group of points should have a subgroup of large prime order
- Condition 2: #E(F_q) should not be a prime or a prime power
- Condition 3: The least r such that q^r is congruent to 1 mod #E(F_q) is large

Replay Attack

In this attack, the malicious node records earlier conversation between a token and a SAP and tries to replay parts of it later in order to establish a communication with either the SAP or a token. So say that the node has recorded P^r_s and P^r_t and all subsequent conversation. In addition, it has prior knowledge of P. With this information, it can certainly initiate a communication with either party by sending it P^r_t. In return, it will receive P^r_s, But to actually get a session key, it will need to find out r^s and r^t. This would imply a solution of the discrete log problem and is thought to be hard.

Man in the middle attack

This sort of attack is not considered very feasible in a broadcast radio type of channel. This is due to the following reasons: Firstly, the man in the middle would have to decrypt and encrypt messages sent from a token/SAP to a SAP/token which would cause delays. Secondly, the man in the middle would have to prevent the messages from the token/SAP from reaching their intended recipient SAP/token. In a radio channel, this is unachievable without a sophisticated attack on the hardware itself.

Notwithstanding the above, we suggest the following protocol, due to Rivest and Shamir [RS] that would foil an attempted man in the middle attack. The protocol proceeds along the following steps:

- Step 1. The token generates a random number r_t and computes P^r_t
- Step 2. The SAP generates a random number r_s and computes P^r_s
- Step 3. The token sends P^r_t to the SAP
- Step 4. The SAP sends P^r_s to the token
- Step 5. Both the token and the SAP compute P^r_s^r_t = P^r_t^r_s which is a point on the elliptic curve. This common point is then used to determine the session key.
- Step 6. The token encrypts its message with the session key and sends *half* of it to the SAP.
- Step 7. The SAP encrypts its message with the session key and sends *half* of it to the token.
- Step 8. When the token receives the first half of the SAP's message, it sends the SAP the second half of its message.
- Step 9. When the SAP receives the second half of the token's message, it sends the token the second half of its own message.

Note that the entire message is unreadable by the SAP till step 8 and by the token till step 9. If there is a man in the middle, he would have to generate a message each for the token and SAP and send out its first half in steps 6 and 7. So now, even once he has read the token and SAP's messages, he cannot change his own message's first half, and so cannot mount a successful attack.