A New Secure Pairing Protocol using Biometrics

Ileana Buhan
ileana.buhan@utwente.nl,
University of Twente, Enschede,
The Netherlands

Abstract*—*Secure Pairing **enables two devices, which share no prior context with each other, to agree upon a security association that they can use to protect their subsequent communication. Secure pairing offers guarantees of the association partner identity and it should be resistant to eavesdropping or to a man-in-the-middle attack. We propose a user friendly solution to this problem. Keys extracted from biometric data of the participants are used for authentication. Details of the pairing protocol are presented along with a discussion of the security features, experimental validation with face recognition data and results of the usability analysis survey.**

I. Introduction

Nowadays, users, devices and services are designed to interact spontaneously anytime, anywhere. In many scenarios however is it desirable to associate devices in a secure way. For example when using a mobile phone to pay for tickets or when sharing private contact information via the wireless link in an unsecured environment. This problem is known in the literature as secure pairing [5].

The problem of secure pairing is difficult, because typically there will be no a priori shared information such as passwords, addresses, or PIN codes. Neither can we assume a globally trusted third party for certifying users, devices, and services.

Solutions have to be designed such that secure associations can be realized between previously unassociated devices. Security means that the solution must offer guarantees of the association partner identity and must be resistant to eavesdropping or to a man-in the middle attack.

This problem becomes even more difficult when the secure pairing is done by non-expert users without knowledge or interest in security aspects. Such users may range from children to the elderly, with varied skills and capabilities. The ideal solution must provide a balance between security and ease of use.

Consequently, both security researchers and practitioners have been looking for intuitive techniques for ordinary users to be able to securely pair their devices. The issue of intuitive security initialization is more general, applicable whenever ordinary users need to set up secure communication without the help of expert administrators or trusted third parties.

Technical challenges. When two users, Alice and Bob, meet at a conference and decide to exchange business cards or other documents, they talk for a while until they trust each another sufficiently to exchange information. However they do not wish other participants to eavesdrop on their communication or to tamper with their documents. At this stage the only secure association that they have is their trust in each other. To set up a secure association between their devices a protocol is needed that can transfer this trust to their devices. It is not enough for Alice's device to guarantee a secure pairing with device: 128.196.1.3. Alice needs to know that there is a secure association with Bob. Our solution is a protocol that can transfer the trust relation between people to a trust relation between devices using biometrics as the main tool.

User friendliness. The most important reason why security often fails is the lack of user friendliness. To establish a secure communication, Alice and Bob have to agree on a key. From a usability point of view we want Alice and Bob to have minimal interaction with their devices, and the technical difficulty of the required task should be no worse than to dial a number on a mobile phone. Also we do not like the idea of Alice and Bob having to remember a password or a pin code for establishing the communication key. A user friendly solution is readily provided by appropriate use of biometrics, since a fingerprint or the image of a face has the advantage that it cannot be lost or forgotten and is always available.

Contribution. We present a practical solution to the secure pairing problem where biometrics is used to establish a common key between the pairing devices. Our approach has at least two major advantages. Firstly, it offers the possibility to transfer trust from humans to machines without any available security infrastructure. Biometric authentication offers physical validation, thus guaranteeing the identity of a device owner. Secondly, the process is short and we believe user friendly. We propose a protocol in which the keys extracted from biometric data are combined to form a session key.

The idea is both simple and effective. Suppose that two users wish to set up a secure communication channel. Both own a biometrically enabled handheld device (for example with face recognition biometrics). Both devices are equipped with a biometric sensor (a camera for face recognition) and a short range radio. Each device is capable of recognizing its owner for example by face recognition. Then the users take each others picture. Each device now contains a genuine template of its owner and a measurement that approximates the template of the other user. The idea is that each device calculates a common key from the owner template and the guest measurement. In our solution, all Alice has to do to set up a secure communication with Bob is to take a picture of him and let Bob take a picture of her. The protocol is more general: it can be applied on any type of biometric.

Our protocol is innovative compared to a key exchange

protocol in the sense that legitimate users have to "find" the communication key by performing a related key search attack. The reasons are twofold. Firstly, natural variations and noise in biometric measurements lead to non-repeatable binary sequences and our key search mechanism helps lower the error rates in a practical situation. Secondly, the key search mechanism uses the unpredictable randomness between two measurements as a random salt for the session key thus strengthening the key.

II. Biometrics as cryptographic key

Our protocol requires construction of keys from biometric data. In raw form, biometric data is unsuitable to be used as cryptographic key for two reasons. The first is its representation, usually the real domain while cryptographic keys are represented in the binary domain. The second reason is noise and natural variations between biometric measurements.

Dodis, *et al.* [4] proposed a general construction termed fuzzy extractors that allows cryptographic keys to be generated from noisy, non-uniform biometric data. In principle a fuzzy extractor does two things: providing error correction to compensate for the unpredictable noise in the biometric and smoothening the non-uniform representation of biometric data.

There are two main components in a fuzzy extractor scheme: the encoder and the decoder. The *encoder function* is used during enrolment of a user X. As input, it takes a low noise template t of the biometric feature vector and a binary string m (which will be used as a cryptographic key later on), to compute the public sketch w. The binary string m can be extracted from the biometric data itself [7] or it can be generated independently [6]. During authentication, the *decoder function* takes as input a noisy measurement x of the users biometric (e.g. a photograph of the user for face biometrics) together with the public sketch w, and outputs the binary string m if the measurement is close enough to the original biometric. The exact reproduction of the binary string m is required to authenticate user X.

III. SAfE protocol

The SAfE protocol establishes a shared secret key between devices whose owners happen to meet and who have no prior security association. There are three phases in the lifetime of our protocol. The first, is the enrolment which can be regarded as a necessary precondition. The second, is the SAfE protocol which is the action taken by Alice and Bob to achieve their goal which is secure communication the third and final phase. We detail these phases below.

1. Enrolment, is performed once in the lifetime of the protocol. This step is performed by both Alice and Bob, independently.
Each participant takes multiple measurements of his own biometric, and uses these to calculate his biometric template vector t. Next, each participant picks a random string m, and uses the encoder function of the fuzzy extractor to calculate the matching public sketch w. We use t_A, m_A, w_A for the template, key and public sketch of Alice and t_B, m_B, w_B respectively for Bob.
After enrolment we have achieved that: (1) the identity of a user can be verified by her own device, and (2) a device is prepared to be paired up with another device on which the SAfE protocol has been implemented.
2. Pairing, where the SAfE protocol is used to create a secure channel, a secret key is computed by the decode function of the fuzzy extractor. The protocol description below provides all the details of this step.
3. Secure communication, when the paired users send messages, documents etc. encrypted with the key they derived by the SAfE protocol.

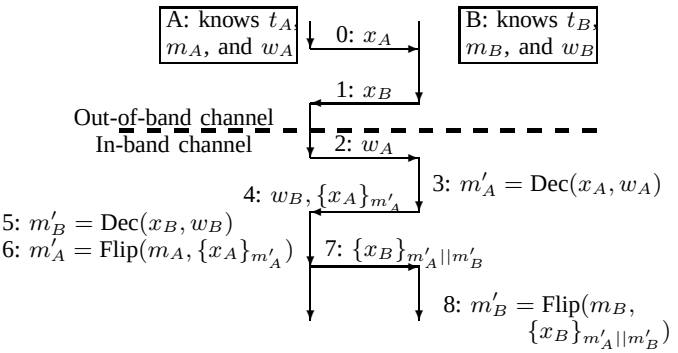

Fig. 1. *Message flow for the SAfE protocol showing the steps taken by Alice to the left (5,6) and Bobs actions to the right (3,8) to pair their mobile devices. The steps in the middle represent the message exchange on the out-of-band channel (0,1) and the in-band-channel (2,4,7).*

SAfE details. The SAfE protocol uses two communication channels for key establishment as in the pairing model proposed by Balfanz, *et al.* [1]. One, the in-band channel, is used for authentication. This channel has a high bandwidth but offers no security guarantees. While the second is the out-of-band channel used for pre-authentication. This channel has a low bandwidth but offers some security guarantees like authentication, integrity or confidentiality. In the SAfE protocol we use the out-of-band channel to exchange a limited amount of information. Later, we use this information to establish the common key by exchanging messages on the in-band channel.

Out-of-band channel. In the SAfE protocol we use biometrics as the out-of-band channel. The first reason for our choice is that biometrics is a source of high entropy data which means high bandwidth compared to other out-of-band channels (e.g. infrared). The type and quality of the biometric modality used (fingerprint, face, iris) determines the value of the bandwidth capacity for the out-of-band channel.

The second reason for biometrics as an out-of band channel is that it is easy to send messages on this channel since the main characteristic of biometrics is user friendliness. For the SAfE protocol, the particular type of biometrics used for sending messages is not important.

In-band channel. The in-band channel is a broadcast channel

(e.g. WLAN) thus all messages sent on this channel are public.

Message Flow. The message flow of the SAfE protocol is presented in *Figure* 1. Without loss of generality we may assume that Alice starts the protocol. We explain each of the steps:

0: Bob measures Alice's biometric. This is shown as a transfer of the measurement x_A from Alice to Bob on the biometric channel.
1: Similarly Alice takes a measurement of Bob's biometrics, yielding x_B.
2: Alice broadcasts her public sketch w_A on the wireless channel.
3: Bob feeds the public sketch w_A and the measurement x_A of Alice to the decode function of the fuzzy extractor to compute a key m'_A.
4: Bob broadcasts $w_B, \{x_A\}_{m'_A}$, i.e. the tuple consisting of w_B and the encryption of x_A using key m'_A.
5: Alice uses w_B received in plain in Step 4 and x_B received in Step 1 to compute m'_B with the decoding function of the fuzzy extractor.
6: The second part $\{x_A\}_{m'_A}$ of the message is used to compensate for eventual errors in decoding m_A. We expect that due to noise or poor quality of the biometric sensor $m_A \neq m'_A$: However, due to their construction m_A and m'_A are close in terms of the Hamming distance so that Alice can perform an efficient key search algorithm to obtain m'_A from m_A. The key search algorithm systematically flips bits in m_A until $\{x_A\}_{m'_A}$ can be decrypted successfully (see the key search algorithm below for details). Since Alice can recognize a measurement of her own biometric, she can check the decryption results.
7: Alice broadcasts $\{x_B\}_{m'_A||m'_B}$.
8: Bob also performs a key search, flipping bits in the concatenation of m'_A and m_B until x_B can be decrypted successfully.

The action on the out-of-band channel "Bob takes a measurement from Alice" can be translated to: "Bob takes a picture of Alice" when face recognition biometric is used.

Key search algorithm. In classical symmetric cryptography to decrypt a message encrypted with a key m one must posses m exactly. In particular, with a key m' that differs only in one bit from m, decryption will fail. The SAfE protocol uses this apparent disadvantage of symmetric key cryptography as an advantage: m' is used to form the session key. The noise of the measurements is used as random salt [10] for the session key. The key search algorithm makes it possible to recover m'. Before the algorithm starts we decide on how many trials we make to discover the key. If we set the error threshold to τ bits the algorithm will try out at most $\sum_{i=0}^{\tau} \binom{N}{i}$ combinations before key search failure is declared. Then the protocol has to be restarted or the user gives up. We note that during the protocol both the devices have to perform the same amount of computation, which makes the protocol fair.

Smart Key Search. When the key space is large the approach described above can become prohibitively expensive and unusable in practical situations. To increase the search speed with which Alice finds m'_A from m_A we propose a method that computes weighting coefficients on each of the key bits. The weight associated with a particular bit represents the probability of error for that bit. The vector of N weighting coefficients for a particular user is the *error profile*. The error profile gives, in fact the order in which bits are flipped. For example assume that 1 bit is changed in m'_A. Without error profile all N bits are equally likely to flip thus on average Alice will have to perform $\frac{N}{2}$ flips. On the other hand the error profile gives her the position of the most likely bit, giving an advantage.

There is another important reason for using the error profile enhanced key search. Due to the nature of the protocol, Alice only has to find variations of her own key m_A and not keys coming from other parties. In particular, this means that we can reduce the false rejection rate without significantly increasing the false acceptance rate.

Error profile. The error profile is a vector of probabilities of length equal to the number of elements in the feature vector. The probability associated to one feature is computed during enrollment and represents the FRR for that feature. This information is useful since it is an indicator for the reliability of a feature. The higher the FRR for a feature the more unreliable that feature is, thus the higher the probability of an error in the output of a fuzzy extractor. In a biometric recognition system this information is disregarded and an overall FRR for all features is estimated. We show that using the error profile as side information can lower the error rates of a biometric recognition system. In the evaluation of our protocol we use the fuzzy extractor proposed by Linnartz, *et al.* [6]. Details about the error profile computation can be found in the extended version of this paper [3].

Key Search with Error Profile. When the template t and measurement x belong to the same user, for example Alice, we expect a small number of errors to appear during decoding. This means that even if m_A and m'_A are different, the difference should not be more then a few bits which can be further corrected using the error profile e_A. Now, the initial Flip function

$$m'_A = \mathrm{Flip}(m_A, \{x_A\}_{m'_A})$$

can be refined as:

$$m'_A = \mathrm{SmartFlip}(m_A, \{x_A\}_{m'_A}, e_A).$$

We start the key search by assuming that there are no errors in m'_A, and we use m_A to decrypt the message $\{x_A\}_{m'_A}$. If decryption fails we assume there is a one bit error. We start flipping one bit of the key according to the position indicated by the largest component of e_A. If the operation is not successful we assume that two bits are wrong and we try combinations of highest two bits from the error profile. Finally if we reach the limit on the number of trials we assume that the key is coming from an intruder and the protocol is aborted.

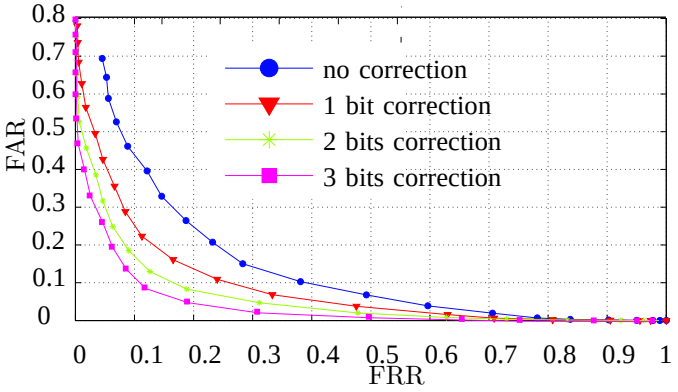

Fig. 2. Experiment 1. ROC *curves for mobile data set, uncontrolled set.*

IV. Experimental Results

We evaluate the performance of the protocol from three different perspectives. Firstly, we evaluate the performance of the protocol with face recognition data. Secondly, we analyze the security of the protocol against an adversary named Eve, who has computational capabilities. Thirdly, we look at our protocol from the perspective of the users. Our usability analysis shows that our subjects find the SAfE protocol fun to use, and that they would like to have the SAfE pairing available on their mobile devices.

Experimental Validation. We present experiments with face recognition data for validating the performance of the protocol. The goal of these experiments is to determine whether it is possible for Alice and Bob to determine their own key using the SmartFlip function.

To verify the potential of constructing cryptographic keys from face data in the ad-hoc settings of our protocol we need a database with faces recorded with a mobile device. Since, as far as we know, such database is not publicly available we recorded our own mobile database. This database contains low-resolution images of 31 individuals, recorded in uncontrolled conditions. For each of the 31 individuals we recorded 4 video files using the same mobile device (ETEN M600+, which has a 2 mega-pixels camera). The four files were recorded in two sessions on two different days, each day we recorded two movies. On the first day each movie was approximately 10 seconds. On the second day we recorded shorter movies of approximatively 5 seconds. Location of subjects (background), pose and light were different in the two sessions.

The algorithm used was proposed by Veldhuis et al. [8] for hand geometry and adapted for face recognition in [2]. The algorithm works as described below. We first trained a generic face model using the FRGC version 1 database. In the recorded movies, we extract frames which contain the face of the individuals. Movies recorded in the first session resulted into 5994 images that were used during enrollment. Movies recorded in the second session resulted into 2959 images that were used during testing.

In each of the images in the data set, we automatically located the faces using the face detection method of Viola-Jones [9] which finds facial landmarks like eyes, nose and mouth. These landmarks are used to align the faces. We only used the first hundred correctly found faces for the recognition in both sessions. For each image the region of interest is selected, the background is removed and the region of interest is normalized to zero mean and unit variance. The difference between the face in the image and the generic face model generated from the FRGC database is computed. As a result each biometric sample can be represented as N (in our case equal to 30) independent feature vectors. On this database, the face recognition is more difficult due to larger deviations in the pose of individuals, illumination and the low quality of the movies.

The ERR using the face recognition algorithm without correction, on the mobile data set is 15.7%. We apply on the processed biometric data the shielding scheme proposed by Linnartz, *et al.* [6], which allows to generate a random key of 30 bits length for each user. *Figure* 2 shows the ROC curves obtained with and without correction. Without any corrections the ERR on the mobile data set is around 29% with 1 bit correction the ERR drops to approximately 19% and after further correcting 2 bits the ERR is approximately equal to the one obtained by the biometric based classifier 15%. By doing 3 bits correction we obtain an ERR of approximately 12%.

We repeated the same experiment on the public FRGC database and on a second biometric modality, hand grip pressure pattern. The goal is to verify whether the improvement of the correction algorithm is independent of the data set. The answer to this question is positive and the details of these experiments can be found in Buhan, *et al.* [3].

Security. Our adversary named Eve, is a passive adversary, i.e. eavesdropper. She can listen to the communication on the in-band channel and can perform a key search operation similar to Alice and Bob to find the communication key. If the out-of-band channel is not confidential she has access to a noisy version of the information sent on this channel. By modeling this adversary we try to answer the following question: If both Alice and Bob have to guess the session key, how much more difficult is it for Eve to do the same?

Due to space constraints, in this paper we look at the particular scenario when Eve has both the picture of Alice and Bob. More scenarios are analyzed in [3]. For the evaluation we use the mobile data set. We show that in the average case when Eve has the biometric measurements of both Alice and Bob her workload is significant. Assume both Alice and Eve execute at the same speed 1 trial operation. Assume further it takes Alice 10 second to perform 10^7 trials. In these settings it would take Eve 3 years to perform 10^{14} trials, expected in an average case scenario. Of course Eve can use more powerful computers or execute operations in parallel. Since our protocol is intended for ad-hoc situations were confidential but not critical information is exchanged, as long as it would take Eve more than 7 days to find the communication key we consider our protocol secure.

Gender	Age	Education
Male: 60% Female: 40%	18−24: 10% 25−29: 56% 30−34: 20% 35−39: 7% 40+: 7%	High school: 7% Bachelor: 17% Masters: 46% Doctorate: 30%

TABLE I
Participant profile.

The workload of Eve, thus the security of the protocol can be increased but it would also increase the error rates. A convenient balance can be found on a case by case basis.

Usability. Our usability experiment had 30 participants from a university environment representing 13 different countries. The demographics such as gender, age and education for our subjects are presented in table I. Most of our subjects have a computer science background. The average computer usage history was around 15 years with an average of 9 computer hours per day. All participants have a mobile phone, a PDA or a laptop.

The main purpose of our experiment was to discover whether users would find it easier to use SAfE protocol compared to a standard 4 digit PIN based pairing. The scores between the two pairing methods were tight. We explain the preference for the PIN based method of some of our subjects by the familiarity with PIN based security (ATMs, Bluetooth) and typing numbers to subjects with a computing background.

Some subjects used the adjective “easy” to describe the SAfE protocol and most of our subjects, 90%, found it fun to perform the pairing using a camera. Others found it easy to understand how the PIN based pairing method works but they used the word “magic” to describe the SAfE protocol. It is worth noting that we experimented with a 4 digit PIN number and approximately 80% of our participants choose the same PIN number(1234).

Our subjects determined some situations where SAfE is not appropriate: (a) when the participants wish to communicate without drawing attention (such as in a restaurant or at a business meeting) (b) when the protocol fails (for example under bad lighting conditions). Therefore a back-up solution for SAfE is needed that is smoothly integrated with the system. The user would then have the choice of a more user friendly biometric based pairing method and a more robust alternative method, 73% of our subjects would like to have both pairing methods available on their mobile device.

V. Conclusions

Personal devices, which are carried around at all times and the dynamic interactions between the owners of such devices demand solutions which are fast, easy and which do not rely on *any* pre-existing security infrastructure. Classical security solutions require either an on-line connection (Certification Authorities which can assign credentials) or previously shared knowledge (a cryptographic key). These assumptions are not always valid in the dynamic world of today. The problem is to find alternative methods to create security credentials, which are both user-friendly and offer good security.

Secure device association is a challenging problem from both the technical and the user interface point of view. Firstly, users need to exploit a common secret source of randomness from which to extract a shared secret key. Secondly, it should be possible to link the device we connect to with the person who owns it. Thirdly, the process should be simple such that for any person with non technical background the protocol is easy to use.

Our approach has at least two major advantages over related work. Firstly, it offers the possibility to transfer trust from humans to machines without any available security infrastructure. Biometric recognition offers physical validation, thus guaranteeing the identity of a device owner. Secondly, the process is short and user friendly. In the pairing protocol the keys extracted from biometric data are combined to form a session key.

We evaluate the performance of the protocol from three different perspectives. Firstly, we evaluate the performance of the protocol with two types of real life biometric data: face recognition and hand grip pressure pattern.

Secondly, we analyze the security of the protocol against an adversary, which can eavesdrop on the communication line between Alice and Bob and has computational capabilities. Thirdly, we look at our protocol from the perspective of the users. Our usability analysis shows that our subjects find the SAfE protocol fun to use, and that they would like to have the SAfE pairing available on their mobile devices.

The advantage of our solution is its inherent user friendliness and strong security guarantees.

References

[1] D. Balfanz, D. K. Smetters, P. Stewart, and H. C. Wong. Talking to strangers: Authentication in Ad-Hoc wireless networks. In *Proceedings of the Network and Distributed Systems Security Symposium (NDSS), San Diego, California, USA*, San Diego, California, February 2002. The Internet Society, Reston, Virginia.

[2] B.J. Boom, G.M. Beumer, L.J. Spreeuwers, and R.N.J. Veldhuis. The effect of image resolution on the performance of a face recognition system. In *9th International Conference on Control, Automation, Robotics and Vision (ICARCV ’06), Sinagpore*, pages 1–6, December 2006.

[3] I.R Buhan, B. Boom, J. Doumen, P.H. Hartel, and R.N.J. Veldhuis. Secure ad-hoc pairing with biometrics: Safe. *International Journal of Security and Networks Special (IJSN), Special Issue on Secure Spontaneous Interaction*, 4(1):to appear, 2009. (Subsumed by Chapter 4 of this thesis.).

[4] Y. Dodis, L. Reyzin, and A. Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In Christian Cachin and Jan Camenisch, editors, *Advances in Cryptology ,Proceedings of International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT 2004), Interlaken, Switzerland*, volume 3027 of *Lecture Notes in Computer Science*, pages 523–540. Springer, May 2004.

[5] T. Kindberg and K. Zhang. Secure spontaneous device association. In Anind K. Dey, Albrecht Schmidt, and Joseph F. McCarthy, editors, *In Proceedings of UbiComp 2003: Fifth International Conference on Ubiquitous Computing,(UBICOMP 03), Seattle, Washington*, volume 2864 of *Lecture Notes in Computer Science*, pages 124–131. Springer, October 2003.

[6] J.P. Linnartz and P. Tuyls. New shielding functions to enhance privacy and prevent misuse of biometric templates. In Josef Kittler and Mark S. Nixon, editors, *4th International Conference on Audio-and Video-Based Biometrie Person Authentication (AVBPA 2003), Guildford, UK*, volume 2688 of *Lecture Notes in Computer Science*, pages 393–402. Springer, June 2003.

[7] P. Tuyls, A. Akkermans, T. Kevenaar, G. Schrijen, A. Bazen, and R. Veldhuis. Practical biometric authentication with template protection. In Takeo Kanade, Anil K. Jain, and Nalini K. Ratha, editors, *Proceedings of the 5th International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA 2005), Hilton Rye Town, NY, USA*, volume 3546 of *Lecture Notes in Computer Science*, pages 436–446. Springer, July 2005.

[8] R.N.J. Veldhuis, A.M. Bazen, W. Booi, and A. Hendrikse. A comparison of hand-geometry recognition methods based on low- and high-level features. In *Proceedings of the 15th Annual Workshop on Circuits Systems and Signal Processing (ProRISC 2004), Veldhoven, The Netherlands*. STW, 2004.

[9] P.A. Viola and M.J. Jones. Rapid object detection using a boosted cascade of simple features. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001), Kauai, HI, USA*, pages 511–518. IEEE Computer Society, December 2001.

[10] T.D. Wu. The secure remote password protocol. In *Proceedings of the Network and Distributed System Security Symposium (NDSS 1998), San Diego, California, USA*. The Internet Society, March 1998.