A Method for Ranking Authentication Products

K. Helkala and E. Snekkenes

Norwegian Information Security laboratory,
Gjøvik University College, Gjøvik, Norway
e-mail: kirsi.helkala@hig.no

Abstract

There is a steady increase in both authentication methods and products implementing these methods. Product selection has impact on strategic factors such as system security, cost and usability. This paper presents a new method for ranking authentication products. The method can contribute towards an improved decision process. Using our method, issues such as technical performance, application/system specific requirements, cost and usability are addressed.

Keywords

Information security, Personnel Authentication, HCI, Decision support, Comparison method, Product evaluation.

1. Introduction

Authentication is about verifying claimed identity. That is, a prover is aiming to convince a verifier that he is who he claims he is. We consider a setting where the prover is a human, and the verifier may be either a machine or a human. Traditionally, authentication makes use of e.g. passwords to log on to computers and keys to enter buildings. However, in the last decade, one has seen an increased availability of new authentication alternatives such as picture based passwords, fingerprints, electronic tokens etc.

Our focus is on the authentication of people. In particular, we are not considering authentication that requires both the verifier and the prover to carry out complicated computations (DES, RSA, AES etc.). However, with respect to hardware tokens, we are including the interactions between the person and the token, but consider electronic communication between the token and other hardware/software system components to be outside our scope. Similarly, authentication between a PC and a file system, or a smart card and a PC is also outside our scope.

Knowing that authentication has an impact on issues such as system security, usability and cost, the choice of authentication product clearly is an important decision. Thus, in a given setting can we establish decisions support by ranking the available authentication alternatives?

We have carried out a small survey among Norwegian enterprises and organizations (a health service, security service, college, and security mechanism provider) to investigate current selection processes. We found that enterprises do not actively engage in authentication product selection, but simply accept what is offered by the vendor. Considering the impact authentication can have on system security, cost and usability, this suggests that enterprises find authentication product selection difficult. To address this problem, we have developed a method that can help enterprises to identify issues that should be considered when choosing authentication products. Our method includes a step-by-step procedure for ranking authentication products taking into account the environment where the authentication product will be deployed. As part of our ranking method we propose an authentication method independent formulation of circumvention hardness.

There exist other methods for comparing authentication methods e.g. (O'Gorman, 2003) and (NIST, 2006). Our method improves O'Gorman's method by including usage scenarios. Thus, we are increasing the applicability of the ranking. While O'Gorman's method is general, focusing on authentication *methods*, our method provides strategy and formulas for ranking authentication *products*. The NIST special publication (NIST, 2006) is limited to the security comparison of password/token based schemes. Our method follows the NIST guidelines but widens the applicability to include biometrics.

2. Related Work

It is common to group authentication methods into the following categories or factors: "something you know" – secrets (methods: passwords, PIN codes, pass phrases, pass images), "something you have" – tokens (methods: keys, magnetic buttons, USB sticks, smart cards), and "something you are" – biometrics (methods: fingerprint, face, iris, voice, gait). Typically, for each authentication method (e.g. fingerprint) there will be several different sensors. This is depicted in Figure 1.

Ranking of authentication alternatives can be carried out by defining a distance metric (product x is n units 'better than' product y). For example, to rank products for the same authentication method, (e.g. face recognition systems), possessing the same attributes, one can make use of standardized samples (e.g. a collection of face pictures). Then, each product can be ranked by combining/ranking values for each of the attributes. Ranking across categories (e.g. comparing a fingerprint sensor and a particular password scheme) is more difficult.

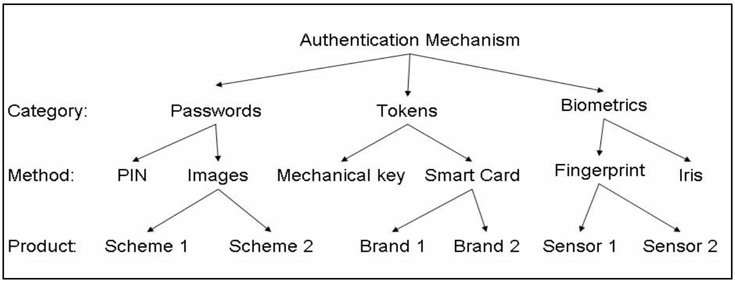

Figure 1: The three diagram of levels of authentication mechanisms

2.1. Comparisons within categories

Within the 'something you know' - category, the password entropies are normally compared. Memorability of passwords is another attribute suitable for comparison (Pond et al. 2000). Within the 'something you have' - category, the encryption algorithms and different attack types are comparison factors (Abott, 2003, Husemann, 1999). For biometrics there is a larger set of comparison factors. There are several well known comparison competitions and tests such as FVC 2000 and FERET (Maio et al. 2002, Phillips et al. 2000). Typically, these use the false acceptance rate (FAR), and false rejection rate (FRR) as comparison factors. The results of the comparisons are presented in the form of receiver operating characteristics, ROC -curves. In addition to FAR and FRR, Mansfield and Wayman use failure to enrol (FTE), failure to acquire (FTA) and throughput rates for comparing performance of biometric devices (Mansfield and Wayman, 2002). Maltoni et al. (Maltoni et al. 2003) suggest that in addition to universality, distinctiveness, permanence and collectability of biometrics, the levels of performance, acceptability and circumvention should also be included as comparison factors for biometric systems.

2.2. Comparisons across categories

O'Gorman (O'Gorman, 2003) compares authentication method across categories. The main comparison factors are security, convenience and cost. With respect to security, he translates discriminative performance to entropy like values that he calls "key space size". In order to compare key space sizes he defines the effective key space size for biometrics as follows

$$k_b = \frac{1}{FMR\ (1)}, \tag{1}$$

where FMR(1) denotes the false match rate for a single verification attempt. Also host-side security and authentication protocols are compared. He also suggests factors like memorability, false non-match rate, enrol, and renew protocols. Cost factors addressed are per-user, infrastructure and administration costs.

NIST special publication 800-63 (NIST 2006) addresses only traditional, widely implemented methods for remote authentication based on secrets. This includes both password schemes and token based authentication where the secret is stored inside the token. Four levels of security are defined. Each level defines requirements with respect to the actual authentication mechanism/product and also its usage procedure. This defines a 4 level ranking method across both password and token based authentication mechanisms.

3. A method for ranking authentication alternatives

We first give a brief overview of our method. Let S be the scenario for which authentication products are to be ranked. The scenario will typically specify security levels for application and data, a user population, physical environment, software, hardware, system requirements and acceptance levels, rules and regulations.

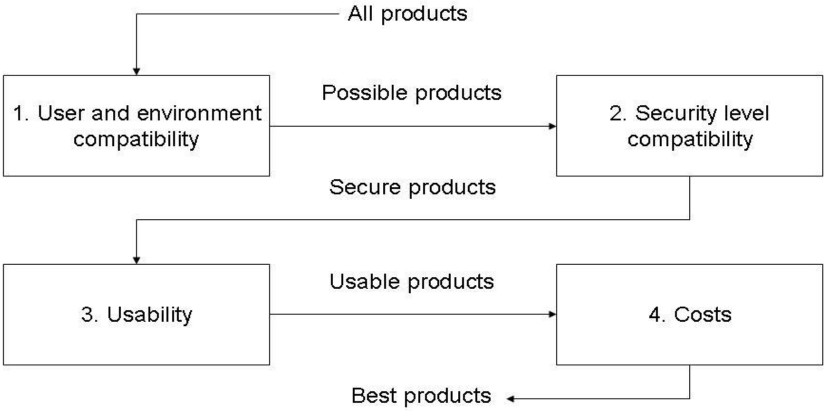

Figure 2: A framework for selecting the most suitable authentication method

Let $A = \{a_1 \dots a_n\}$ be the set of authentication products to be included in the ranking. Each a_i has associated a cost: $Cost(a_i) \in R$ where R denotes the set of real numbers. Then, the output of our method is a sorted list *L* of some subset of *A*, such that $Cost\ (L[j] \leq Cost\ [j+1])$ (for $1 < j \leq length(L)$). Depending on *A* and the particular parameters used, *L* may be empty. Note that in our method, a_i 'is better than' a_j if $Cost(a_i) < Cost(a_j)$.

The overall structure of our ranking method, shown in Figure 2, consists of 4 stages, where each stage may remove some element(s) a_i from *A*, or modify the cost associated with a_i as follows:

1. User and environment compatibility. The usage scenario determines the applicability of authentication methods and products. All authentication methods and products which cannot full fill the requirements are removed from A.

2. Security level compatibility. Remove from A those authentication products which cannot provide security level required by the scenario.

3. Usability. Cost of use (in terms of time consumption) determines the user friendliness. The annual average time consumption of enrolment, verification and error recovery is measured. The sum is compared to acceptability limit. Authentication products that exceed this limit are removed from A. The time spent to authenticate users is not included in the computation of costs in stage 4.

4. Costs. Cost of administration and initial investments rank the remaining authentication products.

Relative to the scenario S specified, the product remaining (in A, if A is not empty) having the lowest cost would be the ‘best’ choice. Below, we give a more detailed description of our method.

3.1. Stage 1: User and environment compatibility

Authentication product must comply with usage and environment related requirements dictated by the scenario. For example, all users in the population must be able to participate in the identity verification process e.g. if the user population contains builders or bricklayers, fingerprint authentication may not be suitable.

We have divided the scenario requirements into three different categories (physical environment, device, and user). As an example, consider a scenario where a woman is using a PDA outside in a public area. In this scenario, the authentication product has to be operational in a ‘non-standard’ (i.e. not a ‘standard’ PC in a normal office environment) and noisy environment (environmental scenario issues), the product has to be suitable for the small size of PDA (device issues) and the woman should be able to provide the required authenticator by her self (specific user issues).

The requirements resulting from the physical usage environment and features of the device may directly exclude some authentication products. Some example of environmental and device requirements are listed to Table 1. Face recognition system and voice recognition system with sensible microphone can be given as examples of the authentication methods which satisfy the surveillance requirements. In scenarios, where a person has to wear gloves, optical fingerprint sensors will not be able to capture data and consequently they must be excluded (Maltoni, 2003). In cases, where authentication is to be carried out using a portable device (e.g. PDA, phone), the size of the device may put size/weight restrictions on biometric sensors.

Modification of A: Remove from A those authentication products that require data or interactions that are incompatible with user population capabilities and those that do not comply with environment and device requirements.

3.2. Stage 2: Security level compatibility

In the second stage, the security level of the remaining products and scenario are determined. The security level dictated by the scenario can usually be determined from applicable laws, regulation and enterprises' rules. The objective of this stage is to remove those authentication products offering a security level below that dictated by the scenario.

Scenario Environment and device requirements	***Derived requirement***
Environmental conditions	Non-standard / standard
Hygiene reasons	Contact / contact less use
Surveillance requirements	Usable in long distance
Size of the device	Small / large sensor
Memory capacity of device	Complex / easy computations

Table 1: Environmental and device limitations versus authentication methods

There exits very few examples on how to map numerical entropy levels to security levels in practice. National Institute of Standards and Technology (NIST, 2006) provides one of them. NIST recommendation is given for a setting when remote authentication of users is done over open networks. Different threat scenarios are defined together with countermeasures. However, these threat scenarios are not taken into account in terms of probabilities as done in our method. For example, in their targeted attack the adversary has no a priori knowledge about the password other than the corresponding user name. In our setting, we consider the probability of social engineering. Similarly to token threats, we consider the possibility that a lost token is found by an adversary. This leads us to the situation that by using our criteria, achieving the NIST security levels may indicate that we have achieved "higher" security level than intended by (NIST, 2006). Many applications are also used in secure, closed networks, which would be covered by our method. In case there are physical or electronic controls increasing the barriers for the attacks this can be taken into account when computing the "entropy" value. This will make the NIST security levels more achievable. Even though the settings differ from each others, we use NIST levels when estimating our security level entropies. We also assume that countermeasures of multi account attacks are implemented.

NIST defines four security levels in "terms of the consequences of the authentication errors and misuse of credentials." They state that minimum online password guessing resistance for the lowest security level, Level 1, should be one in 2^{10} and for Level 2, should be one in 2^{14}. When transforming these to entropy, we get 10 and 14 bits. In Level 3, a one-time-password generator should have at least 10^6 output values, which is about 20 bits. We define our security levels using these entropy thresholds. The

NIST recommendation does not give any numerate values for Level 4. However they state that hardware crypto tokens should be used.

Stalling (Stalling, 2006) uses entropy to define the strength of cryptographic algorithms. Translating his definitions to an authentication setting, we find the entropy thresholds for high (extreme) security to be 56 (128) bits. Consequence levels, described by NIST, can be defined by risk assessments (e.g. Datatilsynet, 2002). Our method uses 6 security levels as illustrated in Table 2. The mapping between consequence levels, entropies, and security levels are shown in Table 2.

Given an assessment of the consequence of illegitimate access, Table 2 specifies minimum entropy requirements on authentication products. The security level of an authentication method is a function of the entropy of the authenticator's search space, H_{auth}, and the difficulty for an attacker to engineer a circumvention attack, H_{circun}. We can formulate this as follows

Entropy	***Our method***	***NIST***	***Consequences***
128<H	Extreme security	Level 4+	Disastrous+
56<H<128	High security	Level 4	Disastrous
20<H<56	Higher medium security	Level 3	Serious
14<H<20	Medium security	Level 2	Moderate
10<H<14	Low security	Level 1	Low
H<10	No security		

Table 2: Security levels versus entropies

$$SL_{a_i} = \min(H_{auth,a_i}, H_{circum,a_i}) .$$

For passwords, the search space is the set of the usable passwords and for tokens the set of different tokens (Statham, 2005). For biometric authentication products, effective key space is estimated using Equation 1. The hardness of the circumvention is computed from estimates of the circumvention probabilities without user awareness. Since circumvention strategies differ for the different authentication factors, we define a separate formula for each category. For passwords, the circumvention hardness is

$$H_{PasswordCircumvention} = -\log_2[\max\{p(soc.eng), p(gue.pw)\}],$$

where *p*(soc.eng) is the probability of user becoming a victim of social engineering, and *p*(gue.pw) the probability of the password being 'easily' guessable, e.g. containing personal information. For tokens, the circumvention hardness is

$$H_{TokenCircumvention} = -\log_2[\max\{p(loss), p(copy)\}],$$

where p(loss) is the probability of a token getting lost and consequently ending up in the hands of the attacker and p(copy) the probability of token being unauthorized copied. For biometrics, the circumvention hardness is

$$H_{Biometrics\,Circumvention} = -\log_2[\max\{p(get.original), p(forge)\}],$$

where p(get.original) is the probability of adversary getting an original biometric sample and p(forge) is the probability of forgery resulting in a usable biometric sample. Note that several of the above probabilities can be computed from incident and loss statistics. In the case of independent multi-factor authentication (e.g. token and password), the security level of the composite authentication product is the sum of the authenticator's individual security entropy (Bhargav-Spantzel et al. 2006).

Modification of A: Remove from A those security products having a lower security level than that dictated by the application.

3.3. Stage 3: Usability

A practical authentication product should be relatively quick to use. Users should not feel that the verification procedure increases their work load or distracts their work. Therefore we are interested in the time of the actual use of the authentication system. In the third stage, we discard the authentication products that result in an unacceptably time consuming authentication process. The authentication product usability is computed from the estimated annual time consumption (in hours or in minutes if preferred) per user by the different authentication activities as follows

$$Time(a_i) = t(enrol) + t(trans) + t(renew) + t(delayhum) + t(delaysys)$$

where the summands are time required for enrolment, identity verification, renewing authenticator, and delayed transaction times when we have a human failure and a system failure. Note that to compute the above, we need estimates of both the number of, the times and duration for each of the authentication activities.

Usually, the identity verification time is the total time that the person uses from beginning the authentication procedure to receiving an accept/reject notification. However, if the authentication can be done without interfering in users' normal activity, then we set identity verification time to zero. In some cases, this may be the case for biometrics such as gait and keystroke dynamics. Also all continuous authentication methods have transaction time as zero. Some implementation of face recognition and speaker verification can be classified as continuous authentication methods. The time for renewal of an authenticator means the time which is spent, starting at the point when user contacts the administrative personnel or other service in order to get a new authenticator and ending at the point, when a new authenticator is usable. By human failure we mean errors, which occur when the person is not able to produce a verification sample matching the template sample. These kinds of errors are: mistyping of password, misplacing tokens and failure to acquire biometric sample. By system failure in login procedure we mean errors which occur when the system rejects a valid authenticator: right password, right procedure with tokens, and false non match or false rejection decisions for biometrics.

Modification of A: Remove those products that have a time consumption exceeding the scenario acceptance threshold.

3.4. Stage 4: Cost of infrastructure and administration

In the last part of the framework, the costs of infrastructure and administration for each authentication method are evaluated. The computation of these costs is based on the suggestions by O'Gorman (O'Gorman, 2003). Infrastructure costs occur when a new system is built, and they can be estimated as follows

$$Inf(a_i) = c(equip) + c(sotf) + c(imp) + c(ins) + c(enrol) + c(stor),$$

where the summands are cost of equipment, software, implementation, installation, first enrolment, and template storage. Enrolment of the users, which take place when building a new system, is considered to belong to infrastructure costs. When the system is up and running, occasional enrolment of the new employees is considered to be part of the administrative costs. Administration costs are computed as follows

$$Adm(a_i) = c(enrol) + c(renew) + c(term) + c(lis) + c(main),$$

where the summands are cost of user enrolment, authenticator renewal, termination of account and authenticator, software licence, and equipment maintenance. The costs are in currency units per year. The cost of each authentication product can then be computed as follows:

$$Cost(a_i) = Inf(a_i) + Adm(a_i)\cdot$$

We can now rank the remaining authentication products in A (assuming A is non-empty) by constructing a list from A by sorting its elements on ascending costs. The first element in the list would then be the 'best product' that complies with all scenario requirements (that we have considered).

4. A practical example

Our case study considers a hospital trust having several hospital units and a total of 11000 employees. The primary application is the authentication of medical personnel having access to the electronic patient records. Terminals used for the access are either in locked offices or under supervision. Each office has an individual key. The network is closed and cabling is secure. A case study is based on the findings in (Helkala, 2007). However, the statistic used (Tables 3-7) is not taken from any real health service scenario. It is based on estimates to demonstrate the use of the method.

Our case is based on the two-factor authentication where the first authentication mechanism is the mechanical key that is needed to open the office door. The second mechanism is used for the log-on. We select the "best" product for the log-on among the set of products A={Standard password, Fingerprint, One-time password}

- SPW: a standard password system
- FP: DigitalPersona U.areU 4000 fingerprint sensor and VeriFinger recognition algorithm
- OTPW: one-time password generator without a PIN code: RSA SecurID®.

Stage 1. User and environment compatibility. The users, environment and device requirements do not cause problems for any of these products, thus the set of products remains same, A={Standard password, Fingerprint, One-time password}.

Stage 2. Security level compatibility. Each office has an individual mechanical key. Keys are anonymous in sense that room numbers cannot be determined by inspecting the key. However there are initials of the hospital unit on each key. We assume there are 500 office doors in each hospital unit. It takes 2 days for the lock to be changed after the key is lost. Finding the right door would need a "brute force" attack, meaning that an adversary has to try each door in order to find the right one. We assume that the adversary can try 15 doors during 2 working days before he will get caught. In a worst case scenario all keys lost are found by an adversary. We find the probability of finding a door to be *p(find door)*=15/500=0,03. The minimum entropy is therefore 5 bits.

The password policy allows only passwords which contain characters upper and lower case letters, digits and special characters. The minimum password length is 8 characters. No account lock down after a certain number of trials is implemented to the system. The personnel have obtained an understanding of password handling, importance of the medical record's confidentiality and integrity, social engineering attacks, security such as writing the password downs, loaning the personal keys, passwords and other security items, etc. Also the guidance on how to generate strong

	Door Key	***SPW***	***FP***	***OTPW***
Search Space		Characters: 8	FMR: 0,001%	Digits: 6
P(sos.eng)	-	0,0009	-	-
P(get.original)	-	-	-	-
P(loss)	-	-	-	0,00009
P(gue.pw)	-	0,002	-	-
P(forge)	-	-	0,0009	-
P(copy)	-	-	-	-
Entropy	5	9	10	13
Sum Entropy		**14**	**15**	**18**
Secure product H>14		**Yes**	**Yes**	**Yes**

Table 3: Security level compatibility.

passwords has been given. The effect of the security education is checked regularly by anonymous questionnaires and on-site inspections. A recent test revealed that the user population had 22 easily guessable passwords such as user name combined with birth dates. A social engineering experiment found that 10 persons gave away their password. These give the probability of guessable password to be *p(que.pwd)*=22/11000=0,002 and the probability of social engineering to be *p(sos.eng)*=10/11000=0,0009.

The fingerprint sensor DigitalPersona U.areU 4000 can be fooled by using 3 dimensional silicon fingerprints (Gravnås, 2005), but in order to manufacture the forged fingerprint, the user's voluntarily help is needed. The forgery of involuntary user's fingerprint may be harder. We assume that the same persons who were fooled by social engineering attack also would be fooled by this. So the probability of forgery would be *p(forge)*=10/11000=0,0009.The residual fingerprints on the sensors cannot be activated, thus the probability of getting an original sample is negligible.

It is extremely hard to "copy" a one-time password generator and synchronize it with the system. Therefore the probability of coping is defined to be negligible. The loss of these generators after the security education is one generator per year. The generators all look identical, only the hospital logo indicates that they belong to hospital personnel. Again, we assume that the lost one-time password generator is found by an adversary. In order to find the real user name, the adversary needs to use a brute force attack. Therefore the probability of loss is *p(loss)*=1/11000=0,00009.

The security levels based on previous estimations are shown in Table 3. The set of product remains the same, A={Standard password, Fingerprint, One-time password}.

	SPW	*FP*	*OTPW*	1:Estimated by authors 2:Estimated, GUC sys. 3:Estimated, (Yan et al. 2004) 4:(NEUROtechnologija, 2008b) VeriFinger
Working days per year	235	235	235	
Auth. sessions per day	8_{1}	8_{1}	8_{1}	
One transaction (sec)	10_{1}	5_{1}	10_{1}	
One renew	4min$_{2}$	0_{1}	7d$_{1}$	
Nr of renews (/user/year)	16	0	1/11000	
Human error (%)	$60_{1,3}$	5_{1}	5_{1}	
System error (%)	5,56$_{2}$	1_{4}	5,56$_{1}$	

Table 4: Needed estimations for usability computations.

	SPW	*FP*	*OTPW*
Enrolment (min/year)	2	2	2
Transaction (min/year)	313	157	313
Renew (min/year)	64	0	1
Human delay (min/year)	188	8	16
System delay (min/year)	17	2	17
Threshold: 1,6 min/day	**2,5 min**	**0,7 min**	**1,5 min**
Usable products	**No**	**Yes**	**Yes**

Table 5: Usability

Stage 3. Usability. In Table 5, the usability variables are listed with estimation of the time (minutes) used for each part in a year per user. These values are computed by using information given in Table 4. Based users' opinion, the acceptable authentication time is 12 sec per session. Taking into account that 8 authentication sessions (see Table 4) are needed in daily basis, we will get the acceptability threshold $12\,\text{sec}\cdot 8 = 1{,}6\,\text{min}$. After the usability selection, the set of products is A={Fingerprint, One-time password}.

Stage 4. Costs. The last selection is done based on the cost of the products. The price information for the RSA SecurID® was collected from (MISCO.CO.UK, 2008 and StorageMojo, 2007). For the fingerprint recognition system, the information was collected from (NEUROtechnologija, 2008a). The prices with other estimations are shown in Table 6. Table 7 shows the sums of the infrastructure and the administrative costs. Therefore the final list is L={Fingerprint, One-time password}.

In our case study, the best product for the authentication of medical personnel is the VeriFinger fingerprint system. However, the prices we have used might not be prises

Infrastructure costs:	***FP***	***OTPW***	1:Estimated by authors 5: (NEUROtechnologija, 2008a) software and licences for each sensor 6: Automated process 7: (MISCO.CO.UK, 2008) computed from the case which contains a hardware, licences and tokens for 250 users 8: (StorageMojo, 2007) computed from the case which contains maintenance for 250 users 9: nre is the number of equipments
Admin. costs (€/h)	85[1]	85[1]	
Single equip. (€/user)	85[5]	116,03[7]	
Nr of equip.	1:5[1]	1:250[1]	
Software (€/user)	7,078[5]	-	
Implementation	0[1]	0[1]	
Installation	1h/nre[1,9]	15min/nre[7,9]	
Enrolment (min)	5[1]	5[1]	
Template Storage	-	-	
Administrative costs:	***FP***	***OTPW***	
Admin. costs (€/h)	85[1]	85[1]	
Staff turnovers (/year)	1100	1100	
Enrolment (min)	5[1]	5[1]	
Renew	0[1]	30 min[1]	
Nr of renewing	0[1]	1/11000[1]	
Termination	0[1,6]	0[1,6]	
Licence	-	-	
Maintenance	10% Inf.costs[1]	6,42 €/user[8]	

Table 6: The needed estimations for the cost computations.

	Fingerprint	***One-time Pwd***
Infrastructure costs (€)	**468 176**	**1 355 225**
Administrative costs (€)	**54 609**	**78 421**
The total sum (€)	**522 785**	**1 433 646**
The total sum per user (€/user)	**48**	**130**
	The "best" product	

Table 7: Cost of the products.

in the real world because we have not taken into account any discounts which are surely given to enterprises buying large quantity of products.

5. Discussion and Future Work

Comparison of authentication product across authentication categories is a difficult task because of the different features of authentication methods and individual products. O'Gorman (O'Gorman, 2003) was one of the first to compare passwords, tokens and biometrics. One of his contributions is that he defined the effective key space for biometrics to ease security comparisons. We have used O'Gorman's method as an important source for ideas. The goal of out method is to provide a tool for helping decision makers in enterprises and organizations to choose the most suitable authentication product for their usage scenario. Thus, we use the term *selection method* instead of *comparison method*. Currently we are working on new authentication methods that might be applicable in health service environment. The selection method is used to determine if the new authentication methods could replace the existing ones. Our selection method will be used to assess the suitability of the new authentication methods and other products in a special health scenario.

6. Conclusion

Currently, it seems like many enterprises leave authentication product selection to vendors. This may result in poor decisions. We have presented a new method for ranking authentication products relative to a particular usage scenario. The ranking addresses issues such as scenario compatibility, security, usability and costs. Our method has a wide applicability, by allowing widely different authentication products (passwords, biometrics, and tokens) to be compared and ranked. Also, our method simplifies and makes the product selection process more transparent by identifying issues that are important when selecting authentication products.

References

Abott, J. (2003), "Smart cards: How secure are they?", www.sans.org/reading_room/whitepapers/authentication/131.php, (Accessed 18.4.07).

Bhargav-Spantzel, A., Squicciarini, A., Shimon, M., Young, M., Bertino, E., and Elliot, S. (2006), "Privacy Preserving Multi-Factor Authentication with Biometrics", In *Proc. of the 2006 Workshop on Digital Identity Management,* pp. 63-72.

Datatilsynet (The Data Inspectorate) (2002), "Risikovurdering av informasjonssystem med utgangspunkt i forskrift til personopplysningsloven", www.datatilsynet.no, (Accessed 18.4.07)

Gravnås, H. (2005), "User's trust in Biometric Authentication Systems – Do not take end-user for granted", Master's thesis, Master of Science in Information Security, Gjøvik University College, 2005.

Helkala, K. (2007), "Authentication in a Norwegian Health Service (survey report)", In *Proc. of International Symposium on Health Informatics and Bioinformatics*, 2007.

Husemann, D. (1999), "The smart card: don't leave home without it", *IEEE Concurrency,* Vol. 7, pp 24-27.

Maio, D., Maltoni, D., Cappelli, R., Wayman, J. and Jain, A. (2002), "FVC2000: Fingerprint Verification Competition", *Trans. on pattern analysis and machine int.,* Vol. 24, pp 402-412.

Maltoni, D., Maio, D., Jain, A. K. and Prabhakar, S. (2003), *Handbook of Fingerprint Recognition*, Springer, ISBN: 0387954317.

Mansfield, A. and Wayman, J. (2002), "Best practices in Testing and Reporting Performance of Biometric Devices" *NPL Report CMSC 14/02, Version 2.01.*

MISCO.CO.UK Web site (2008), "RSA SecurID® Applicance", www.misco.co.uk/ CONTENT/PROMOS/RSA/SECUREID.ASP?bp=1, (Accessed 26 March 2008).

NEUROtechnologija Web Site (2008a), "Prices for our products", www.neurotechnologija.com/prices.html, (Accessed 26 March 2008).

NEUROtechnologija Web Site (2008b), "VeriFinger, PC-based Fingerprint Recognition Technology", www.neurotechnologija.com/verifinger.html, (Accessed 26 March 2008).

NIST (2006), "Information Security: Electronic Authentication Guideline", *NIST Special Publication 800-63, version 1.0.2.*

O'Gorman, L. (2003), "Comparing passwords, Tokens, and Biometric for User Authentication", In *Proc. of IEEE,* Vol. 91, pp 2019-2040.

Phillips, P. J., Moon, H., Rizvi, S. A. and Rauss, P. J. (2000), "The FERET Evaluation Methodology for Face-Recognition algorithms", *Trans. on pattern analysis and machine intelligence,* Vol. 22, pp 1090-1104.

Pond, R., Podd, J., Bunnell, J. and Henderson, R. (2000), "Word Association Computer Passwords: The Effect of Formulation Techniques on Recall and Guessing Rates", *Computers & Security ,* Vol. 19, pp 645-656.

Stalling, W. (2006), *Cryptography and Network Security,* Prentice Hall, ISBN: 0131873164.

Statham, P. (2005), "Threat Analysis:How can we compare different authentication methods?" *Pres. in Biometric Consortium,* 2005, Hyatt Regency Crystal City, Arlington, VA, USA.

StorageMojo Web site (2007), "RSA Security Price List", storagemojo.com/?page_id=417, (Accessed 26 March 2008).

Yan, J., Blackwell, A., Anderson, R. and Grant, A. (2004). "Password Memorability and Security: Empirical Results", *IEEE Security and Privacy,* September/October, pp 25-31, 2004.