A Continuous LoA Compliant Trust Evaluation Method

Julien Hatin, Estelle Cherrier, Jean-Jacques Schwartzmann, Vincent Frey, Christophe Rosenberger

A Continuous LoA Compliant Trust Evaluation Method

J. Hatin[1,2,3,4,5], E. Cherrier[1,2,3,4], J.-J. Schwartzmann[1,2,3,4,5], V. Frey[6], C. Rosenberger[1,2,3,4]
[1]*Normandie Univ, Caen, France*
[2]*GREYC, UNICAEN, F-14032 Caen, France*
[3]*GREYC, ENSICAEN, F-14032 Caen, France*
[4]*UMR 6072, CNRS, F-14032 Caen, France*
[5]*IMT/OLPS/ASE/NPS, Orange Labs, F-14000, Caen, France*
[6]*IMT/OLPS/ASE/IDEA/PIA, Orange Labs, F-35510, Cesson-Sévigné, France*
{*julien.hatin, jeanjacques.schwartzmann, vincent.frey*}*@orange.com*, {*estelle.cherrier, christophe.rosenberger*}*@ensicaen.fr*

Keywords: mobile device, trust level, continuous authentication

Abstract: The trust provided by authentication systems is commonly expressed with a Level of Assurance (LoA see 3). If it can be considered as a first process to simplify the expression of trust during the authentication step, it does not handle all the aspects of the authentication mechanism and especially it fails to integrate continuous authentication systems. In this paper, we propose a model based on the Dempster Shafer theory to merge continuous authentication system with more traditional static authentication scheme and to assign a continuous trust level to the current LoA. In addition, this method is proved to be compliant with the LoA frameworks.

1 INTRODUCTION

With more and more transactions and services available over the Internet, users are asked to authenticate themselves repeatedly throughout the day. An ideal authentication solution should be safe and non intrusive, to allow various security levels together with transparency, without any authentication burden.

A solution to avoid constant re-authentication is to use Single Sign On (SSO) services like *Google sign on* or *Facebook connect* (Wang et al., 2012). In a SSO environment, the identity verification is not directly performed by the entity providing the service: an identity provider (IdP) both handles the enrollment and authentication steps for this service provider (SP). To get enough confidence in the identity of its customer, the SP requires the IdP for a specific level of assurance. The choice of this specific level is performed according to the risks associated to the service.

The current problem with this kind of authentication system is that the trust is based on pre-established rules corresponding to a maximum security level.

The authentication have to be defined, to take into account the trust level and the actual needs of a continuous authentication mechanism. Based on risks assessment related to governmental transactions and services, Levels of Assurance (LoA, which are defined in section 3) provide the following definition (ISO, 2013):

Definition 1 (Authentication)**.** *Provision of assurance in the claimed identity of an entity*

In this paper, we use definition 1 to formalize authentication as the process of providing elements in order to establish a trust level in the identity of a user. To establish this trust level, authentication factors are required to provide a proof of the user's identity. The user provides a (cryptographic or biometric) proof that she/he owns an authentication factor. Those factors are traditionally divided into four categories (ISO, 2013), grouped into two types: *inherent authentication factors* (*i.e.* biometric and behavioral authentication factors) and *secret based* (ie knowledge ad possession authentication factors) ones. Secret based factors strength is usually evaluated through entropy computation, while inherent factors strength is more generally associated to a false match rate (FMR) (Jain et al., 2004).

Beside this classical categorization, authentication can be split in two other groups, namely continuous authentication and static authentication (Syed

et al., 2014). A static authentication can be either explicit (such as a password entry), or implicit (such as a facial recognition during a session). By contrast, continuous processes can transparently authenticate the user without any time interruption. This can be achieved by behavioral authentication methods like keystroke dynamics (Clarke, 2011), gait recognition (Derawi and Bours, 2013) or even with the pattern usage recorded by the mobile phone (Renaud and Crawford, 2014).

The current LoA frameworks only consider static authentication mechanisms. Introducing continuous authentication will enhance the usability by decreasing the number of explicit authentication during a session.

In this paper, our contribution is twofold: (i) we combine continuous authentication mechanisms with more traditional static authentication mechanisms that fit the current LoA standards; (ii) we translate the current Levels of Assurance into a continuous trust score. We propose to remain compliant with the current Levels of Assurance framework to facilitate the integration of the proposed method to existing services.

This paper is organized as follows. In section 2, we expose the related work in the literature. The Levels of Assurance frameworks are detailed in section 3. We express the wished properties for our model in section 4 and propose a conceptual model in section 5. Then, we simulate an usage scenario in section 6 and discuss the benefits of the proposed framework in section 7. We finally expose future works and conclude in section 8.

2 RELATED WORK

This section presents a brief state of the art of recent authentication mechanisms. To give a scale for the trust level on user authentication and to be able to choose and adapt the authentication factors in function of the SP needs has already been dealt with in the literature.

Based on the mobile phone, the framework proposed by authors in (Furnell et al., 2008) requires the user to reauthenticate himself if the confidence level given by behavioral biometrics sensors decreases to much. This framework called NICA (Non Intrusive Continuous Authentication) is composed of a discrete scale that goes from -5 to $+5$. If a user wants to access a sensitive application, he must reach a sufficiently high level.

In (Crawford et al., 2013), the authors construct an authentication framework to merge both behavioral informations and a classical PIN. The required authentication level could be adapted by setting up a threshold that is dependent of the application the user is trying to access. Even if it merges continuous authentication informations with a more classical authentication method (the PIN code), this framework cannot be translated into a concrete level of assurance.

In (Nag and Dasgupta, 2014) and (Nag et al., 2014), the authors propose to use a genetic algorithm to build a scalable framework to choose the modalities and biometric authentication factors according to the network and the device used to access a service or data. This allows to adapt factors to the perceived risk but again, it is not possible to express an explicit level of assurance within this framework.

In (Helkala and Snekkenes, 2009), the authors describe 6 levels of assurance using the entropy and biometric equivalent entropy defined in (O'Gorman, 2003). The entropy is computed by considering different attacks vectors like an easy to guess password. In this comparison framework, the rule to combine multiple factors is the addition of the entropy of the factors. Continuous authentication is not taken into account and even if this method proposes more levels, the granularity is still limited to six levels.

In (Peisert et al., 2013), the authors propose to gather all information that may help for the authentication of any user and to let a human operator decide when high security is required.

For evident time and cost reasons, this could not be adapted to every authentication systems, where users need to be massively and immediately authenticated.

To cope with the usual lack of granularity and to take the continuous authentication into account, we propose to construct a model for the levels of assurance and to use the Dempster Shafer theory in order to deal with the uncertainty on the user's identity.

3 THE LEVELS OF ASSURANCE

Historically, the first authentication assurance levels framework has been published by the NIST in (United State gouvernement, 2006). This framework has recently been normalized in (ISO, 2013). Those recommendations, originally intended for governmental and industry services, are now considered as the standard authentication framework for Internet services (ISO, 2013). Multiples frameworks have been published, since, by other governmental services at a worldwide scale. We can mention:

- EAG (USA) normalized in ISO 29115 (ISO, 2013)
- eID Interoperability for PEGS (Europe) (Europe, 2007)

- National e-Authentication Framework (Australia) (Australian governement, 2009)
- e-Pramaan (India) (Government of India, 2012)

Even if these frameworks share the same number of levels (4 to 5), the descriptions of these levels are rather various, based on specific terms for each framework. This could therefore lead to a misinterpretation or at least a misunderstanding of their common roots. For illustration purpose, we present in table 1 and table 2 the Australian and European frameworks, respectively. For more details in the correspondence between the levels of the above frameworks, we refer the reader to the reference (Jøsang, 2013).

However, as pointed out when comparing table 1 and table 2, the authentication level for a given risk analysis depends on the considered framework. We notice that the European framework is a lot more restrictive than the Australian one. The Indian and ISO frameworks propose a mapping close to the Australian one.

Table 1: Indicative application sensitivity level in Australian framework.

	Consequences				
Likelihood	Insignificant	Minor	Moderate	Major	Severe
Almost certain	Nil	Low	Moderate	High	High
Likely	Nil	Low	Moderate	High	High
Possible	Nil	Minimal	Low	Moderate	High
Unlikely	Nil	Minimal	Low	Moderate	Moderate
Rare	Nil	Minimal	Low	Moderate	Moderate

Table 2: Indicative application sensitivity level in the European framework.

		Impact of damages				
	Likelihood	Very High	High	Medium	Low	Negligible
Risk i	Almost certain	(1)	(1)	Level 4	Level 3	Level 3
	Likely	(1)	Level 4	Level 3	Level 3	Level 2
	Moderate	Level 4	Level 3	Level 3	Level 2	Level 2
	Unlikely	Level 3	Level 3	Level 2	Level 2	Level 1
	Rare	Level 3	Level 2	Level 2	Level 1	Level 1
(1): Not applicable to remote authentication over open networks.						

Another inconsistency between those frameworks is the technological choice recommended for highly secured applications. In the case of western frameworks, a tamper-resistant element like a smartcard is mandatory, while in the Indian framework a biometric reference is required to reach such a level.

The development of these cultural differences can be explained by the different authentication technologies adopted in those different regions. In western countries and especially in Europe, Public Key Infrastructure and smartcard authentication are widely deployed. They are also considered to be the state of the art in the computer security area, whereas in the meantimes, in India, citizens are massively enrolled with biometric ID through the Aadhaar project. This means that, even if all frameworks agree that all factors are not of equal strength, they do not consider the same class of authentication factors as the strongest. This also denotes freedom of interpretation and subjectivity associated with the different Levels of Assurance.

Finally, there is no linearity in the levels scale. For instance, there exists a strong gap between LoA2 and LoA3. This gap has a consequence for usability and security: in order to select an appropriate authentication procedure, a service must either choose LoA2 to maximize usability or LoA3 to maximize security. An intermediate step should be reachable to allow a trade-off between usability and security. In the following section, we expose the wished properties for our authentication framework, based on the previous remarks.

4 AUTHENTICATION SYSTEM PROPERTIES

In this section, we focus on the properties required to define an authentication system. The proof of the user identity is the core element in the different authentication frameworks. This proof relies on an authentication factor.

The confidence in the proof provided by an authentication factor will legitimate its usage. Recall that the highest confidence degree, namely LoA4 relies either on a biometric template in the Indian framework, or on tamper resistant equipments in the European and ISO frameworks.

Based on our state of the art in section 2, we propose to consider the following properties that an authentication framework must fulfill to combine both static authentication factors and continuous authentication elements.

4.1 Neutral State

In this state, the trust level should not express any trust nor distrust about the user: this can be the case when no proof has been given yet.

4.2 Correlation Between Factors

Two factors can be correlated. In current models, in order to perform a two-factor authentication and to avoid the correlation between factors, the second factor is required to be of another category (among biometrics, knowledge, behavior or possession). If this requirement correctly expresses what non-correlated factors are, it does not explain what partially correlated proofs are. Two proofs can be correlated for two reasons:

- The acquisition and/or the transport of proofs are correlated, *e.g* using the same channel.

- The factors are correlated, *e.g.* using a PIN code for a payment requires to have the smartcard: therefore a PIN code and a smartcard cannot be considered separately.

4.3 Ordered Proofs

Some authentication proofs are more secure than others. For instance a PIN code as a lower entropy than a password. This allows to classify proofs according to their strength. In addition, trust in provided proofs can be combined to increase the trust level. This implies that Levels of Assurance consider multi-factor authentication and is a property that needs to be considered in our authentication framework.

4.4 Nested Proofs

A nested proof is not directly presented to the verifier. Instead, the verification is performed through another proof verification. A simple example is the smartcard authentication with a PIN code. According to (Europe, 2007), this is at the highest level of security (LoA4) and is considered as a two-factor authentication. It is important to recall that the PIN verification is performed in the card. This means that there is no way for the final verifier to know if the PIN has been correctly entered on the card or if the card has been compromised. The verifier then accepts this second factor because he trusts the first one as strong enough to verify the second. There is a dependency between the PIN verification and the card. This dependency should be expressed in a formal way and authentication relying on the couple smartcard, PIN should not be considered as a two-factor authentication when considering the possible weakness of the verification protocol.

4.5 Continuous Authentication

Continuous authentication provides additional information about the user. Behavioral authentication systems are based on the fact that most people exhibit habits (Zheng and Ni, 2012). This permits to construct a model for each individual. Continuous authentication evaluates a coherence with this stored user behavior model. Continuous authentication based systems are generally less intrusive than physiological biometric systems (such as fingerprints, face, iris...), but present lower performances. Therefore, for low security applications, continuous authentication could be sufficient to authenticate user.

4.6 Trust Erosion

Once a session has been opened using a static authentication factor (like a password), the user is authenticated with a certain trust level. If the user leaves, the session is still open with a constant amount of trust, equal to the initial level. Conversely, in our system, the confidence offered by a proof should decrease with the time. We call this phenomenon *trust erosion*. This erosion could be lessened, according to the continuous authentication score.

4.7 Trust Representation

Trust should be represented with a value on a continuous scale and not with four or five levels. An authentication framework must be able to give a score and compare two different authentication methods when they combine multiple authentication factors. For this purpose, the framework must be able to propose a simple and efficient way to evaluate a proof and to combine several proofs.

In the next section we present an implementation respecting the previously exposed properties.

5 A USABLE MODEL FOR AUTHENTICATION

The main problem in authentication is: it is impossible to be completely sure that the current user is who he/she pretends to be. Indeed, the user may be both in the state 'genuine user' and in the state 'attacker'. This uncertainty can be handled with the Dempster Shafer theory (Shafer et al., 1976). In the following, we recall the principle of this theory.

5.1 Dempster Shaffer Theory

Authentication system are usually based on probabilistic scenarios. In order to authenticate a user, the system tries to answer the question: is this the claimed user? There is a set of possible solutions to this problem: $\Theta = \{g, a\}$ where g stands for genuine and a stands for attacker. If we apply classical probability P to this problem, we have a solution where $P(g) = 1 - P(a)$. This means that if a user is not genuine then he/she is automatically an attacker. There is no possible doubt nor uncertainty.

For this reason, classical probabilities are not able to correctly manipulate the trust level related to authentication. We would like to have a more realistic vision where the estimated trust in the user identity could include uncertainty about the user state (genuine or attacker).

The Dempster Shafer belief theory (Shafer et al., 1976) permits to take into account this state of uncertainty. The belief theory could be seen as an extension of classical probability theory by allowing the explicit expression of ignorance.

Let θ be a frame of discernment. This set contains a list of exhaustive and mutually excluding elements.

For instance $\theta = \{g,a\}$. The propositions $\wp(\theta)$ will be all the possible parts of θ including the empty set $\emptyset$. In our example, $\wp(\theta) = \{\emptyset, g, a, \{g,a\}\}$.

When a sensor performs a measurement about a state X, it assigns a basic belief assignment, also called a belief mass function or just a mass, $m(X)$. This mass verifies the following equation (Shafer et al., 1976):

$$m(\emptyset) = 0 \qquad \text{and} \qquad \sum_{X \in \wp(\theta)} m(X) = 1 \tag{1}$$

From there, the belief function $Bel()$ and plausibility function $Pl()$ are defined as:

$$Bel(A) = \sum_{B|B \subseteq A} m(B) \tag{2}$$

$$Pl(A) = \sum_{B|B \cap A \neq \emptyset} m(B) \tag{3}$$

Equation (2) represents the lower bound and equation (3) the upper bound of expectation of state A. For more details about the Dempster Shafer theory, the reader may refer to (Shafer et al., 1976).

5.2 Definition of a Proof

Using the Dempster Shafer theory, we could define the possible states of an authentication result as $\theta = \{g,a\}$ where g represents a genuine user and a represents an attacker. This gives $\wp(\theta) = \{\emptyset, g, a, \theta\}$. The presentation of a proof is considered as a measure realized on the user identity. The proof is a combination of the protocol and the modality as shown in figure 1. If the user presents a successful proof of his/her identity, like a correct password for instance, then the measure $m(g)$ will be considered as a positive contribution.

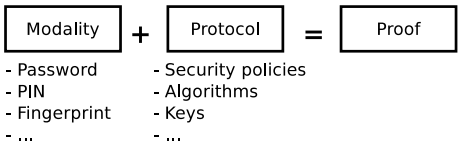

Figure 1: Proof construction.

However, since there is no perfect authentication factor and consequently no perfect proof, a successful proof presentation results in two measurements: $m(g)$ and $m(\theta)$. We propose to present this measurement as a couple:

$$\alpha = (m(g), m(\theta)) \tag{4}$$

We remark that the three masses $m(a)$, $m(g)$, and $m(\theta)$ are linked together and this is why only two values are needed to define α. Indeed, the mass of the attacker element can be deduced from equation (1): $m(\theta) = 1 - m(g) - m(a)$.

5.3 Trust Level Computation

Combining proofs provides a range of possible values between $Bel(g)$ and $Pl(g)$. A solution to take into account the continuous authentication score C and provide a single value, is to use a pignistic function (Smets and Kennes, 1994). This function permits to convert the range to a concrete scalar value with the help of the continuous authentication. It could be seen as placing a bet on the trust score from Bel(g), Pl(g) and the continuous authentication score. We calculate the trust level L with the formula below:

$$L = Bel(g) + C \times (Pl(g) - Bel(g)) \tag{5}$$

A neutral value for the trust value is $\frac{1}{2}$. In case of distrust, the value decreases under $\frac{1}{2}$ and increase over $\frac{1}{2}$ in case of trust. This permits to express trust and distrust in function of the provided proofs on a continuous scale between 0 and 1.

5.4 Confidence in a Proof

The basis of this authentication model is how to evaluate the masses $m(g)$ and $m(\theta)$ assigned to a proof. To calculate those masses, two criteria must be observed:

- The inherent strength of the proof
- The correlation with previously exposed proofs

We give more details about these two criteria thereafter.

5.4.1 Inherent Strength Calculation

The proof strength S is very subjective by nature. It depends on the proof robustness and on other properties like the ability to detect if one modality has been stolen for instance. It is usual and convenient to classify proof strength into three categories going from weak to good. The hypothesis is made that proof strength can be classified into three categories: Weak, Medium, Good.

A numerical value could then be given to every single category for S according to figure 2. We only selected the upper half ($[\frac{1}{2}; 1]$) because we assume that an authentication proof should always add more trust than doubt in the identity of the user.

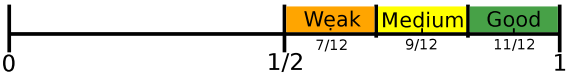

Figure 2: Strength of the proofs.

5.4.2 Correlation

The inherent strength is weighted in function of its correlation with previously exposed proofs. Correlation is determined by the correlation *Corr* between the modality (same factor category) and the protocol used (same media, ...). We attribute the values 0, $\frac{1}{2}$ or 1 to *Corr* if the modality or/and protocol of a proof is correlated with previously exposed proofs or not.

We then could attribute a mass $m(g) = S \times (1 - Corr)$. In case of a failed authentication (lost password, blocked pin...) a mass could be assigned to $m(a)$.

5.5 Strength of a Nested Proof

In case of a nested proof, the verification of the first proof is necessary to observe the second proof. We estimate that the strength of the nested proof depends on the strength of the proof that handles it. We could formally write $S = S_{handler} \times S_{nested}$. A correlation appears between the proofs. We attribute the value of $\frac{1}{2}$ to the correlation.

5.6 Proofs Combination

If proofs are independents, the Dempster Shafer combination rule can be applied. Then the combination can be defined by $\alpha_{1,2} = \alpha_1 \oplus \alpha_2$ for the evaluation of proof 1 combined with proof 2, where α values are defined in equation (4). If the proofs lead to successful authentication results, we obtain the following values (Shafer et al., 1976):

$$\begin{cases} m_{1,2}(g) = m_1(g) + m_2(g) - m_1(g)m_2(g) \\ m_{1,2}(\theta) = m_1(\theta)m_2(\theta) \end{cases} \quad (6)$$

5.7 Trust Erosion

Trust erosion could occur on account of inactivity. The resulting confidence about state g could be calculated as follows:

$$^{\gamma}\alpha = \begin{cases} m(g) = \gamma \cdot m(g) \\ m(\theta) = \gamma \cdot m(\theta) + (1 - \gamma) \end{cases} \quad (7)$$

For instance, γ could be determined by using the following equation from (Crawford et al., 2013) : $\gamma = \frac{1}{(t_{now} - t/\rho)^r}$, where r is the rate of ageing. Increase r will imply that the trust in a proof will decrease faster. ρ is the granularity: a coarser granularity permits to regroup events that happen simultaneously.

5.8 Neutral Element

A neutral element could be introduced to define the initial state where no proofs are available yet. This particular value $\alpha = (0, 1)$ represents a total ignorance about the genuine and attacker elements of the user. In the following section, we simulate a usage scenario and attribute values to the discrete LoA in order to compare both approaches.

6 FRAMEWORK UTILISATION

To demonstrate the feasibility of our model, we proceed in two steps. In a first time, we translate the current ISO LoA levels into numerical values, to obtain thresholds. Then, a trust level is computed by merging continuous authentication with an explicit authentication. This is illustrated through a usage scenario and shows how this model can be used in real conditions, where a LoA framework is currently used.

6.1 Establishing LoA Threshold

The proposed model could be used to compute values for the LoA. This enables to give a numerical threshold for the discrete levels. In this paper, we choose to follow the ISO framework, but it can be transposed to any other LoA framework. Our system relies on the authentication factors presented in table 3, classified following the categories exposed in section 5.4.1.

Table 3: Strength for each authentication factor.

Weak	Medium	Good
Password	Fingerprint	Smartcard
PIN code	OTP	

The LoA do not take into account continuous authentication. To find an equivalent to the current levels, we set $C = 0$, since it is not taken into account in the current frameworks. The $m(a)$ value is always considered to be 0. This leads to $m(\theta) = 1 - m(g)$. The strength of the different factors are set according to figure 2. The calculation is detailed in the appendix in order to give a concrete example to the reader. Results are exposed in table 4.

We now illustrate the model practicability through a usage scenario.

Table 4: Thresholds equivalent to the LoA.

Level	LoA1	LoA2	LoA3	LoA4
Equivalent score	0.58	0.75	0.89	0.96

6.2 Usage Scenario

The figure 3 shows a simulation of the evolution of the trust level during an afternoon. The continuous authentication data are extracted from the MIT Reality Mining Dataset collected in (Eagle and Pentland, 2006). For sake of clarity, the continuous authentication score is computed by evaluating the probability of launching an application at a given time. If a user called Alice launches a usual application at a usual time, then the continuous authentication score increases. Even if this is a non optimal solution, this is sufficient to present our model. Developing a new continuous authentication system is out of the scope of this paper.

At the beginning of the simulation (point A on the figure), the continuous authentication score is under 0.5. We could deduce from this score, that Alice has an unusual behavior at this time. In the beginning of the afternoon, she wants to access her professional mail account: she needs to enter her password at point B. As a consequence, the trust level is increased and reaches the LoA2 threshold. Because the system observes usual behaviors, the continuous authentication grows up with the time (point C). With this continuous authentication score increase, the trust level is increased. At 13:30, Alice wants to access her bank account. Even if the continuous authentication score has grown up, it is still too low to reach the LoA3 level. A One Time Password is entered by Alice and the trust level reaches a sufficiently high score, so she can access her bank account (point D).

7 DISCUSSION

We observe, the more factors there are, the less the trust level is influenced by the continuous authentication score. This permits to counteract the effect of a low continuous authentication score by increasing the number of required authentication modalities. Conversely, a high continuous authentication score, ensures a good usability for the system by requesting less authentication factors.

An effect of the proposed theoretical model is that the trust based on a multi-factor authentication decreases faster than the trust based on a single-factor authentication. This is due to the trust erosion operation that is independently applied on each provided proof. Take the continuous authentication into account would counteract for this effect and maintain the authentication level through the time.

If the trust level L permits to always have a measurement depending on the continuous authentication, when there is no proof at all, the overall level is only given by the continuous authentication because of equation (5).

The improvements proposed by our solution for evaluating the trust in the authentication are summarized in table 5.

Table 5: Authentication systems properties defined in section 4.

Framework	Neutral state	Correlation between factors	Ordered proofs	Nested proofs	Continuous authentication	Trust erosion	Trust representation
ISO			✓				
eID			✓				
NeAF	✓		✓				
ePramaan	✓		✓				
(Furnell et al., 2008)	✓				✓		
(Crawford et al., 2013)	✓				✓	✓	✓
(Nag et al., 2014)		✓			✓	✓	
(Helkala and Snekkenes, 2009)			✓				
(Peisert et al., 2013)	✓	✓	✓	✓	✓	✓	
Our	✓	✓	✓	✓	✓	✓	✓

8 CONCLUSION AND FUTURE WORK

In this paper, we presented a computation model for the LoA based on the Dempster Shafer theory. This permits to merge a continuous authentication system with more traditional static authentication scheme and to assign a continuous trust level to the current LoA.

The performances of the proposed model directly depend on the performance of the inherent continuous authentication system. Of course in terms of security and usability, since the trust accorded to the continuous authentication system depends on its performances but also regarding privacy. A continuous authentication system requires to collect personal data. For this reason, in further works, we intend to build a privacy protecting continuous authentication mechanism that can be easily integrated within this framework. The final goal is to implement a complete authentication framework in a real world scenario.

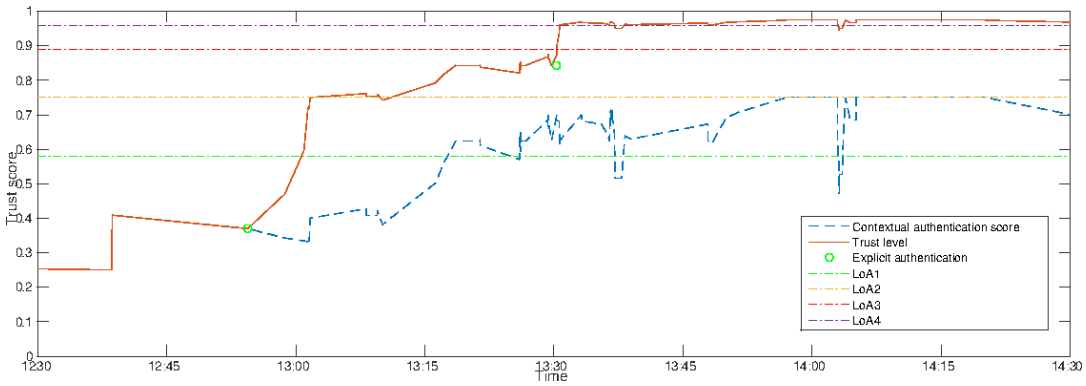

Figure 3: Trust evolution through the time for a given continuous authentication.

9 AKNOWLEDGEMENT

The authors would like to thank Orange and the ANRT for the financial support.

REFERENCES

Australian governement (2009). National e-authentication framework.

Clarke, N. (2011). *Transparent User Authentication Biometrics, RFID and Behavioural Profiling*. Springer.

Crawford, H., Renaud, K., and Storer, T. (2013). A framework for continuous, transparent mobile device authentication. *computers & security elsevier*.

Derawi, M. and Bours, P. (2013). Gait and activity recognition using commercial phones. *Computers & Security*.

Eagle, N. and Pentland, A. (2006). Reality mining: sensing complex social systems. *Personal and ubiquitous computing*, 10(4):255–268.

Europe (2007). eid interoperability for pegs. Technical report, iDABC European eGovernement services.

Furnell, S., Clarke, N., and Karatzouni, S. (2008). Beyond the pin: Enhancing user authentication for mobile devices. *Computer Fraud & Security*.

Government of India (2012). e-pramaan: Framework for e-authentication. Technical report, Ministry of Communications and Information Technology.

Helkala, K. and Snekkenes, E. (2009). Formalizing the ranking of authentication products. *Information Management & Computer Security*, 17(1):30–43.

ISO (2013). Information technology security techniques entity authentication assurance framework (iso 29115).

Jain, A. K., Ross, A., and Prabhakar, S. (2004). An introduction to biometric recognition. *Circuits and Systems for Video Technology, IEEE Transactions on*, 14(1):4–20.

Jøsang, A. (2013). Identity management and trusted interaction in internet and mobile computing. *Information Security, IET*.

Nag, A. K. and Dasgupta, D. (2014). An adaptive approach for continuous multi-factor authentication in an identity eco-system. In *Proceedings of the 9th Annual Cyber and Information Security Research Conference*, CISR '14, pages 65–68, New York, NY, USA. ACM.

Nag, A. K., Dasgupta, D., and Deb, K. (2014). An adaptive approach for active multi-factor authentication. In *9th Annual Symposium on Information Assurance (ASIA14)*, page 39.

O'Gorman, L. (2003). Comparing passwords, tokens, and biometrics for user authentication. *Proceedings of the IEEE*, 91(12):2021–2040.

Peisert, S., Talbot, E., and Kroeger, T. (2013). Principles of authentication. In *Proceedings of the 2013 workshop on New security paradigms workshop*, pages 47–56. ACM.

Renaud, K. and Crawford, H. (2014). Invisible, passive, continuous and multimodal authentication. In *Mobile Social Signal Processing*, pages 34–41. Springer.

Shafer, G. et al. (1976). *A mathematical theory of evidence*, volume 1. Princeton university press Princeton.

Smets, P. and Kennes, R. (1994). The transferable belief model. *Artificial intelligence*, 66(2):191–234.

Syed, Z., Banerjee, S., and Cukic, B. (2014). Continual authentication. *Biometric Technology Today*.

United State gouvernement (2006). Electronic authentication guideline. Technical report, NIST.

Wang, R., Chen, S., and Wang, X. (2012). Signing me onto your accounts through facebook and google: A traffic-guided security study of commercially deployed single-sign-on web services. In *Security and Privacy (SP), 2012 IEEE Symposium on*, pages 365–379. IEEE.

Zheng, J. and Ni, L. M. (2012). An unsupervised framework for sensing individual and cluster behavior patterns from human mobile data. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pages 153–162. ACM.