1,2, Pause: Lets Start by Meaningfully Navigating the Current Online Authentication Solutions Space

Ijlal Loutfi(✉) and Audun Jøsang

University of Oslo, Oslo, Norway
ijlall@uio.no, josang@ifi.uio.no

Abstract. Consuming services online is an exponentially growing trend within different industries. A robust online authentication solution to these services is a critical requirement for establishing trust between end users and service providers. Currently, the shortcomings of password based authentication solutions are well recognized. Hence, many alternative strong authentication solutions are being adopted. However, the latter create a siloed model where each service provider dictates a different method (OTP, SMS...) to end users. To resolve these challenges, considerable efforts are being deployed by both academia and industry. However, assessing these efforts is not a trivial task. This paper provides a framework of a well-motivated set of attributes, for categorizing and assessing online authentication solutions. Based on this framework, two main approaches for online authentication are identified and exemplified: LUCIDMAN and FIDO. The results of this research are anticipated to make the navigation of the online authentication solutions space more systematic, and facilitate knowledge transfer between all stakeholders.

Keywords: Authentication · FIDO · LUCIDMAN · Framework

1 Introduction and Background

A growing number of service providers from different industries are moving their businesses online. In online environments, Trusted interactions between users and their chosen service provider depends on robust mutual authentication. Given the exponential growth in the number of interconnected online entities, the challenge of ensuring robust authentication between all these identities is daunting [2]. Currently, password based authentication methods are the most prevalent solution. However, their shortcomings are very well documented and recognized [6]. Indeed, the average end user is unable to handle the growing number of online accounts they own. Hence, end users compromise their own security by resorting to using weak passwords. In order to resolve this issue, the adoption of different alternative strong authentication solutions is quickly growing. By definition, strong authentication mechanisms are cryptographically based and

C.D. Jensen et al. (Eds.): IFIPTM 2015, IFIP AICT 454, pp. 165–176, 2015.
DOI: 10.1007/978-3-319-18491-3_12

have the properties of not being trivially susceptible to common attack techniques, such as credential theft and replay (i.e., phishing), and credential forgery (e.g., password guessing). Strong authentication mechanisms can be implemented in various fashions. They involve at least protocol support and may leverage physical authenticators implemented as platform hardware features, or discrete security tokens, for example: Trusted Platform Module (TPM), biometric sensors, One-Time-Password (OTP) generators, etc. [3]. While strong authentication may seem at first glance to be a perfect solution, the way it has been adopted and implemented presents many challenges. While strong authentication increases the strength of a solutions, it often decreases its usability. Furthermore, It is creating an ever more siloed authentication scene, where each service provider (hereafter abbreviated as SP) governs a separate silo domain with its own name space of user identities, and dictates a different solution to its end users. This silo model is the result of online Identity management being still studied in its traditional way. It dictates that managing user identities should be focused on processes located on the server side, where solutions are optimized for simple management from the service provider point of view [1]. One of the main most recent approaches to identity management which is identity federation. Identity federation comes in many variations but is typically based on the SAML1 standard which is implemented in applications such as Shibboleth, and FacebookConnect. However, Identity federation does not fundamentally solve the problem of identity overload. There will always be different federation domains, because not all SPs will merge their respective user identity domains into a single federated domain.

Having recognized the shortcomings of both password based authentication solutions and the currently implemented strong authentication solutions, further efforts are being deployed by both academia and industry. Currently, the solutions space of online authentication is cluttered, with little resources available to interested stakeholders to assess and analyze the different proposed solutions.

Hence, the aim of this paper is to present a framework/taxonomy of a well-motivated set of attributes, for categorizing and assessing online authentication solutions. Based on this framework, two main approaches for online authentication are identified and exemplified: LUCIDMAN and FIDO. The goal of this research is that the framework would make the categorization and assessment of online authentication solutions more systematic. It will also focus discussions around two oimportant proposed solutions: LUCIDMAN and FIDO. The framework is not meant to present a detailed analysis of each online authentication solution, but rather a basis of comparison and assessment between a set of solutions.

2 Navigating the Current Solutions Space: Proposed Framework

The solutions space of online authentication is a complex one. Different proposed schemes are making claims about the superiority of their solution. Having a systematic way to categorize and assess these solutions, would allow us to

form a well-rounded judgment about each one, and transfer knowledge between teams. For our proposed framework to be as holistic as possible, we identified the below aspects as a point of reference, around which the framework properties are defined:

- **Strength of the solution**.
- **Usability for end users and service providers**.
- **Privacy**.
- **Readiness of adoption by the market**.

Before presenting the framework properties, we would like to formally introduce the definitions of the below concepts.

2.1 Basics First

Digital Identities. A digital entity is a set of digitally expressed attributes of an entity, where one of the attributes typically is a name or identifier for uniquely selecting the entity within a name-space domain. Each entity can have multiple identities simultaneously or at different points in time [2] (Fig. 1).

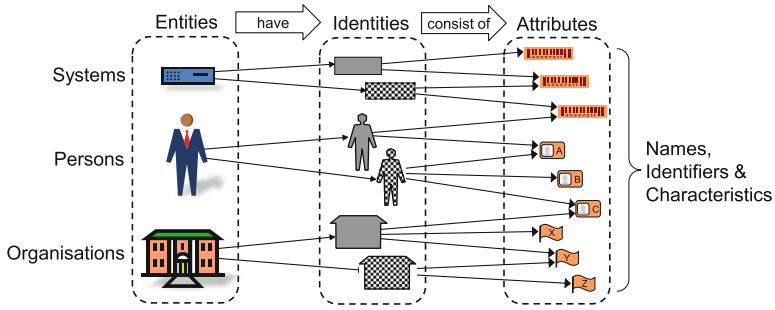

Fig. 1. Relationships between entities, identities and attributes

Entity Authentication. In the context of this research, it is crucial to make a clear distinction between a system entity (client or server) and a legal/cognitive entity (person or organization). This brings into play multiple entities on each side in the client-server model. This is because a requirement for trusted interaction in this scenario is to have mutual authentication between pairs of interacting entities whenever relevant, leading to 4 possible types of mutual entity authentication described in Tables 1 and 2.

For online service access the entity authentication classes [U → S] (user authentication) and [S → U] (cognitive server authentication, defined below) are the most relevant. The importance of these authentication classes emerges from the need for end-to-end security. End-to-end communication between the human user (U) and the server system (S) takes place during online service access. It is

Table 1. Authentication classes for user-side entities

Class	Authentication of user-side entities
[U → P]	User (U) authentication by the service provider (P)
[U → S]	User (U) authentication by the server system (S) (commonly called *user authentication*)
[C → P]	Client (C) authentication by the service provider (P)
[C → S]	Client (C) authentication by the server system (S)

Table 2. Authentication classes for SP-side entities

Class	Authentication of SP-side entities
[P → U]	Service provider (P) authentication by the human user (U)
[P → C]	Service provider (P) authentication by the user client (C)
[S → U]	Server (S) authentication by the human user (U) (here called *cognitive server authentication*)
[S → C]	Server (S) authentication by the user client (C)

therefore pragmatic to require mutual authentication between those two entities. Traditional user authentication can provide [U → S] authentication. It is often incorrectly assumed that traditional server authentication with Browser PKIX[1] server certificates and TLS[2] provides [S → U] authentication, however in reality it does not. This might seem surprising but is in fact easy to understand [9].

According to the X.800 standard, entity authentication is *"the corroboration that a peer entity in an association is the one claimed"* [7]. So in case a phishing victim user intends to connect to https://www.paypal.com, but is tricked into connecting to a phishing website called https://www.peypal.com, then the server certificate claims that the server identity is www.peypal.com which then is correctly authenticated according to X.800. However, something is clearly wrong here, and this indicates that the above definition of entity authentication is inadequate. What is needed is a richer modality of authentication. In the next section, we define three authentication modalities [12].

2.2 Authentication Modalities:

The above definition of entity authentication, and the distinction made between a system and end user, as well as the server provider and the system service provider, gives rise to three main authentication types, which will be called modalities:

– **Syntactic Entity Authentication:** *The verification by the relying entity that the unique name of the remote entity in an interaction is as claimed.* This

[1] PKIX: Public-Key Infrastructure based in X.509 certificates [8].
[2] TLS: Transport Layer Security.

basic form of entity authentication is equivalent to peer-entity authentication as in X.800. Alone, it does not provide any meaningful security and can e.g. not prevent phishing attacks since the relying party is indifferent to the identity of the authenticated entity.

- **Semantic Entity Authentication:** *The verification by the relying entity that the unique name of the remote entity in an interaction is as claimed, and in addition the verification by the relying entity that semantic characteristics of the remote entity are compliant with a specific security policy.* It can be enforced by an automated system e.g. with a white list of authorized identities.
- **Cognitive Entity Authentication:** *The verification by the cognitive relying party that the unique name of the remote entity in an interaction is as claimed, the verification by the relying party that semantic characteristics of the remote entity are compliant with a specific security policy, where the latter is supported by presenting in a user-friendly way identity attributes that enable the cognitive relying party to recognize relevant aspects of the remote entity and to judge policy compliance.* It requires the relying party to have cognitive reasoning power, such as in humans, animals or advanced AI systems. This authentication modality effectively prevents phishing attacks because users recognize the identity of a server and decide whether it is the intended one.

2.3 Frameworks Properties

In this section, we will formally define the attributes upon which our proposed framework is base.

Entity Authentication. User identity management is frequently discussed in the identity management literature, whereas SP identity management is mostly discussed in the network security literature. One added value of our proposed framework is that it consolidates types, because users must manage their own identities as well as those of SPs, in order to be authenticated by server systems on a semantic level, and to authenticate the server systems on a cognitive level. Our proposed framework is aimed at providing adequate security assurance and usability for the management of both user identities and server identities, with the goal of enabling trusted interaction between online entities.

Server Authentication and User Authentication. For mutual authentication, user authentication is implemented on the application layer while server authentication on the transport layer. Since mutual authentication goes between the user and the server, we require server authentication by the user as [S $\rightarrow$ U] which most often is not satisfied in current implementations, and user authentication by the server as [U $\rightarrow$ S] which currently is satisfied in most implementations.

Our proposed framework mandates that server authentication as well as user authentication should be classified as following one of the below 3 modalities:

- *Syntactic.*
- *Semantic.*
- *Cognitive.*

Threat Immunity. A paradox in today's Internet computing environment is that we continue to build vulnerable client platforms while still expecting them to support trusted online interactions. According to PandaLabs' estimates, approximately one third (31.63 %) of the world's PCs are infected with some sort of malware (Q2 2012) of which most (78.92 %) are Trojans [11]. It must therefore be assumed that sooner or later a client platform will be infected by malware, which means that it can never be trusted.

Our proposed framework mandates that the above assumption of having an infected client platform is a reasonable one to hold, and that the solution being studied should be evaluated against it.

Data Authentication. According to the X.800 standard, data origin authentication is *"the corroboration that the source of data received is as claimed"* [7]. This is different from entity authentication because knowing the identity of a remote entity in a session is different from knowing whether the data received through a session with a specific remote entity genuinely originates from that entity. This difference might seem subtle at first glance but it is in fact fundamental for security, as explained below.

Malware infection on client platforms opens up for attacks against data authentication that entity authentication can not prevent. A typical example is the situation of online banking transactions with mutual entity authentication. Even with strong 2-factor user authentication, and users' correct interpretation of server certificates, there is the possibility that malware on the client platform can modify data communicated between client and server platforms. Current attacks against online banking are typically mounted in this way. Such attacks lead to breach of data integrity, but not a breach of entity authentication. The separation between the human/legal entity and the system entity on each side of a client-server session makes it necessary to specify which entity in particular is the assumed origin of data. Our proposed framework mandates that data authentication, just like entity authentication, should be classified as following one of the 3 modalities defined below:

- *Syntactic.*
- *Semantic.*
- *Cognitive.*

User Experience. The success of any online authentication solution requires that many stakeholders come together: end users, service providers, policy makers, device manufacturers and integrators. These stakeholders are what we refer to, in the context of this framework, when we talk about the concept of the ecosystem.

The framework mandates to measure how the identified stakeholders relate to the solution in the present time as well as in the future.

Privacy. By definition, online authentication deals with sensitive and private users data. The strength or the usability of the solution, should not overshadow the way the solution deals with the privacy of users data.

The framework requires the evaluation of the privacy level of the solution. The below list identifies a minimum set of requirements that a solution should be compared against before forming a proper judgment about its privacy level. The items outlined in the table below are defined by the FIDO alliance [4].

- Require explicit, informed user consent for any operation using personal data.
- Limit collection of personal data to the solution-related purposes Only.
- Use personal data only for the solution authentication operations.
- Prevent identification of a user outside of solution operations.
- If biometric data is used, it must never leave the users' personal computing environment.
- Protect user-related data from unauthorized access or disclosure.

3 Two Main Trends

3.1 Overview

Based on this proposed framework, a hypothetically ideal online authentication solution would try to implement the above defined properties up to their highest level. Assuming that we have two solutions A and B, which implement the properties identified in the framework up to the same level, one still would not be able to conclusively state that the two solutions are equally appropriate or not. Indeed, at this point of the analysis, one needs to answer one more question: at which end are these properties implemented (end user system, service provider system, end user device, network etc...)? The research work leading up to this paper, made us involved in evaluating a great number of solutions. By using an earlier, less elaborate version of the proposed framework, we identified two major emerging current trends. They can be contrasted to traditional online authentication solutions where the solutions implementations were focused on the service provider side.

- **Local Centric/Top-Down:** It can be described as a puritarian local-centric approach, where the solution requirements are locked into the implementation. Most often, this approach is implemented with a special hardware which the end user leverages to perform the online authentication ceremonies. In this case, the solution would mandates to the device manufacturers the specific requirements they need to adhere to, as well as how the properties described in the above framework should be implemented. In this approach, the communication protocol used between end users and their service providers remains unchanged.

– **Network Centric/Bottom-Up:** In this approach, the core properties are implemented in the network protocol layer. Any changes to the solution would be invisible to the communicating end user/service provider pair. In this approach end users and service providers need to have dedicated software that would allow them to communicate with the newly implemented protocol.

We will focus in the subsequent section on real life example for each one the two approaches.

3.2 Top Down Approach: LUCIDMAN Basics

LUCIDMAN, which stands for local user-centric identity management, is a principle for identity management. It aims to distinguish between its identity management model and other models that are often called user-centric in the literature. More specifically, LUCIDMAN not only provides robust security and adequate security usability, it is also based on placing technology and computing processes for identity management locally on the user side in the client-server architecture. LUCIDMAN as a solution belongs to the user centric Top/down approach, as it defines a specific set of requirements for its implementation, that are achieved through a user device named OffPAD. The OffPAD represents technology for improving the user experience and for strengthening security, which already makes it user centric. Since the OffPAD is, in addition, located on the user side, it is physically local to the user, and thereby represents technology for local user-centric identity management. The OffPAD can be used for a number of different security services [12], but this article only focuses on how it enables trusted interaction through mutual entity (user and server) authentication as well as data origin authentication. The OffPAD can also support incremental authentication modalities, i.e. syntactic, semantic or cognitive authentication, as shown in Figure [12]. The OffPAD (Offline Personal Authentication Device) described by Klevjer *et al.* [10] and Varmedal *et al.* [12] is an enhanced version of the PAD, where an essential characteristic is to be offline, i.e. not connected to the Internet. Keeping the OffPAD offline strengthens its security by eliminating exposure to Internet threats.The OffPAD supports authentication of both user and SP identities (i.e. mutual authentication) and can in addition support data authentication. A possible OffPAD design is illustrated in Fig. 2.

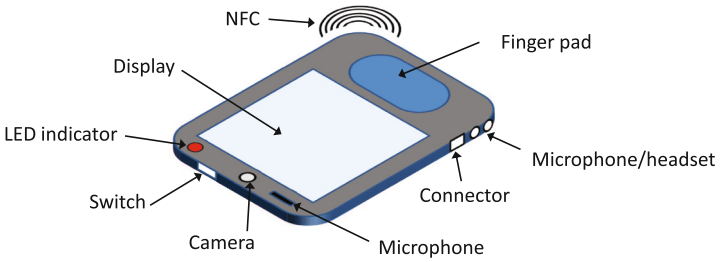

Fig. 2. OffPAD design ideas

The OffPAD enables mutual user-server entity authentication as well as data authentication. Each authentication type is illustrated with a ceremony [2] which is simply a protocol where relevant actions by users and the context environment are included. The 3 ceremonies can be chained and seen as one general ceremony that provides all 3 types of authentication, starting with server authentication, followed by user authentication and finally data authentication. The novelty of this solutions is that it supports trusted interaction even in the presence of malware infected client platforms [12].

3.3 Bottom Up: FIDO Basics

The FIDO (Fast IDentity Online) Alliance is a 501(c)6 non-profit organization nominally formed in July 2012 to address the lack of interoperability among strong authentication devices as well as the problems users face with creating and remembering multiple usernames and passwords. its FIDO is to break with the server centric online authentication solutions, and bring devices to end users which would allow them to have a strong, yet a usable authentication experience. While the analysis of FIDO can be a lengthy one, for the purposes of this paper, we will focus on presenting its approach, and then evaluating its claims using our proposed framework (Fig. 3).

Being an example of a network centric solution, at FIDO's core lies the FIDO client. The FIDO client implements the client side of the FIDO protocols, and interfaces with the FIDO Authenticator abstraction layer via the FIDO Authenticator API. While the FIDO client software implementation will be platform-specific, the FIDO specifications will define the interactions and APIs between the protocol-handling code provided by any FIDO- specific browser extensions, the devices that handle the user authentication and provide the authenticators,

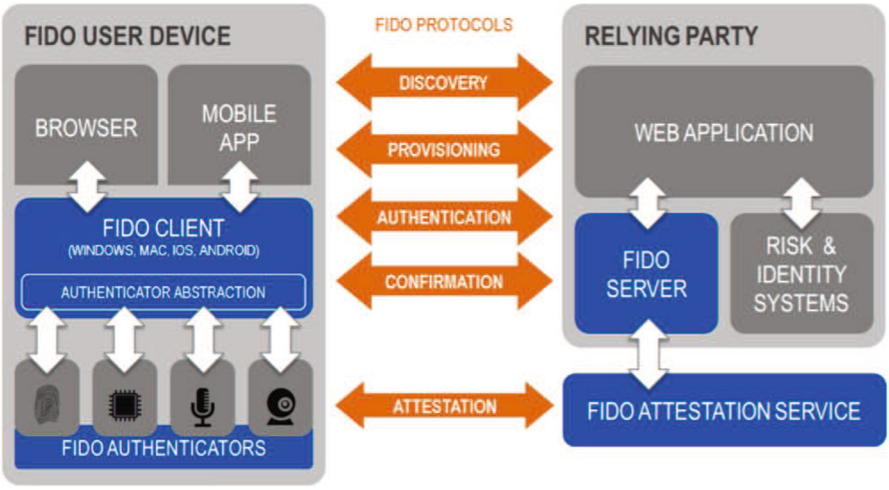

Fig. 3. Overview of FIDO Architecture

and what the user experience should be, in order to ensure as consistent an experience as possible from platform to platform [3]. To better illustrate the concept of FIDO the below figure shall be used.

FIDO Authenticator. A FIDO authenticator is a secure entity, attached to or housed within FIDO user devices, that is remotely provisional with key material by relying parties, and is then capable of participating in cryptographic strong authentication protocols. For example, the FIDO authenticator will be capable of providing a cryptographic challenge response based on the key material thus authenticating itself [3,4].

In order to meet the goal of simplification of authentication capability integration, a FIDO authenticator will be able to attest to its particular type (e.g., biometric) and capabilities (e.g., supported crypto algorithms), as well as to its provenance [3,5].

3.4 FIDO and LUCIDMAN Under the Proposed Framework

The framework presented in this research paper provided us with a structured and systematic way to categorize and compare LUCIDMAN and FIDO. Table 3 below summarizes the results of our analysis.

Table 3. Summary of LUCIDMAN and FIDO analysis

Properties	LUCIDMAN	FIDO
User Authentication	Cognitive	Syntactic
Server Authentication	Cognitive	Syntactic
Data Authentication	Cognitive	Syntactic
Threat Immunity	Yes	No
Privacy	Yes	No
User Experience	End Users	End users, SP, Integrators

4 Discussion and Conclusions

While there are numerous angles from which the above comparison table can be analyzed, the two most notable differences between FIDO and LUCIDMAN are threat immunity and user experience. The network centric approach of FIDO allows for great gains for service providers as well as device manufacturers. Indeed, by just having the FIDO module on the server side once, the service provider can change and allow for different user experiences (fingerprint, voice, ping etc.) without incurring extra costs. Further, device manufacturers also benefit from this approach, as they have a good balance between implementing the user authentication method of their choice, while not having to worry about how to communicate with the service provider. They can hence communicate with

any incremental number of service providers with no extra costs. All of these great benefits for service providers, device manufacturers as well as end users, were achieved because of the top down approach. Since all the properties of the solution are network centric, service providers and device manufacturers have to satisfy a reasonable set of requirements in order to be FIDO compliant. However, as stated in the FIDO security reference, preventing malicious applications from arriving to the FIDO user device are outside the scope of the solution [5]. Given the extent to which user platforms are infected with malware, this raises more than one question. On the other hand, LUCIDMAN through its OffPAD implementation, makes the requirement of protecting the user dwevice, from outside threats as well as platform ones, its main priority. This, unsurprisingly, came at the price of flexibility. Given the specific requirements a LUCIDMAN device needs to adhere to, device manufacturers lose the flexibility to bring their own expertise to the table. At the same time, many service providers might judge the security requirements to be too good for them, and that the cost of obliging end user to acquire a specific type of device in order to consume their services, outweighs the security benefits the OffPAD night brings. Given that having a healthy ecosystem of service providers, device manufacturers and end users is a crucial success factor for the success of any new online authentication solution, the lack of flexibility fo LUCIDMAN might become an inhibiting feature. Interestingly enough, the two approaches put forward different types of strengths and weaknesses. Having used the framework proposed in this paper for their assessment, has allowed us to strategically spot and analyze them in meaningful ways. We argue that LUCIDMAN and FIDO can be integrated or at least take some learnings from each other in order to enhance their current specifications. One way to achieve this would be to for the OffPAD team to work on making their device FIDO ready. This will give the OffPAD a very strong ecosystem and a real chance to gain scale within the market. On the other hand, FIDO would have a device that satisfies strong immunity requirements, and which can be used for service providers that are working in industries that are highly regulated and which require high assurance levels. The Online authentication problem has long been a very challenging one. While different stakeholders are coming up with great innovative ideas to resolve it, all implementation have, so far, failed to cover all aspects of the problem. Breaking the siloed state of online authentication is no easy task. However, we believe that as a security community, academia and industry alike, we should start by breaking our own working silos, and have more knowledge transfer between our solutions. The framework proposed in this paper as well as the two approaches exemplify, have helped us meaningfully evaluate and cross compare two seemingly different solutions, to only arrive at the conclusion that there is much each can learn from the other. If we are to dissolve the silos of online authentication, we are first to dissolve our own learning silos.

Acknowledgments. This work has been partially supported by eurostars project E!8324 OffPAD.

References

1. Jøsang, A., et al.: Assurance requirements for mutual user and service provider authentication. Journal of Trust Management 2(1) (2014)
2. Jøsang, A., et al.: Local user-centric identity management. Journal of Trust Management 2(1) (2015)
3. Alliance, F.: Draft reference architecture (2014). http://fidoalliance.org
4. Alliance, F.: Fido Alliance Whitepaper: Privacy Principles (2014). http://fidoalliance.org
5. Alliance, F.: Fido Security Reference (2014). http://fidoalliance.org
6. Florencio, D., Herley, C.: A large-scale study of web password habits. In: Proceedings of the 16th International Conference on the World Wide Web, pp. 657–666. Association for Computing Machinery, Inc., May 2007. http://research.microsoft.com/apps/pubs/default.aspx?id=74164
7. ITU: Recommendation X.800, Security Architecture for Open Systems Interconnection for CCITT Applications. International Telecommunications Union (formerly known as the International Telegraph and Telephone Consultantive Committee), Geneva (1991). (X.800 is a re-edition of IS7498-2)
8. ITU: Recommendation X.509 v3, The Directory: Authentication Framework (also known as ISO/IEC 9594–8). International Telecommunications Union, Telecommunication Standardization Sector(ITU-T), Geneva, June 1997
9. Jøsang, A.: Trust extortion on the internet. In: Meadows, C., Fernandez-Gago, C. (eds.) STM 2011. LNCS, vol. 7170, pp. 6–21. Springer, Heidelberg (2012)
10. Klevjer, H., Varmedal, K.A., Jøsang, A.: Extended HTTP digest access authentication. In: Fischer-Hübner, S., de Leeuw, E., Mitchell, C. (eds.) IDMAN 2013. IFIP AICT, vol. 396, pp. 83–96. Springer, Heidelberg (2013)
11. PandaLabs: PandaLabs Quarterly Report, Q2, June 2012. http://press.pandasecurity.com/wp-content/uploads/2012/08/Quarterly-Report-PandaLabs-April-June-2012.pdf
12. Varmedal, K.A., Klevjer, H., Hovlandsvåg, J., Jøsang, A., Vincent, J., Miralabé, L.: The OffPAD: Requirements and Usage. In: Lopez, J., Huang, X., Sandhu, R. (eds.) NSS 2013. LNCS, vol. 7873, pp. 80–93. Springer, Heidelberg (2013)