Identity Management with Petname Systems

Md. Sadek Ferdous

28th May, 2009

Overview

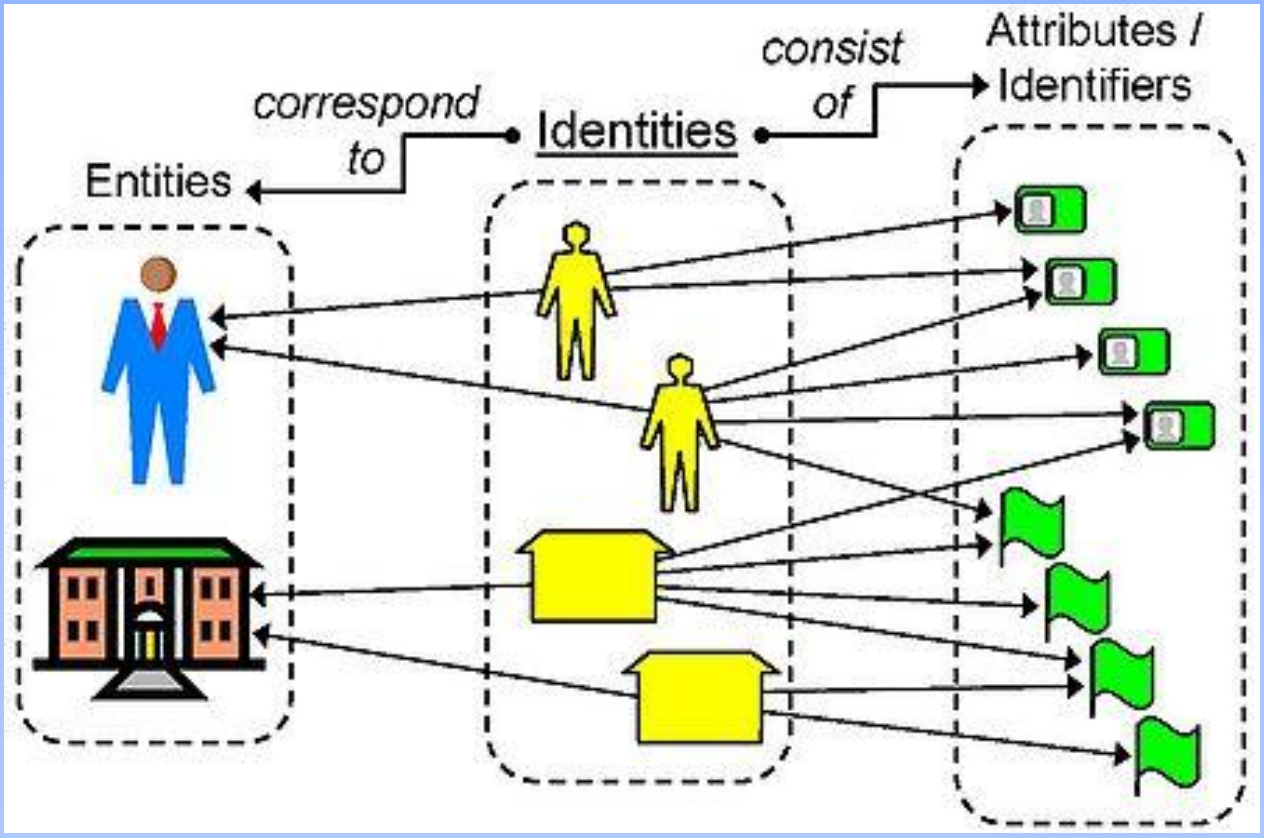
- Entity, Identity, Identity Management
- History and Rationales
- Components and Properties
- Application Domain of Petname Systems
- Security Usability Properties
- Security Usability Analysis of two applications
- Conclusions

Entity & Identity

- Entity
  - A physical or logical object which has a separate distinctive existence either in a physical or a logical sense.
  - A person, an organization or a machine (computer) operated by any person or organization will be denoted as entity.
- Identity
  - Identity is the fundamental property of any entity that declares the uniqueness or sameness of itself and makes it distinctive from other entities in a certain context.
  - An entity can have multiple identities, but an identity cannot be associated with more than one entity.
  - Each identity can consist of multiple attributes that are known as identifiers.
  - Attributes can have different properties, such as being transient or permanent, self-selected or issued by an authority, suitable for human interpretation or only by computers.

Entity & Identity

Conceptual relationship between identities, the entities and the attributes:



Identity Management

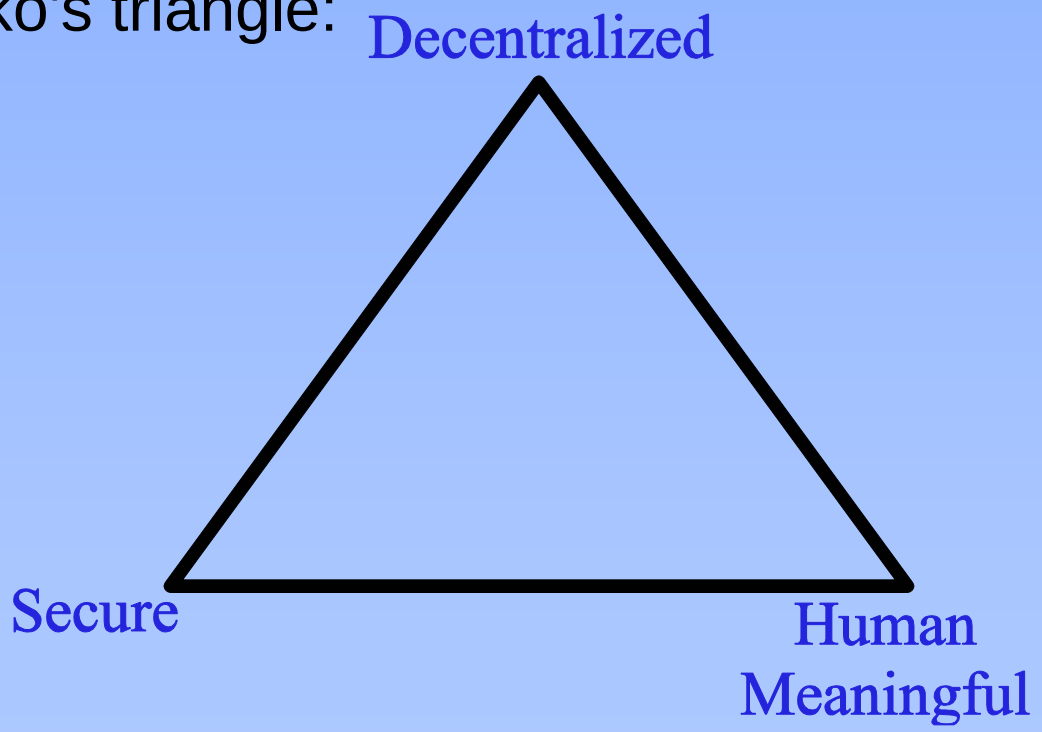
- Identity Management (IdM, in short) consists of technologies and policies for representing and recognizing digital identities of entities.
- Basically IdM can be of four types:
  i) Managing user identities on the server side,
  ii) Managing user identities on the client side,
  iii) Managing server identities on the server side, and
  iv) Managing server identities on the client side.
- Traditionally, IdM refers to the Type 1, overlooking all other three types.
- Meaning users currently have little support in the form of software solutions on the client side to manage service provider identities.
- Petname Systems provide support for the management of SP identities on the client side.
- This specifically solves problems related to the difficulty of verifying the identity of web sites, as e.g. in case of phishing attacks.

Petname Systems: Background

- Identity Management System (IdMS) uses a namespace, a logical and abstract set of identifiers, to generate an identifier for an entity.
- In such case, the main requirement is uniqueness such that each identifier maps to a unique entity.
- The larger the namespace, the more unique identifiers it contains.
- However, a global namespace is no more easily memorable, e.g. IP addresses.
- Zooko Wilcox-O'Hearn in his influential web article published in 2001 mentioned three desirable properties of an identifier: Global, Unique and Memorable.
- To be memorable, an identifier has to pass the so-called “moving bus test”.
- An identifier will be unique if it is collision-free within the domain.
- Zooko also concluded and gave evidence that no identifier can have all the three properties and suggested to choose any two of them.

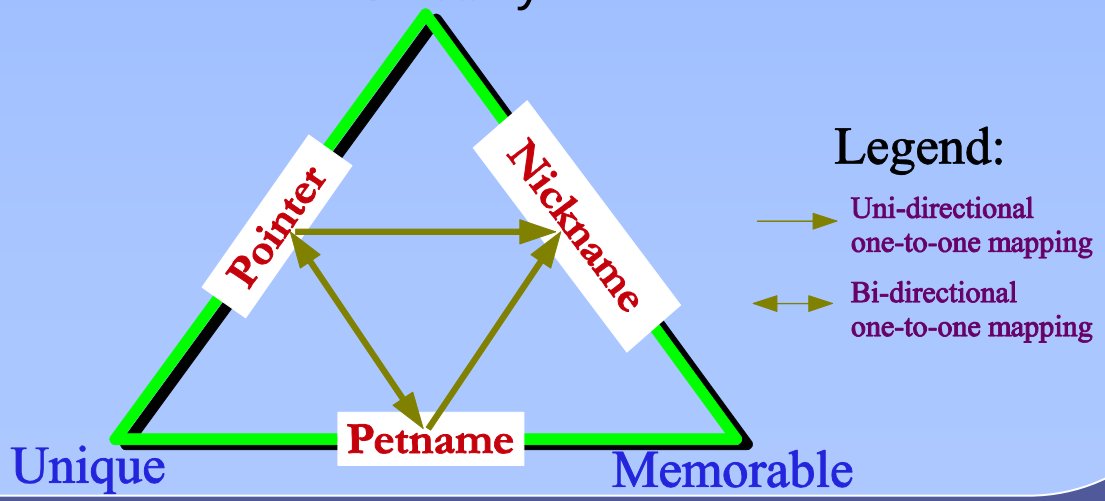
Petname Systems: Background

A triangle where the three properties are placed in the three corners is commonly known as Zooko's triangle:



Petname Model: Definition

- Three unique pairs can be created using these three properties: 1) Global-Memorable, 2) Memorable-Unique and 3) Global-Unique.
- A naming system can be designed to achieve all the three properties of the Zooko's triangle using this three pairs. The Petname Model represents one such naming system.
- Petname Model will be used to denote the abstract properties of Petname Systems. An implementation of the Petname Model is a Petname System.



Petname Model: Components

- **Pointer:** Pointer implies a globally unique and securely collision free identifier which can uniquely identify an entity. Example: A public/private key pair and a fully qualified pathname of a file in an Internet file server. Pointer (e.g. Public key) may not be memorable to human.
- **Nickname:** It is a non-unique but global and memorable name created by the owner of the Pointer. Example: Title of a webpage.
- **Petname:** The Petname is a memorable name created by the user to refer to a specific Pointer of an entity. It is unique within the domain of a single user and obviously not global.
- **Relationship:** There is a bidirectional one-to-one mapping between Pointers and Petnames within the domain of each user. A Nickname has a one-to-many relationship to the set of Pointers. A single Nickname can always be uniquely resolved from the Petname, but the Nickname is not necessarily unique for the Petname. For that reason, a Petname can not be uniquely resolved from a Nickname.

Petname Systems: Properties

- **Functional Properties:** Functional properties are those basic properties that are mandatory for a Petname System. The properties are:

[F1.] A Petname System must consist of at least a Pointer and a Petname.

[F2.] Nickname is optional.

[F3.] Pointers must be strongly resistant against forgery so that the Pointer can not be used to identify a false entity.

[F4.] For every user there must be a bi-directional one-to-one mapping between the Pointer and the Petname of each entity.

Petname Systems: Properties

- **Security Usability Properties:**
- **Security action** is when users are required to produce information and security tokens, or to trigger some security relevant mechanism. For example, typing and submitting a password is a security action. Security action properties are:

[SA1.] It is the user who must assign the Petname for the each Pointer.

[SA2.] Users must assign the Petname for the Pointer with explicit action.

[SA3.] Each Petname should be editable.

[SA4.] Suggestion on the Petname based on the Nickname can be provided as an aid for the user to select a Petname for a Pointer.

[SA5.] If a suggestion is provided and the user wants to accept it as the Petname, then he must do so with explicit action.

[SA6.] Petname Systems must make sure that the user-selected, created or suggested Petname is sufficiently distinct from the Nickname so that the user does not confuse them with each other.

[SA7.] Petname Systems must make sure that the user-selected, created or suggested Petname must be sufficiently different from existing Petnames so that the user does not confuse them.

[SA8.] If the user chooses a Petname that may resemble a Nickname or other Petnames, he should be warned explicitly.

[SA9.] The User should be alerted to apply a Petname for the entity that involves in highly sensitive data transmission.

Petname Systems: Properties

- **Security Usability Properties:**
- **Security conclusions** enable the user to conclude on the security state of the system by observing security relevant evidence and assessing this together with assumptions. Security conclusion properties are:

[SC1.] The Pointer and the corresponding Petname must be displayed at all times through the user interface of the Petname System. This will make the user confident about his interaction and help to draw the security conclusion easily.

[SC2.] The Petname for a Pointer should be displayed with enough clarity at the user interface so that it can attract the user's attention easily.

[SC3.] The absence of a Petname for a Pointer should be clearly and visually indicated at the user interface so that the user is surely informed about its absence.

[SC4.] The visual indication for suggested Petnames and Nicknames should be unambiguous enough so that the user does not confuse them with each other.

[SC5.] The warning message that will be provided when there is a direct violation of any of the above properties should be clear enough so that the user can understand the problem and take the necessary security action.

Application Domains

- Real World
- Phone/E-mail Contact List
- IM Buddy List
- DNS & Anti-Phishing Tool
- IP Address
- CapDesk and Polaris
- OpenPGP
- Process Handling

Security Usability Analysis

- **Security Action Usability Principles*:**

[A1.] Users must understand which security actions are required of them.

[A2.] Users must have sufficient knowledge and the ability to take the correct security action.

[A3.] The mental and physical load of a security action must be tolerable.

[A4.] The mental and physical load of making repeated security actions for any practical number of instances must be tolerable.

- **Security Conclusion Usability Principles*:**

[C1.] Users must understand the security conclusion that is required for making an informed decision.

[C2.] The system must provide the user with sufficient information for deriving the security conclusion.

[C3.] The mental load of deriving the security conclusion must be tolerable.

[C4.] The mental load of deriving security conclusions for any practical number of instances must be tolerable.

***A Jøsang, M AlZomai and S Suriadi Usability and Privacy in Identity Management Architectures *Proceedings of the Australasian Information Security Workshop AISW'07* , Ballarat, January 2007**

Security Usability Analysis

- When a Petname System satisfies SA1-SA3 and SA6-SA9 of the Security Action properties, it implicitly implies that principles A1 and A2 are also satisfied.
- SA4-SA8 will act as the aid for the user to select a Petname for a Pointer. Therefore satisfying these five properties will implicitly lead to the principles A3 and A4 also being satisfied.
- Whenever a Petname System satisfies SC1-SC3, it will explicitly satisfy C1 and C2.
- The security conclusion properties SC2-SC5 should be applied to enable a user to draw conclusion with ease and thus if followed will satisfy principles C3 and C4.

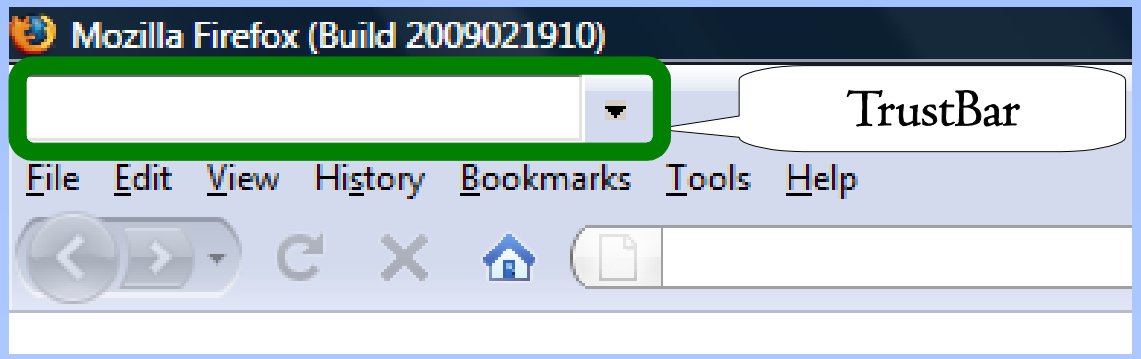
Security Usability Analysis

- Security usability of two Petname System applications have been analyzed using Cognitive Walkthrough method to find out their level of compliance with the Security Usability principles.
- Both toolbars are Firefox extension, and are aimed at simplifying client-side management of SP identities and at providing a better defense mechanism against Phishing attacks..

Security Usability Analysis

- The Petname Tool:

- TrustBar:



Security Usability Analysis

Major findings:

- Both tools suffer from the absence of the crucial property F4, that is, they allow the same Petname to be assigned for different websites and thus violating the one-to-one mapping principle,
- Both tools lack in providing more convincing techniques to catch user attention,
- The Petname Tool lacks in providing the standard Help or About functionality of a software which can make a user confused about its functionality.
- TrustBar has another major lacking: it does not provide any sort of warning when something goes wrong.
- TrustBar is not consistent with its user-interface when it is switched back and forth between Petname and Petlogo.

Security Usability Analysis

Full analysis between two tools can be summarized in the following table:

Tool Name	F				SA									SC				
	1	2	3	4	1	2	3	4	5	6	7	8	9	1	2	3	4	5
Petname Tool	Y	N	Y	N	Y	Y	Y	N	N	N	N	Y	N	Y	Y	Y	N	Y
TrustBar	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	N	N	N	Y	Y	Y	Y	N

Conclusions

This presentation has:

- introduced you to the concept of Petname Systems, its background, components and properties,
- shown how Petname Systems can be applied for client site management of Service Provider identities,
- covered the Security Usability issues of Petname Systems based on Security Usability principles, and
- presented the analysis of Security Usability of two Petname System applications and found that they did not meet the Security Usability principles.

Thank you.
Questions or comments?