Threshold Things That Think: Usable Authorisation for Resharing

[Poster Abstract SOUPS 2009]

Roel Peeters[*], Markulf Kohlweiss and
Bart Preneel
K.U.LEUVEN, ESAT/COSIC and IBBT
Kasteelpark Arenberg 10 bus 2446
3001 Heverlee, Belgium
firstname.lastname@esat.kuleuven.be

Nicky Sulmon
K.U.LEUVEN, CUO and IBBT
Parkstraat 45
3000 Leuven, Belgium
nicky.sulmon@soc.kuleuven.be

1. INTRODUCTION

People start carrying around more and more mobile devices that can contain sensitive data. To protect these devices, Desmedt et al. [1] proposed a threshold security architecture for Things That Think. These things are personal devices that are frequently in the user's proximity and able to interact with each other. In the proposed architecture, security is the result of the cooperation of at least the threshold number of personal devices. For threshold security each personal device possesses a part of the shared key. When at least the threshold number of these devices cooperate, this shared key can be used to, for instance, place signatures or decrypt encrypted information.

The advantages of deploying a threshold cryptography scheme are twofold: a user does not need all his personal devices (e.g. dead battery, device left at home) to access the shared key and an adversary does not gain any knowledge of the shared key when he does not compromise the threshold number of devices.

For a threshold security architecture on Things That Think to be practical, a mechanism allowing the user to add or remove devices from the set of personal devices is essential. Refreshing the shared key enhances security. Adding a device, removing a device and refreshing the shared key are essentially the same in terms of the underlying "resharing" protocol. One example of a protocol for resharing can be found in [6].

Little attention has been paid to the problem of authorisation for resharing. Proper authorisation is necessary to prevent an adversary from altering the set of personal devices in such a way that he would be able to break the scheme. Moreover authorisation should not enable the adversary to succeed in a *Denial of Service* (DoS) attack and prevent the genuine user from signing and/or decrypting data.

The authors developed a protocol to manually authorise resharing in [4]. This paper focuses on the usability aspect, an essential part of the protocol development. Although the proposed manual authorisation protocol is studied in the context of resharing, it could also be used to authorise signing data or as a bootstrapping mechanism. An overview of related work on usability regarding the pairing of two devices is given by Saxena et al. [5].

2. MANUAL AUTHORISATION

The user can manifest himself towards his personal devices by entering/confirming his request at the threshold number of personal devices. The resharing protocol is then triggered at each device after the threshold number of request approvals have been collected.

New information, in this case public keys, needs to be authenticated by deploying a *Group Message Authentication* (GMA) [2] protocol. By visually comparing the *Short Authenticated Strings* (SAS) the user ensures that the information was exchanged between the intended devices.

2.1 User interactions

The user is provided with three options:

- add a device to the set of personal devices;
- remove a device from this set;
- refresh the shared key.

Any device can be added to the set of personal devices by initiating a procedure on the specific device itself.

When adding or removing a device, the user selects this device from a list of discovered devices, personal devices respectively. After selecting one option his request is broadcast to all personal devices.

The user then confirms his request at the threshold number of his personal devices. It is recommended that the user visually checks his request on the displays of his other personal devices before confirming. The displayed request consists of the name of the initiating device and the selected option. If a device is added, the user is also requested to compare the SAS resulting from the GMA protocol between that device and the personal devices.

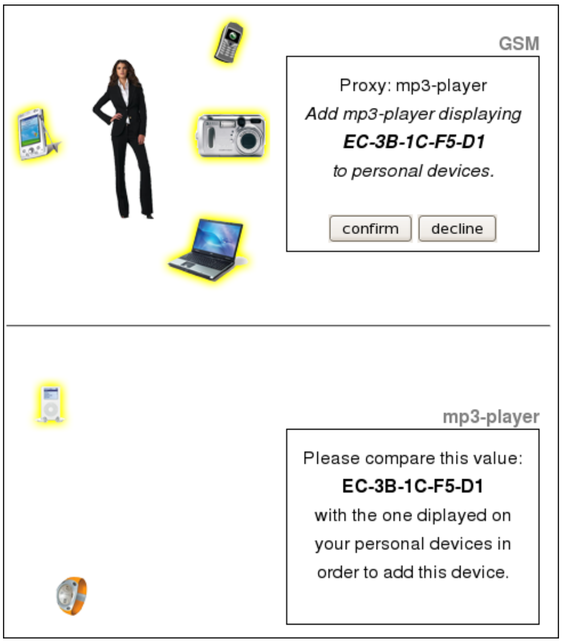
When at least the threshold of devices have broadcast the user's approval, all personal devices conclude that resharing is authorised and resharing takes place. The personal devices indicate that the chosen option is in progress. Upon successful resharing the devices indicate success.

3. USABILITY

We developed a web-based mock up interface, as depicted in Fig. 1. The interface and user interactions were evaluated by two experts in the field of *Human Computer Interface* (HCI) with no specific knowledge of the domain of

[*] Roel Peeters is funded by a research grant of the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen).

security systems. Afterwards a preliminary study was conducted among students from different backgrounds.



Proxy: mp3-player

Add mp3-player displaying

EC-3B-1C-F5-D1

to personal devices.

Please compare this value:

EC-3B-1C-F5-D1

with the one diplayed on your personal devices in order to add this device.

Figure 1: Web-based interface. Available on-line at http://homes.esat.kuleuven.be/~rpeeters/usability/

3.1 Expert evaluation

Although the interface is somewhat limited in the offered functionality, the review indicated some potential usability obstacles.

Match between system state and the real world [3]. A significant usability problem was located at the very first point-of-contact between the system and the user. People with a security background might not be the typical end users of the proposed security scheme. In this light, it is undesirable to confront your end users with any technical details about the algorithms behind the security system. To explain the scheme, the reviewers rewrote system-oriented terms, e.g., "threshold secret sharing" to match more familiar concepts, e.g., "network of trusted devices".

Explicit authorisation [7].
Users granting or removing authorisations to/from other actors must unambiguously know the consequences of their actions. On that account, many labels (buttons, titles, etc.) have been revised. For example, when adding a new device to the set of personal devices, the button "next" has become "add" to prevent users from assuming that there will be another step in a wizard-like setting.

3.2 Preliminary study

The most important thing for a user to successfully complete a scenario is the ability to imagine a real-life use case. Test subjects were initially provided with an interface presenting the option to do resharing (without adding or removing a device) to renew the shared key. Because most users did not see any reason for doing this, this option led to confusion. The redesigned interface abstracted away from resharing and introduced the "refreshing" of a shared key.

After the authorisation of adding/removing a device or refreshing the shared key, the display showed that resharing is in progress and eventually ended successfully. This confused the test subjects who see the three options (adding, removing and refreshing) as three distinct concepts. This also made clear that there should be a clear distinction between the underlying protocol, resharing, and the provided options that caused the resharing. We abstracted away the underlying protocol and now display that the selected option is in progress or ended successfully.

Removing a device was generally considered straightforward. Although it was not possible to authorise the removal of a device on the device itself, half of the test users wanted to be able to start the authorisation from this device.

The actions for adding a device used to consist of: a manual authentication step between the new device and one of the personal devices; a confirmation step on the threshold number of personal devices; and finally a verification step on the threshold number of personal devices. This clearly put quite a high burden on the user. We redesigned the protocol for authorisation to make use of a group authentication protocol. This allows to get rid of the verification step. The user could only start adding a device from a personal device, but all test subjects wanted to be able to start from the device to be added. We also learnt that the values for a user to compare in the manual authentication step should not be displayed in two groups on one display, e.g. in two consecutive lines. Some thought that they needed to compare these two values instead of comparing the values across displays.

4. CONCLUSIONS

Adding a device, removing a device and refreshing a shared key are three instances of resharing. However, users think of these as three different concepts, and this should be translated as such in the user interface. In terms of protocol design we learnt that: the protocol should allow starting adding or removing a device from the device in question; authentication of new data needs to be integrated with authorisation and should take place among all participating devices.

5. REFERENCES

[1] Y. Desmedt, M. Burmester, R. Safavi-Naini, and H. Wang. Threshold Things That Think (T4): Security Requirements to Cope with Theft of Handheld/ Handless Internet Devices. In *Symposium on Requirements Engineering for Information Security*, 2001.

[2] S. Laur and S. Pasini. SAS-Based Group Authentication and Key Agreement Protocols. In *11th International Workshop on Practice and Theory in Public-Key Cryptography*, volume 4939 of *LNCS*, pages 197–213. Springer, 2008.

[3] R. Molich and J. Nielsen. Improving a human-computer dialogue. *Communications of the ACM*, 33(3):338–348, 1990.

[4] R. Peeters, M. Kohlweiss, and B. Preneel. Threshold things that think: Authorisation for resharing. In *iNetSec*, 2009.

[5] N. Saxena, M. B. Uddin, and J. Voris. Universal Device Pairing using an Auxiliary Device. In *2008 Symposium on Usable Privacy and Security*, pages 56–67, 2008.

[6] T. M. Wong, C. Wang, and J. M. Wing. Verifiable Secret Redistribution for Threshold Sharing Schemes. Technical Report CMU-CS-02-114, Carnegie Mellon University, 2002.

[7] K.-P. Yee. User interaction design for secure systems. In *ICICS '02: Proceedings of the 4th International Conference on Information and Communications Security*, pages 278–290, London, UK, 2002. Springer-Verlag.