Summit: Recovering from lost devices in WebAuthn/FIDO2

Summit Date: August 20, 2018
Summit Location: University of Washington, Seattle, WA

Attendees:
DDS: Aiki Matsushita
Google: Alexei Czeskis, Arnar Birgisson, Christiaan Brand
Microsoft: Anthony Nadalin
MIT/W3C: Samuel Weiler
Nitrokey: Jan Suhr
Nok Nok Labs: Matt Lourie
NTT Labs: Hideo Nishimura
RSA: Salah Machani
University of Washington: Tadayoshi Kohno, Alex Takakuwa
Yubico: David Treece, Derek Hanson

Presentation 1: Salah Machani, RSA
Slides: https://homes.cs.washington.edu/~alextaka/ARFIDO_Aug20.pdf

Presentation 2: Alex Takakuwa, UW
Slides: https://homes.cs.washington.edu/~alextaka/summit_presentation_ORS.pdf

Full notes:
https://docs.google.com/document/d/1m1iCEU2X5CxFoaIef7HLgfC6IOMxcNKQsTwtxru-r8Y/edit?usp=sharing

Summary

The goals of the summit were as follows:
1. Create a list of requirements for solutions to “Recovering from Device Loss”
2. Determine next steps

This document summarizes the discussions and progress we’ve made towards these goals.

Requirements:
Usability Requirements:

- We are trying to solve this problem (Recovering from Device Loss) in a scalable way. In other words, we require solutions to allow WebAuthn/FIDO2 users to recover *all* accounts with one recovery action instead of requiring a recovery action for *each* RP.

    - There are cases where users may want to recover only a subset of accounts using a given mechanism. See item #3 under User Choice.
- Users should *not* have to carry multiple devices simultaneously for authentication or registration events.
- We discussed whether recovery processes should have a revoke/temporary revoke/resume state comparable to putting a temporary block of a lost (but not known to be stolen) credit card, but did not explicitly place this in scope for recovery work. We should discuss further whether this is necessary for recovery protocols or whether recovery and revocation should be separate.

User Choice:

- Should users decide to use a recovery method with a lower “Security Assurance Level” (ex: copying keys, using a third party for federation, etc), they should be able to do so as long as that type of recovery has a “Security Assurance Level” that meets or exceeds the minimum requirements of the Relying Party.
- Should users decide to recover with each RP instead of recovering with a single action, they should be able to do so (fallback to status quo).
- It may be useful for users to be able, at the time of each registration, for the user to be able to opt-out of recovery for that account - i.e. if the user wishes to NOT empower a recovery device or service to recover that particular account.
- Similarly, users should be empowered, at the time of each registration, to prevent enumeration of a particular account by the normal recovery device or service.

Relying Party Choice:

- We should allow a way for RPs to specify any registration or recovery security requirements should they choose to enforce them.
- RPs should also be allowed to deny registrations or recoveries coming from or going through insufficient devices/services, or deny recoveries altogether.

Security Goal:

- We discussed whether users should be able to see when recoveries have occurred (non-concealable recoveries). Some solutions provide this, while others do not. Though we don’t know whether this is in scope for any standards push, it merits future discussion.

There is considerable nuance in the choices that both users and relying parties will make regarding security and usability. We have logically split our use cases into “high security” and “low security” cases. We believe that most accounts fall into the “low security” category, but should still benefit from the increased usability and security guarantees WebAuthn provides. We should allow for recoveries that satisfy the above requirements in both “high” and “low” security cases.

However, we recognize that users may have difficulty differentiating between high and low security for many apps (ex: social media, email). Further discussion is required to determine what the user flow should be in these cases. For example, should a user be prompted to select the level they want compared to having it default to, e.g., low security? Should this be left up to users or relying parties?

Next Steps:

We discussed actionable items that will help us meet the above requirements.

1. Expand the “Security Assurance Level” to account for recoveries
    a. This may require a framework to allow Relying Parties to describe the “Security Assurance Level” to authenticators ahead of time
2. We would like to push this to a standard.
    a. Find a place to push standards changes. Proposals are ISO and IETF, but further discussion is necessary.
3. Evaluate Concrete Proposals
    a. Proposals should be discussed and evaluated against the above requirements as well as the usability/security goals for different user groups. Ideally, we will continue to refine proposals until we are comfortable enough to push a solution to spec.
4. Continue discussion at the FIDO plenary in Singapore in the FIDO TWG, will schedule hour time slot