Recovering from Lost Devices in WebAuthn

Pre-emptively syncing recovery keys

Alex Takakuwa, University of Washington

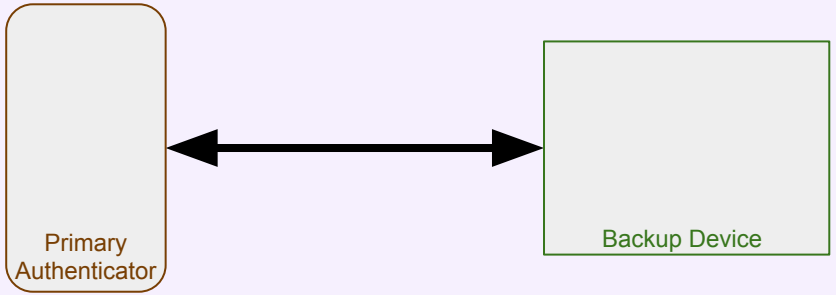
Pre-emptively Sync Keys (PSK) Solution

- Assumptions:
  - Secure channel between backup device and primary authenticators
  - Secure storage and computation will continue to get cheaper/easier
- Goals:
  - Recovery from lost primary authenticator
  - Do not need to trust anyone for privacy or availability
  - Allow a single recovery device to recover many primary authenticators
  - Do not otherwise weaken existing WebAuthn scheme
  - Allow for multiple backup devices
  - Allow many types of backup (third party server, hardware device, key splitting)
  - Allow Transfer of Access from Primary Authenticators
  - Allow Transfer of Access from Backup Authenticators
- Open Problems:
  - Storage Overhead
  - Revocation

User Experience

PSK Solution - Setup (User Experience)

1. User syncs backup device with primary authenticator
2. User uses primary authenticator for future authentications and registrations



PSK Solution - Registration (User Experience)

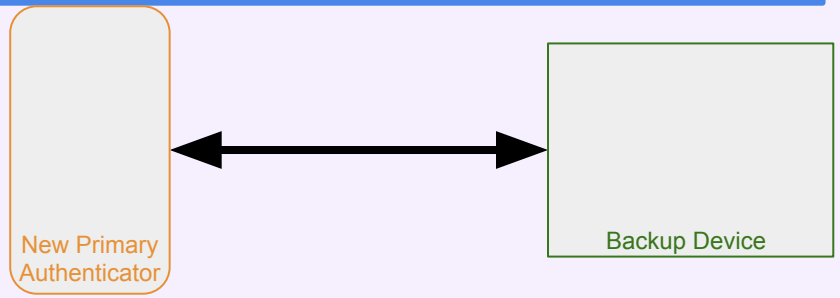
- Unchanged from Standard WebAuthn
  - Does not require the backup device

PSK Solution - Authentication

- Unchanged from Standard WebAuthn
  - Does not require the backup device

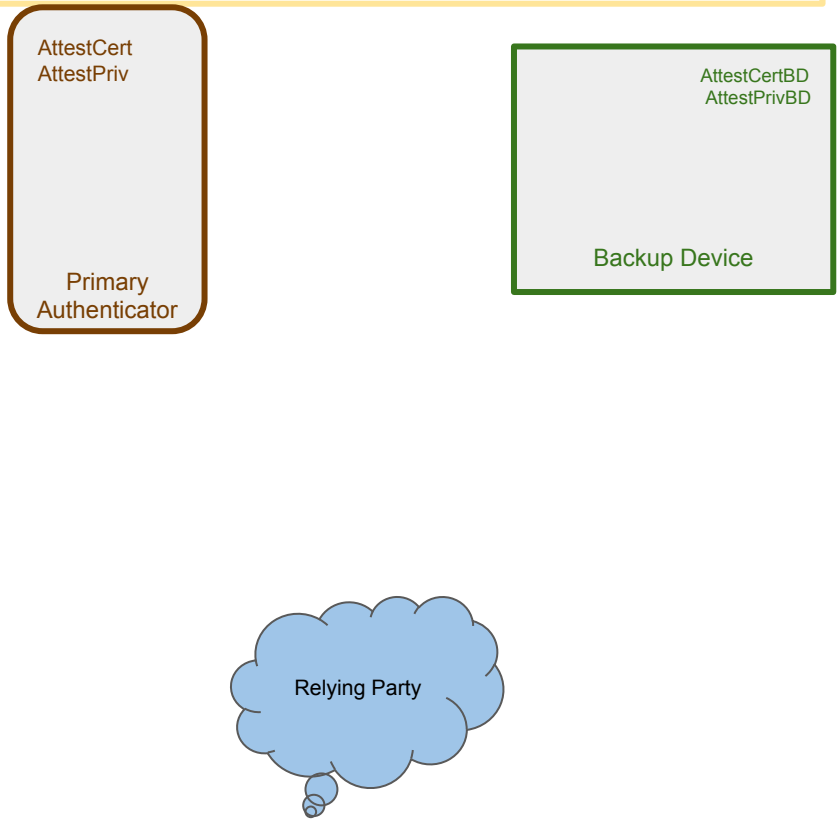
PSK Solution - Recovery (User Experience)

1. User syncs backup device with new primary authenticator
2. User selects “recover from Old Primary Authenticator”.
3. User uses new primary authenticator for future authentications and registrations



Technical Details

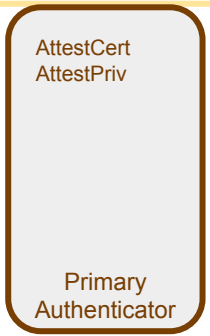
PSK Solution - Setup (Technical)

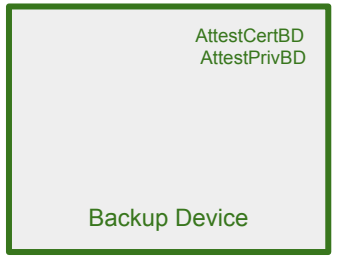


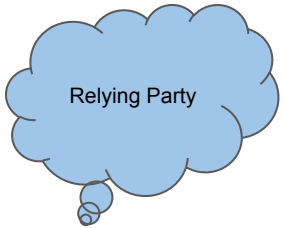
PSK Solution - Setup (Technical)

1. User creates secure channel between devices.
2. Backup device generates (* some number) of key pairs and corresponding key handles and associates each of them with the Primary Authenticator.
3. Backup Device sends its Attestation Certificate to the Primary Authenticator
4. Backup Device sends all generated public keys and corresponding key handles to the Primary Authenticator, each signed with its Attestation Private Key

* The backup device should generate (at first setup) enough key pairs to last for all registrations performed by this particular Primary Authenticator. This number can be reduced, but we would need to notify the user to re-sync with the Backup Device should the user run low on generated Backup Device public keys.



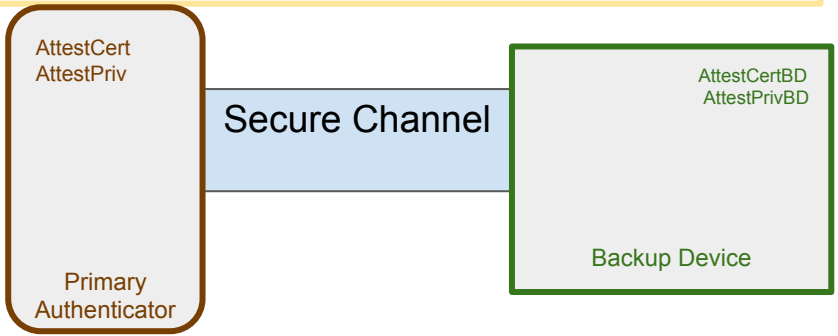




PSK Solution - Setup (Technical)

1. User creates secure channel between devices.
2. Backup device generates (* some number) of key pairs and corresponding key handles and associates each of them with the Primary Authenticator.
3. Backup Device sends its Attestation Certificate to the Primary Authenticator
4. Backup Device sends all generated public keys and corresponding key handles to the Primary Authenticator, each signed with its Attestation Private Key

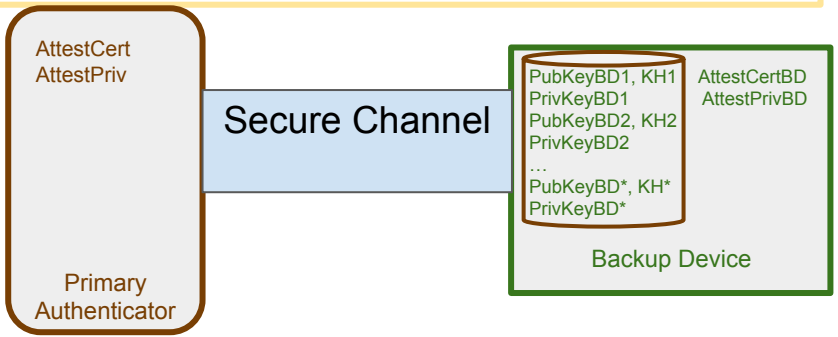
* The backup device should generate (at first setup) enough key pairs to last for all registrations performed by this particular Primary Authenticator. This number can be reduced, but we would need to notify the user to re-sync with the Backup Device should the user run low on generated Backup Device public keys.



PSK Solution - Setup (Technical)

1. User creates secure channel between devices.
2. Backup device generates (* some number of) key pairs and corresponding key handles and associates each of them with the Primary Authenticator.
3. Backup Device sends its Attestation Certificate to the Primary Authenticator
4. Backup Device sends all generated public keys and corresponding key handles to the Primary Authenticator, each signed with its Attestation Private Key

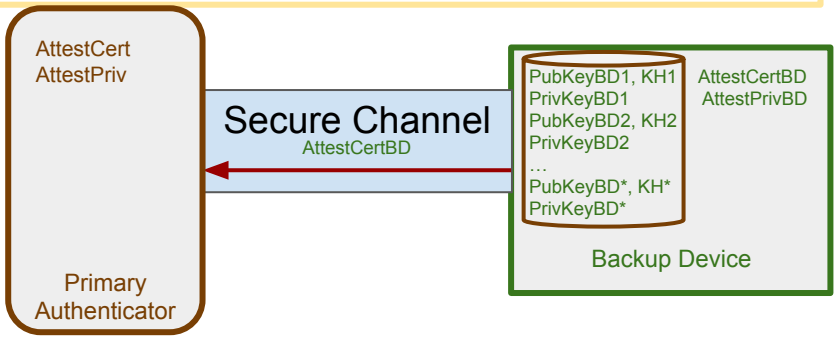
* The backup device should generate (at first setup) enough key pairs to last for all registrations performed by this particular Primary Authenticator. This number can be reduced, but we would need to notify the user to re-sync with the Backup Device should the user run low on generated Backup Device public keys.

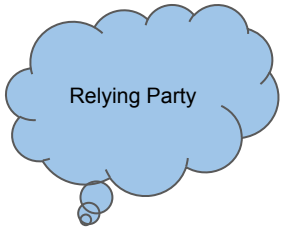


PSK Solution - Setup (Technical)

1. User creates secure channel between devices.
2. Backup device generates (* some number of) key pairs and corresponding key handles and associates each of them with the Primary Authenticator.
3. Backup Device sends its Attestation Certificate to the Primary Authenticator
4. Backup Device sends all generated public keys and corresponding key handles to the Primary Authenticator, each signed with its Attestation Private Key

* The backup device should generate (at first setup) enough key pairs to last for all registrations performed by this particular Primary Authenticator. This number can be reduced, but we would need to notify the user to re-sync with the Backup Device should the user run low on generated Backup Device public keys.

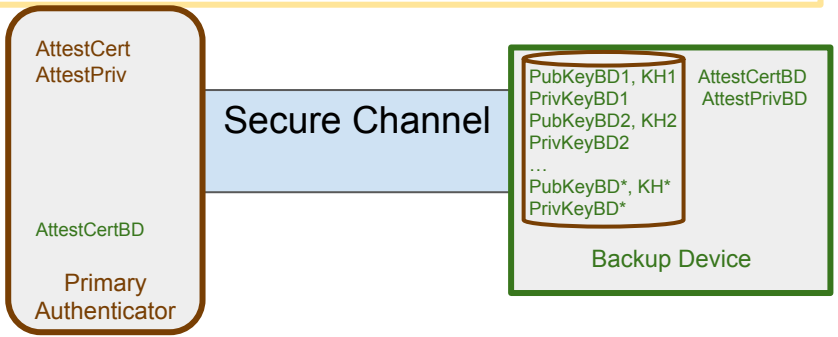


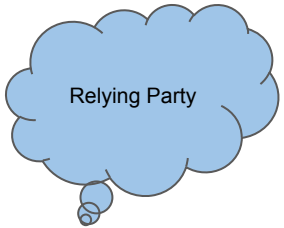


PSK Solution - Setup (Technical)

1. User creates secure channel between devices.
2. Backup device generates (* some number of) key pairs and corresponding key handles and associates each of them with the Primary Authenticator.
3. Backup Device sends its Attestation Certificate to the Primary Authenticator
4. Backup Device sends all generated public keys and corresponding key handles to the Primary Authenticator, each signed with its Attestation Private Key

* The backup device should generate (at first setup) enough key pairs to last for all registrations performed by this particular Primary Authenticator. This number can be reduced, but we would need to notify the user to re-sync with the Backup Device should the user run low on generated Backup Device public keys.

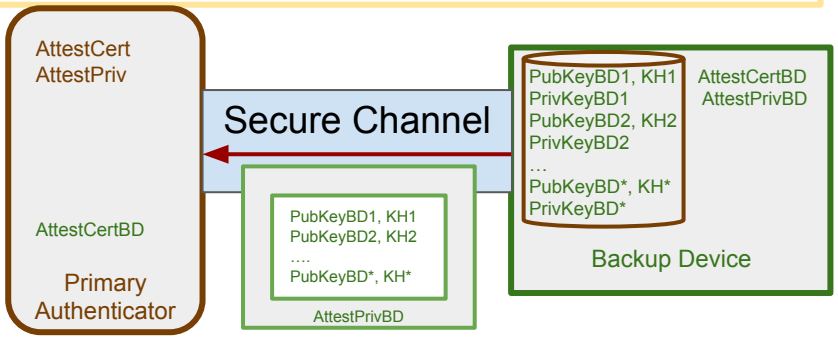




PSK Solution - Setup (Technical)

1. User creates secure channel between devices.
2. Backup device generates (* some number of) key pairs and corresponding key handles and associates each of them with the Primary Authenticator.
3. Backup Device sends its Attestation Certificate to the Primary Authenticator
4. Backup Device sends all generated public keys and corresponding key handles to the Primary Authenticator, each signed with its Attestation Private Key

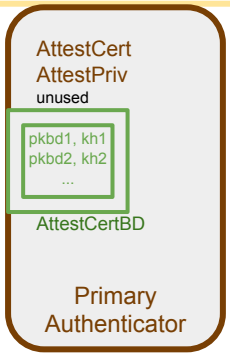
* The backup device should generate (at first setup) enough key pairs to last for all registrations performed by this particular Primary Authenticator. This number can be reduced, but we would need to notify the user to re-sync with the Backup Device should the user run low on generated Backup Device public keys.

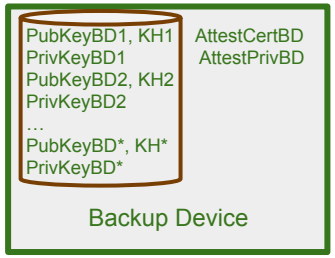


PSK Solution - Setup (Technical)

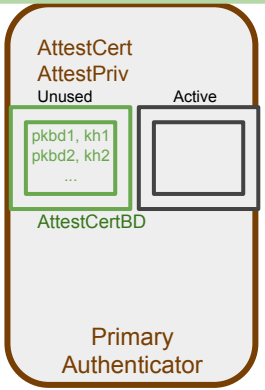
1. User creates secure channel between devices.
2. Backup device generates (* some number of) key pairs and corresponding key handles and associates each of them with the Primary Authenticator.
3. Backup Device sends its Attestation Certificate to the Primary Authenticator
4. Backup Device sends all generated public keys and corresponding key handles to the Primary Authenticator, each signed with its Attestation Private Key

* The backup device should generate (at first setup) enough key pairs to last for all registrations performed by this particular Primary Authenticator. This number can be reduced, but we would need to notify the user to re-sync with the Backup Device should the user run low on generated Backup Device public keys.



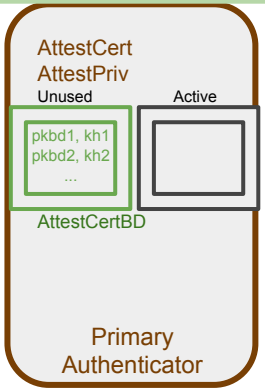


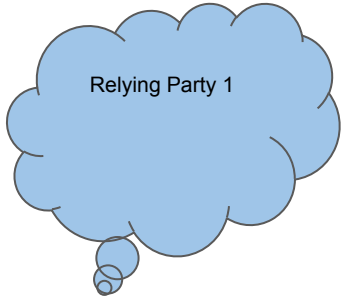
PSK Solution - Registration (Technical)



PSK Solution - Registration (Technical)

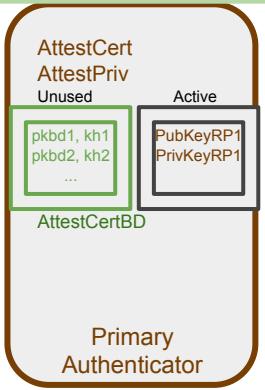
1. Primary Authenticator generates key pair for RP1
2. Primary Authenticator selects an unused recovery key
3. Primary Authenticator registers its generated public key (PubKeyRP1), the selected Backup Key (pkbd1), and the key handle generated by the backup device for that backup key (KH1) with RP1
4. Primary Authenticator stores the public key and key handle generated by the backup device with its own generated key pair, moving the backup key handle and public key out of the “unused” portion of storage.





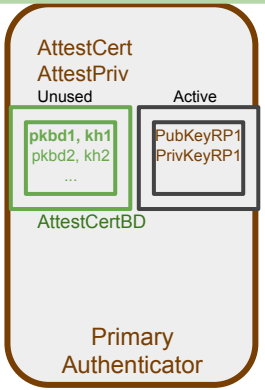
PSK Solution - Registration (Technical)

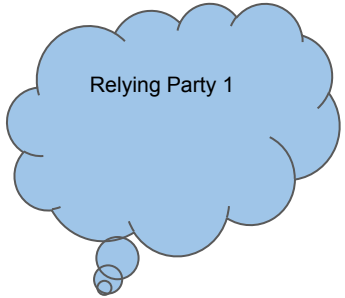
1. Primary Authenticator generates key pair for RP1
2. Primary Authenticator selects an unused recovery key
3. Primary Authenticator registers its generated public key (PubKeyRP1), the selected Backup Key (pkbd1), and the key handle generated by the backup device for that backup key (KH1) with RP1
4. Primary Authenticator stores the public key and key handle generated by the backup device with its own generated key pair, moving the backup key handle and public key out of the “unused” portion of storage.



PSK Solution - Registration (Technical)

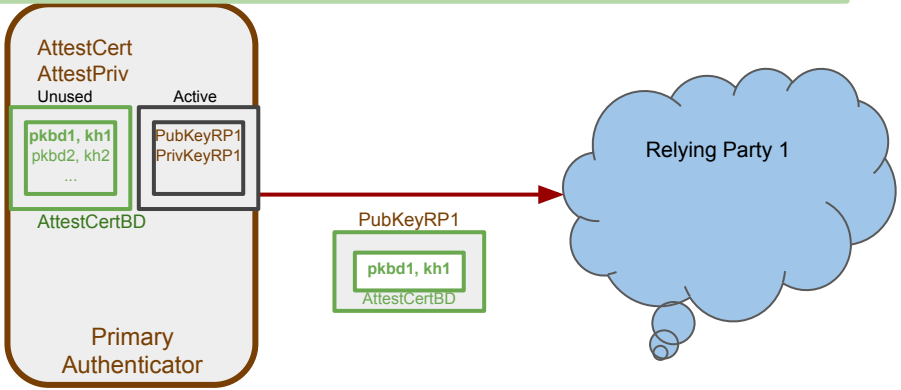
1. Primary Authenticator generates key pair for RP1
2. Primary Authenticator selects an unused recovery key
3. Primary Authenticator registers its generated public key (PubKeyRP1), the selected Backup Key (pkbd1), and the key handle generated by the backup device for that backup key (KH1) with RP1
4. Primary Authenticator stores the public key and key handle generated by the backup device with its own generated key pair, moving the backup key handle and public key out of the “unused” portion of storage.





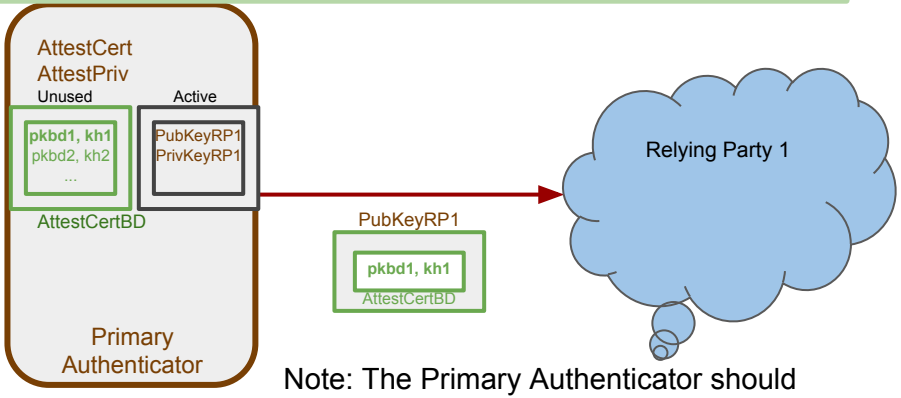
PSK Solution - Registration (Technical)

1. Primary Authenticator generates key pair for RP1
2. Primary Authenticator selects an unused recovery key
3. Primary Authenticator registers its generated public key (PubKeyRP1), the selected Backup Key (pkbd1), and the key handle generated by the backup device for that backup key (KH1) with RP1
4. Primary Authenticator stores the public key and key handle generated by the backup device with its own generated key pair, moving the backup key handle and public key out of the “unused” portion of storage.



PSK Solution - Registration (Technical)

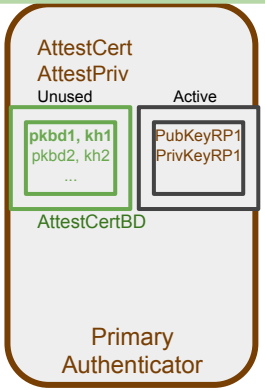
1. Primary Authenticator generates key pair for RP1
2. Primary Authenticator selects an unused recovery key
3. Primary Authenticator registers its generated public key (PubKeyRP1), the selected Backup Key (pkbd1), and the key handle generated by the backup device for that backup key (KH1) with RP1
4. Primary Authenticator stores the public key and key handle generated by the backup device with its own generated key pair, moving the backup key handle and public key out of the “unused” portion of storage.

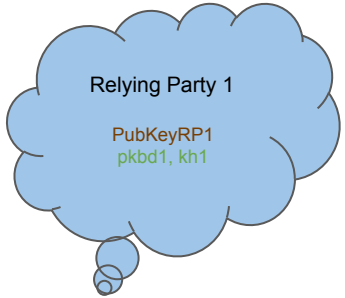


Note: The Primary Authenticator should also send the hardware attestations for both itself and the Backup Device during registrations. The hardware attestation certificate for the backup device cannot be in response to any RP challenge, as it is pre-generated, but can be bound to pkbd1 and kh1.

PSK Solution - Registration (Technical)

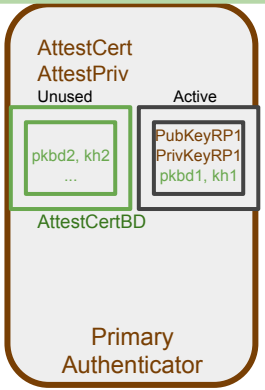
1. Primary Authenticator generates key pair for RP1
2. Primary Authenticator selects an unused recovery key
3. Primary Authenticator registers its generated public key (PubKeyRP1), the selected Backup Key (pkbd1), and the key handle generated by the backup device for that backup key (KH1) with RP1
4. Primary Authenticator stores the public key and key handle generated by the backup device with its own generated key pair, moving the backup key handle and public key out of the “unused” portion of storage.

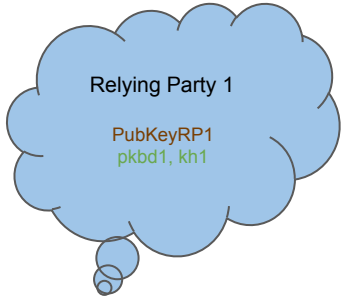




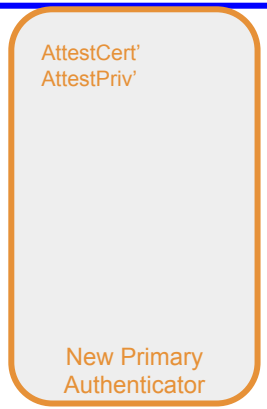
PSK Solution - Registration (Technical)

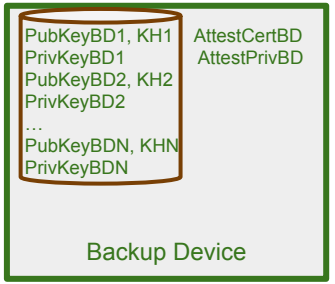
1. Primary Authenticator generates key pair for RP1
2. Primary Authenticator selects an unused recovery key
3. Primary Authenticator registers its generated public key (PubKeyRP1), the selected Backup Key (pkbd1), and the key handle generated by the backup device for that backup key (KH1) with RP1
4. Primary Authenticator stores the public key and key handle generated by the backup device with its own generated key pair, moving the backup key handle and public key out of the “unused” portion of storage.





PSK Solution - Recovery (Technical)

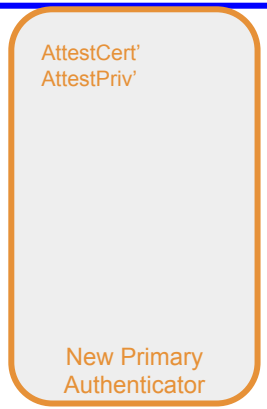


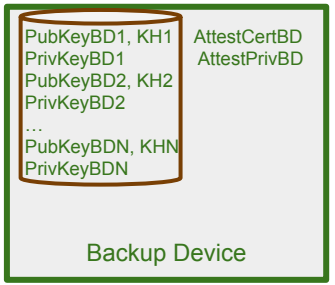


PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.

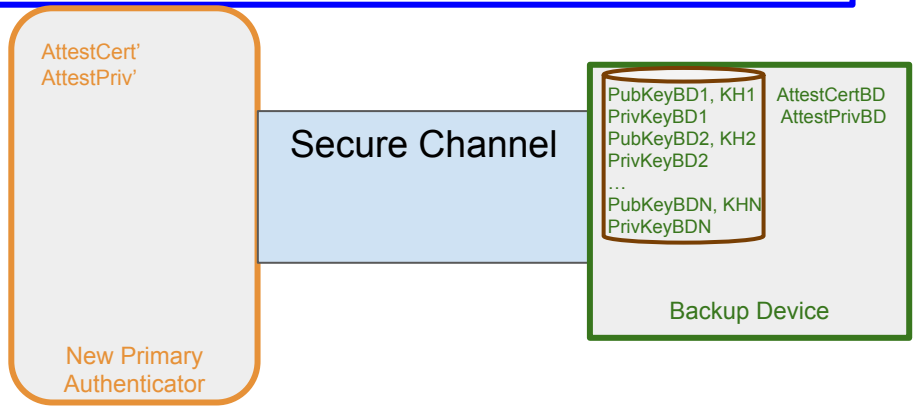




PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

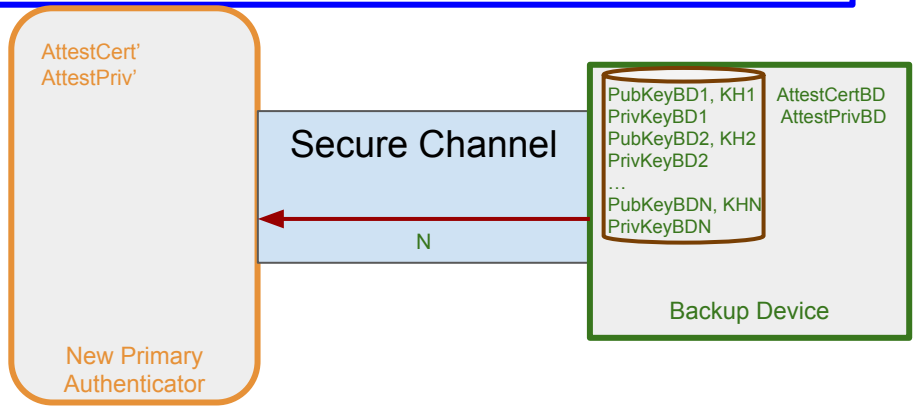
* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.



PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

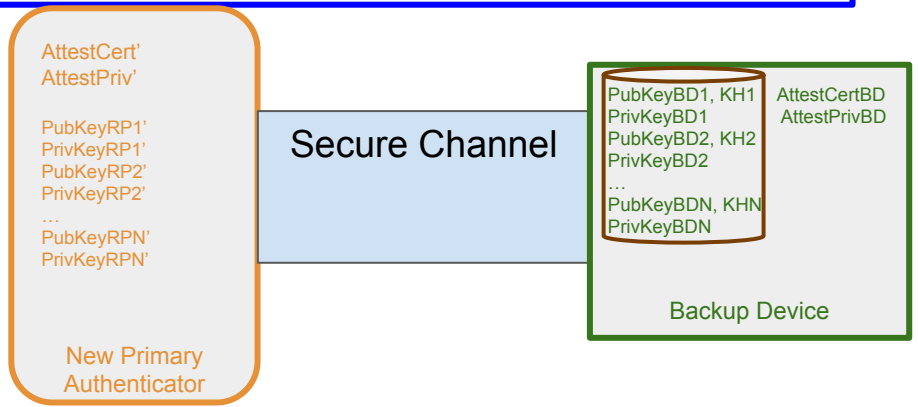
* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.



PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

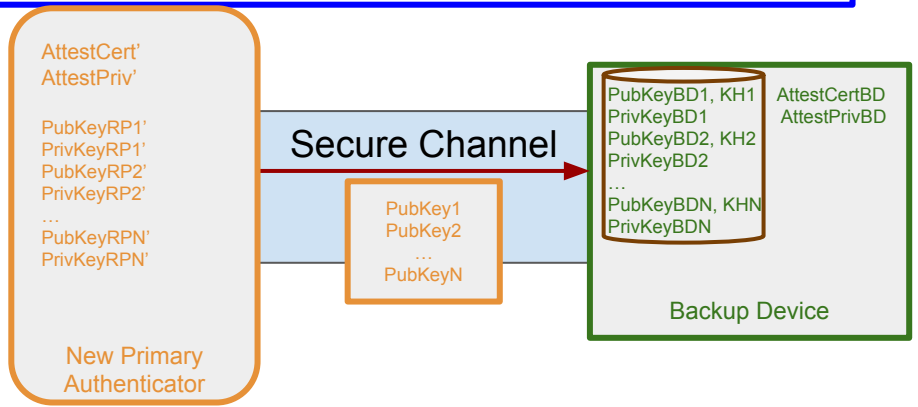
* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.



PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

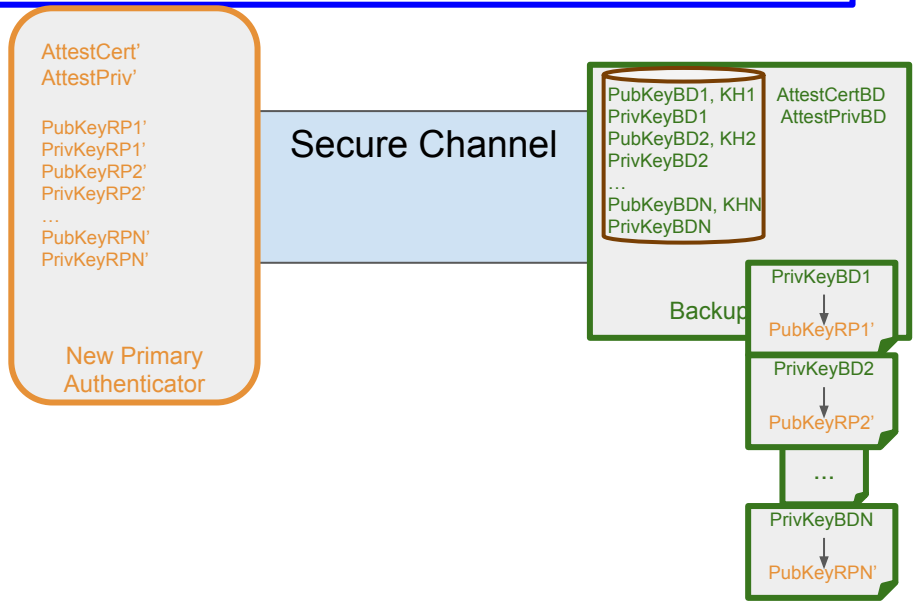
* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.



PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

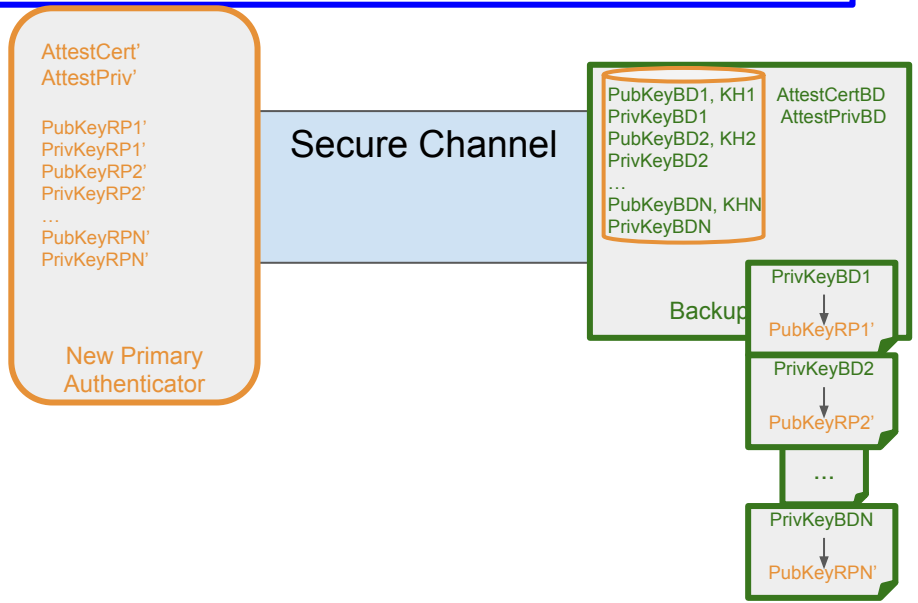
* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.



PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

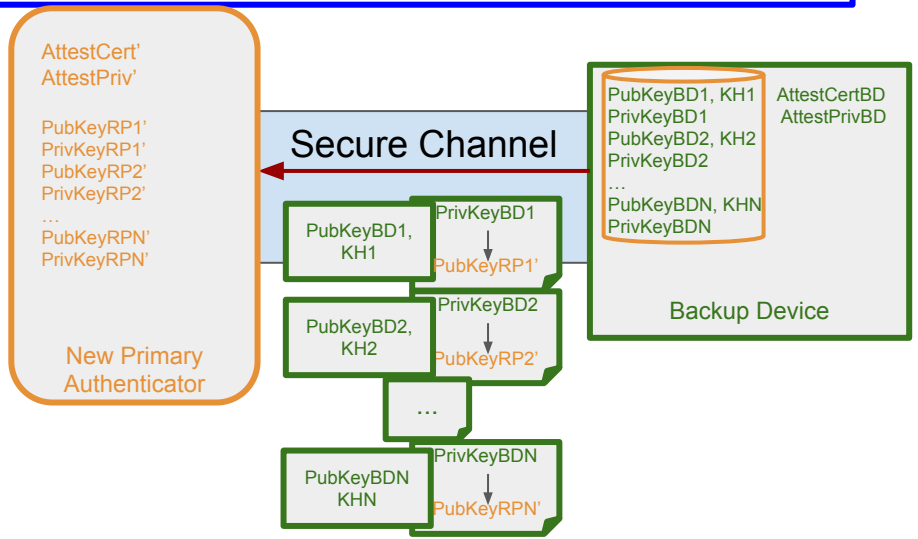
* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.



PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

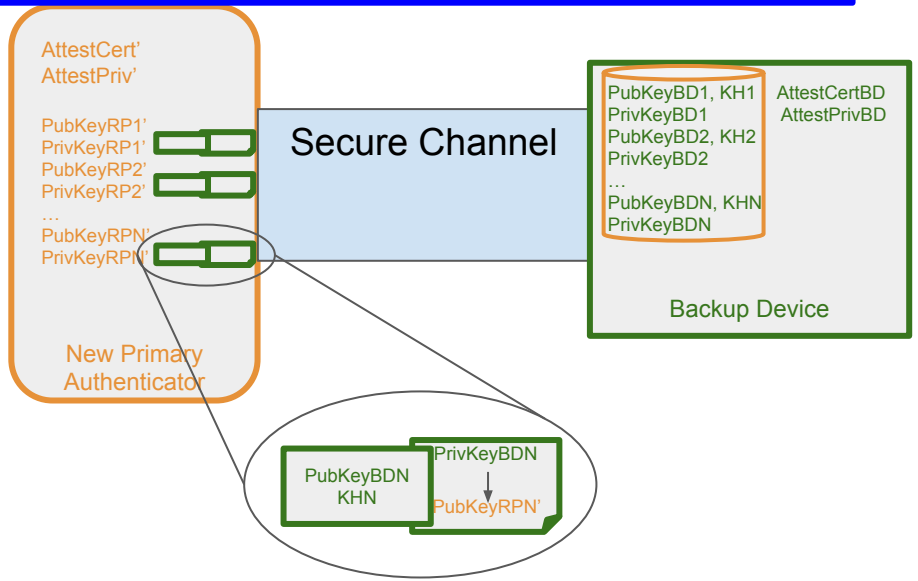
* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.



PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

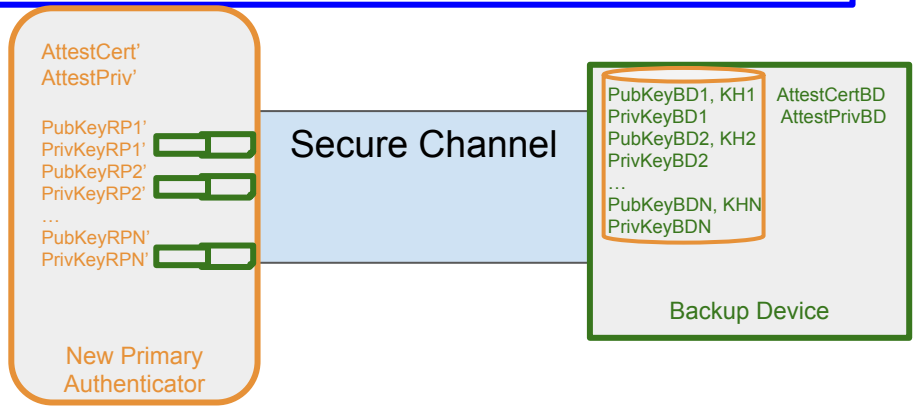
* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.



PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

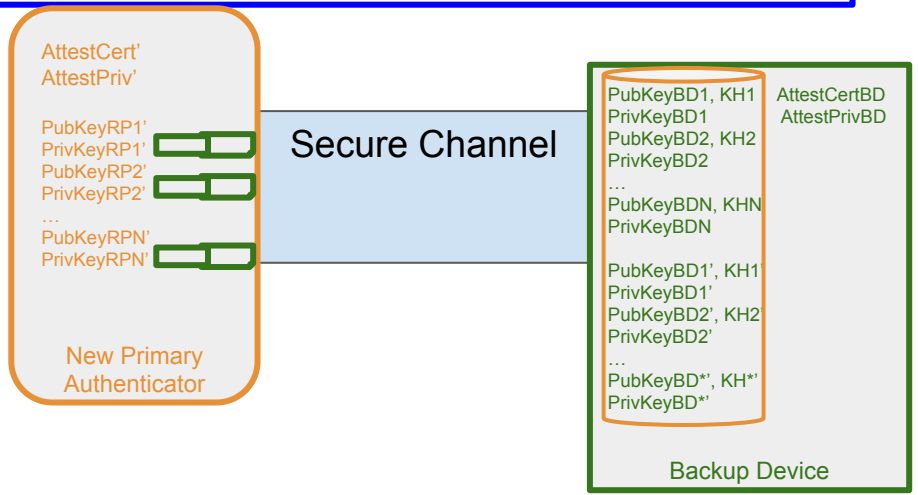
* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.



PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

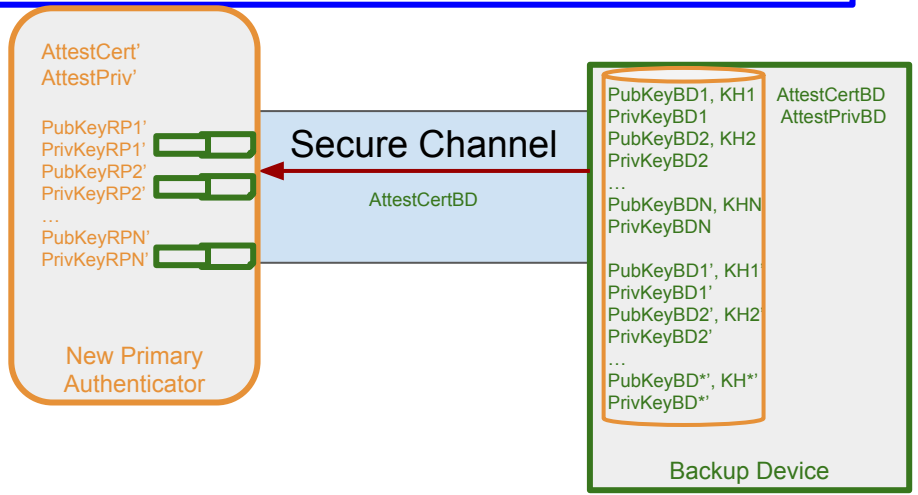
* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.



PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

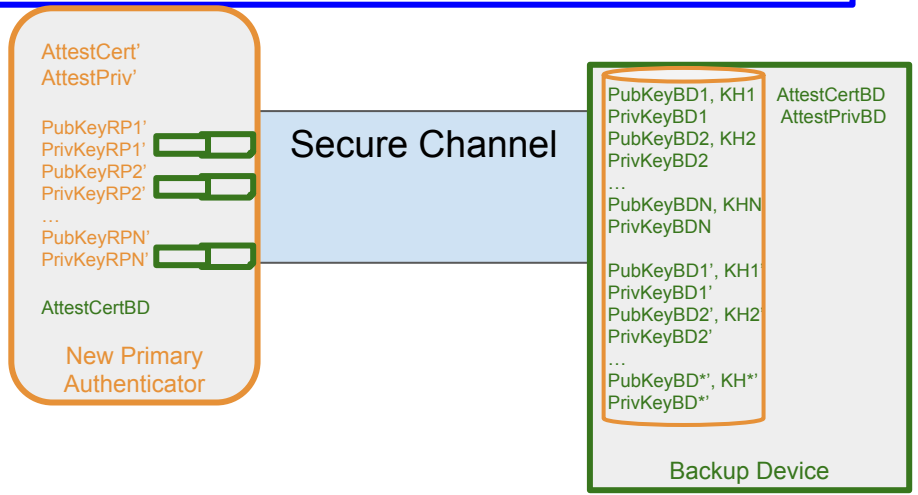
* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.



PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

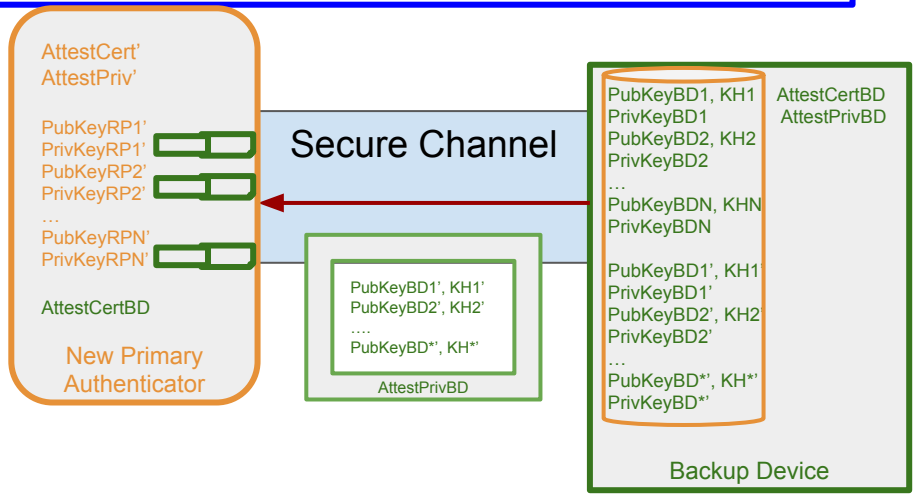
* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.



PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

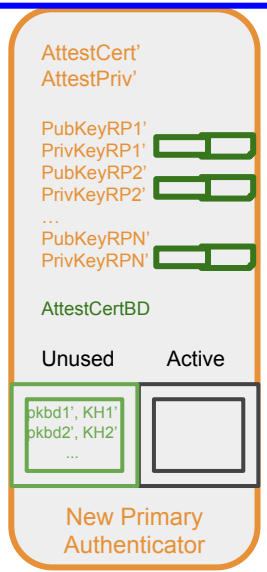
* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.

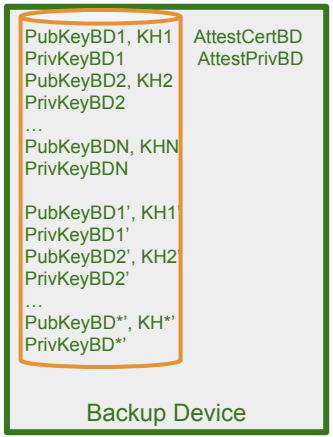


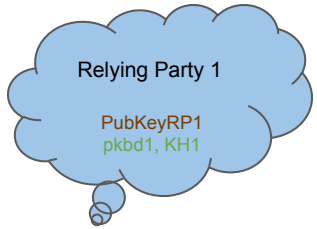
PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.



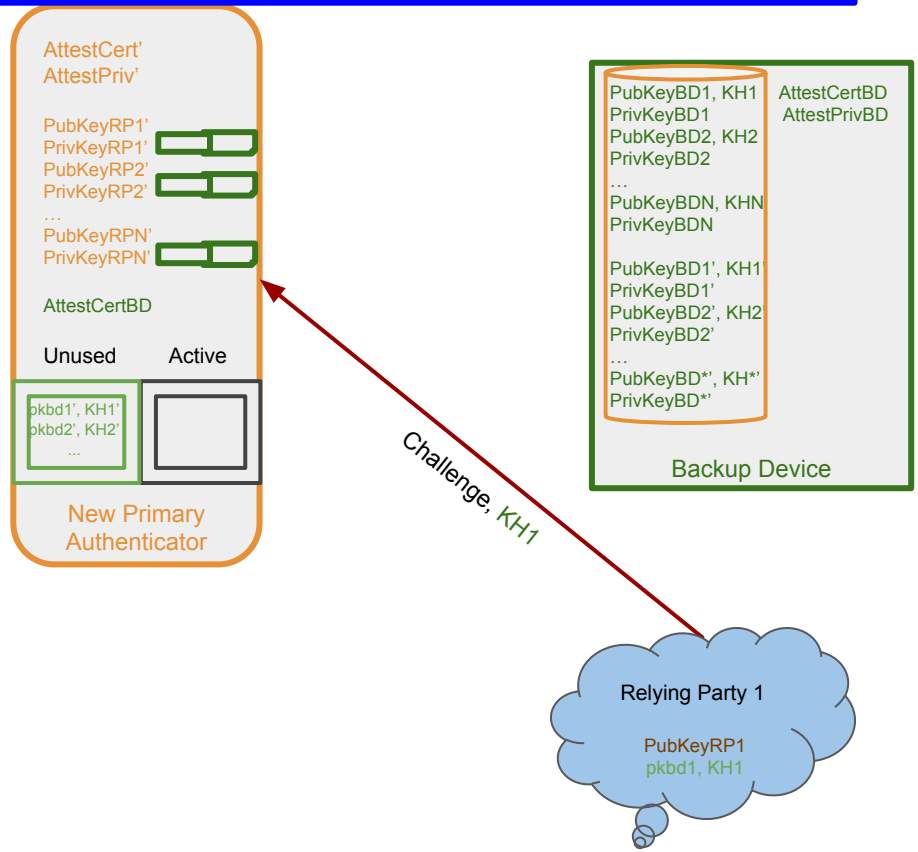




PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

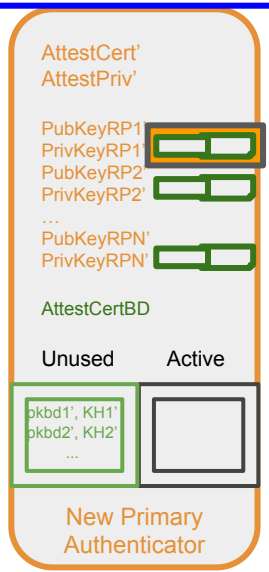
* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.

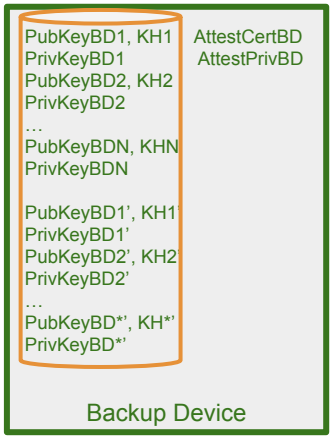


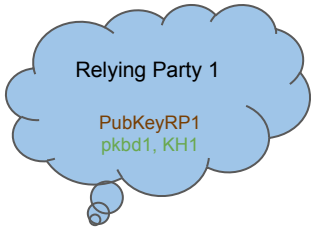
PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.



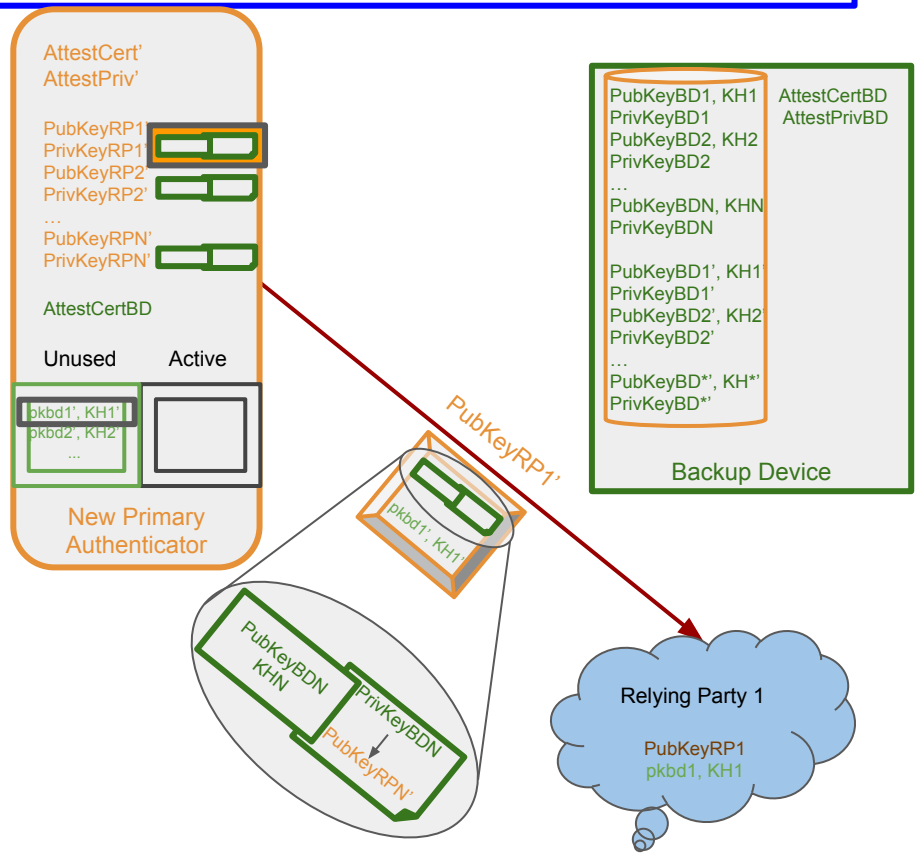




PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

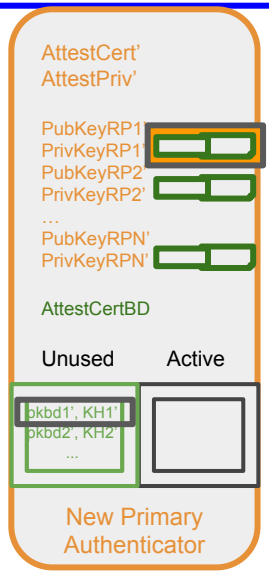
* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.

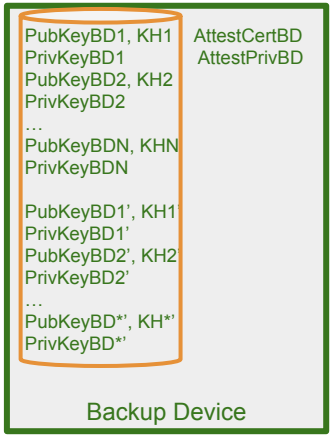


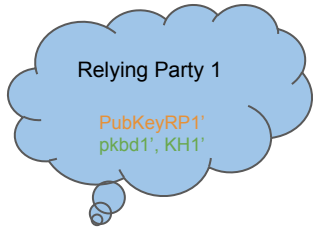
PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.



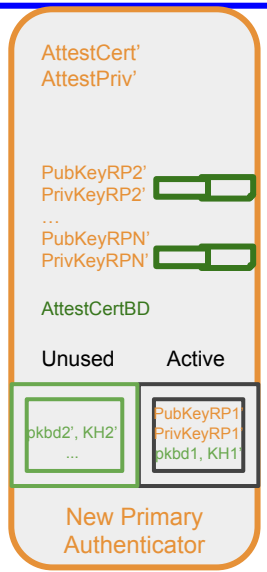


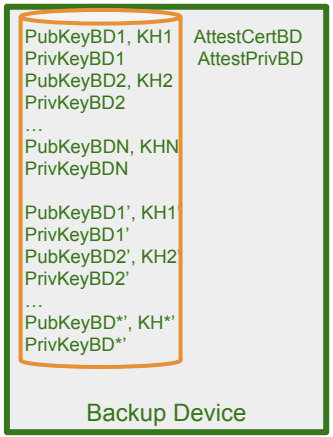


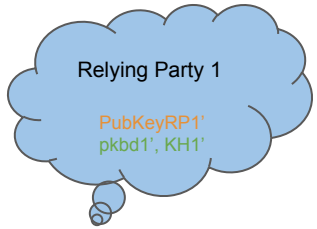
PSK Solution - Recovery (Technical)

1. Create a secure channel between New Primary Authenticator and Backup Device.
2. User selects “Recover from Old Primary Authenticator”
3. Backup Device looks up how many keys it gave to the Old Primary Authenticator (say it created and provided N total keys to the Old Primary Authenticator), and sends that integer N to the New Primary Authenticator.
4. New Primary Authenticator creates N corresponding key pairs and sends all N generated public keys to the Backup Device.
5. Backup Device signs a delegation from each of its private keys associated with the Old Primary Authenticator to one of the public keys given it it by the New Primary Authenticator and internally re-associates its key pair with the New Primary Authenticator.
6. Backup Device sends each of those N public keys, associated key handles, and delegations to the New Primary Authenticator
7. As in Setup, Backup also device generates (* some number) of key pairs and corresponding key handles for future Registrations.
8. Backup Device sends its Attestation Certificate to the Primary Authenticator
9. Backup Device sends all generated public keys to the Primary Authenticator, each signed with its Attestation Private Key.
10. At next login, the RP sends a challenge to the Key Handle it knows, KH1. The New Primary Authenticator Recognizes that key handle and responds with the recovery message, signed with PrivKeyRP1’, and a new backup public key and key handle pair.
11. Primary Authenticator and Relying Party 1 update storage to replace the old keys/handles with the new ones.

* As in Setup, the Backup Device should generate enough key pairs to last for all registrations performed by the New Primary Authenticator.







PSK Solution - Transfer Access (Replacing Devices)

PSK Solution - Transfer Access (Replacing Devices)

- Transfer Access should be straightforward, as in the Transfer Access Protocol
  - Old Primary simply conveys stored information from the Backup to the new Primary Authenticator
  - New Primary can Register as normal
- Transferring to a new Backup authenticator should be simple as well. There are two options
  - User can transfer access from the old backup to the new backup by creating a secure channel directly between the two, without Phone A.
  - User can “switch” backup devices by creating a secure channel between the new Backup Device and the Primary Authenticator.
    - Primary gets new backup keys for each of its existing accounts
    - Primary gets fresh “unused” keys for future registrations
    - Primary must inform the RPs of the change on the next visit and then can delete information about the old backup

Appendix

Information regarding these slides and other proposals at:

https://docs.google.com/document/d/1tRLbXYLb9Z65QqhOX7v9D-aq_RUODyn5oALpCXj46K8/edit?usp=sharing