Passwordless Authentication

Peter Balcirak, CESNET & Masaryk University
Authentication

- Validation process of proclaimed identity
- Factors
  - Something you **know**
  - Something you **have**
  - Something you **are**
Passwords

- Easy to use
- Easy to implement
- Relatively secure on server side
- Not secure on user side
Passwords - Easy to break

Email*
422570@muni.cz

Password *
password

Confirm password *
password

Password must be at least 8 characters long. Please avoid using accented characters. It might not be supported by all backend components and services.
Passwords - Easy to break

1. Password must be at least 8 characters long. Please avoid using accented characters. It might not be supported by all backend components and services.
Passwords - Easy to break

Password must be at least 8 characters long. Please avoid using accented characters. It might not be supported by all backend components and services.
Passwords - Easy to forget

Email*
422570@muni.cz

Password *
1kuX3mdk0Lxr

Confirm password *
1kuX3mdk0Lxr

Password must:

- contain only printable (non-accented) characters
- be at least 12 characters long
- consist of at least 3 of 4 character groups:
  - lower-case letters
  - upper-case letters
  - digits
  - special characters
Passwords - Easy to forget

MUNI Unified Login

UCO
422570

Primary password

Remember me

LOG IN

› I have trouble logging in

© Masaryk University
The MUNI Unified Login service is provided by Institute of Computer Science
Passwords - Easy to forget

Wrong UČO or password

UČO

422570

Primary password

Remember me

LOG IN

I have trouble logging in
Passwords - Phishing
What can we do?

- Increase password requirements
- Implement weak pass check
- Force to use password managers
What can we do?

- Increase password requirements
- Implement weak pass check
- Force to use password managers
- Enable Multi-Factor Authentication (MFA)
MFA

- Combines more authentication factors (at least 2)
- Usually something you know + something you have
- Improves security
- Should not be difficult for users
MFA - User flow
MFA - User flow
MFA - User flow
MFA - User flow
MFA - Attacker flow
MFA - Attacker flow
MFA - Attacker flow
MFA - Attacker flow
MFA - Our solution

- Built on top of the PrivacyIDEA
  - Open-source project
  - Easy to integrate and extend
  - Large community
  - UI for token management
  - API for authentication
- Solution follows the REFEDS MFA Profile
- Currently supports TOTP and WebAuthn
MFA - Architecture

IdP

SAML (SimpleSAMLphp)

PrivacyIDEA module

Authswitcher module

PrivacyIDEA (Docker container)

OIDC, OAuth 2.0 (MitreID)
MFA - PrivacyIDEA SimpleSAMLphp module
MFA - Authswitcher SimpleSAMLphp module
MFA - Token management
MFA - Authentication page

Multi-factor authentication

One time code

Enter a verification code from authenticator app or a recovery code.

One time code

© Masaryk University
The MUNI Unified Login service is provided by Institute of Computer Science
MFA - Services management (end user)
MFA - Summary

- Is it secure?
- Is it user friendly?
- Is it necessary?
MFA - User flow
Passwordless - User flow
Passwordless

- Based on something user has
- Trade-off between security and user experience?
- Eliminates the most common vector of attacks
Passkeys

• Created by W3C and FIDO alliance
• Built on top of the WebAuthn standard
• Can be synchronised between devices
• Does not necessarily require username
• Highly phishing resistant
Passkeys - Registering a new passkey
Passkeys - Registering a new passkey

1. Access service
Passkeys - Registering a new passkey

1. Access service

2. Sign-in
Passkeys - Registering a new passkey

1. Access service
2. Sign-in
3. Send response with challenge
Passkeys - Registering a new passkey

1. Access service
2. Sign-in
3. Send response with challenge
4. Send response
Passkeys - Registering a new passkey

1. Access service
2. Sign-in
3. Send response with challenge
4. Send response
5. Create new passkey
Passkeys - Registering a new passkey

1. Access service
2. Sign-in
3. Send response with challenge
4. Send response
5. Create new passkey
6. Create passkey and sign challenge
Passkeys - Registering a new passkey

1. Access service
2. Sign-in
3. Send response with challenge
4. Send response
5. Create new passkey
6. Create passkey and sign challenge
7. Send signature and public key
Passkeys - Registering a new passkey

1. Access service
2. Sign-in
3. Send response with challenge
4. Send response
5. Create new passkey
6. Create passkey and sign challenge
7. Send signature and public key
8. Verify signature and save public key
Passkeys - Sign in with a passkey
Passkeys - Sign in with a passkey

1. Access service
Passkeys - Sign in with a passkey

1. Access service

2. Request sign-in page
Passkeys - Sign in with a passkey

1. Access service

2. Request sign-in page

3. Sign-in page with challenge
Passkeys - Sign in with a passkey

1. Access service

2. Request sign-in page

3. Sign-in page with challenge

4. Sign-in page
Passkeys - Sign in with a passkey

1. Access service
2. Request sign-in page
3. Sign-in page with challenge
4. Sign-in page
5. Passkey sign-in with verification
Passkeys - Sign in with a passkey

1. Access service
2. Request sign-in page
3. Sign-in page with challenge
4. Sign-in page
5. Passkey sign-in with verification
6. Sign challenge
Passkeys - Sign in with a passkey

1. Access service

4. Sign-in page

5. Passkey sign-in with verification

2. Request sign-in page

3. Sign-in page with challenge

6. Sign challenge

7. Send signature
Passkeys - Sign in with a passkey

1. Access service

2. Request sign-in page

3. Sign-in page with challenge

4. Sign-in page

5. Passkey sign-in with verification

6. Sign challenge

7. Send signature

8. Verify signature
Passkeys - Sign in with a passkey on other device
Passkeys - Sign in with a passkey on other device

1. Access service
Passkeys - Sign in with a passkey on other device

1. Access service

2. Request sign-in page
Passkeys - Sign in with a passkey on other device

1. Access service
2. Request sign-in page
3. Sign-in page with challenge
Passkeys - Sign in with a passkey on other device

1. Access service

2. Request sign-in page

3. Sign-in page with challenge

4. Show QR code
Passkeys - Sign in with a passkey on other device

1. Access service
2. Request sign-in page
3. Sign-in page with challenge
4. Show QR code
5. Scan QR and sign-in
Passkeys - Sign in with a passkey on other device

1. Access service
2. Request sign-in page
3. Sign-in page with challenge
4. Show QR code
5. Scan QR and sign-in
6. Sign challenge
Passkeys - Sign in with a passkey on other device

1. Access service
2. Request sign-in page
3. Sign-in page with challenge
4. Show QR code
5. Scan QR and sign-in
6. Sign challenge
7. Send signature
Passkeys - Sign in with a passkey on other device

1. Access service
2. Request sign-in page
3. Sign-in page with challenge
4. Show QR code
5. Scan QR and sign-in
6. Sign challenge
7. Send signature
8. Send signature
Passkeys - Sign in with a passkey on other device

1. Access service
2. Request sign-in page
3. Sign-in page with challenge
4. Show QR code
5. Scan QR and sign-in
6. Sign challenge
7. Send signature
8. Send signature
9. Verify signature
Passkeys - Situation on the market

- Many commercial vendors **support** passkeys
- SimpleSAMLphp **supports** passkeys
- Shibboleth does **not support** passkeys (in progress)
- Keycloak **supports** passkeys
- PrivacyIDEA does **not fully support** passkeys yet
Passkeys - Our approach

- Aim to build it on top of our MFA solution
- Need to replace some underlying components first
- Need to solve the UX part before implementation
Passkeys - How to do the transition?

- Change our authentication gateway
- Need to support both approaches
- Passkeys through a button or autofill
- Need to educate users
Which multi-factor authentication solution is right for me?

We know that it can be difficult to understand all the options for multi-factor authentication. It can be challenging to decide which method, device or software to use.

However, this guide can help you with that. Come and give it a try!
Passkeys - How to do the recovery and onboarding?

• What if users lost their passkey?
• How to onboard new users?
• Password as a backup?
Passkeys - What if passkeys are not enough?

• What if some community requires another factor?
• We need to know which passkey was used as a first factor
• Doable at one IdP
• Difficult when interconnecting separate AAIas
Passkeys - What if an IdP does not support passkeys?

- IdP does not tell us what factor was used
- Need to handled on the AAI side
- Need to navigate users seamlessly
If not passwordless, at least use passwords less