
# Passwordless Linux

## Where are we?

Alexander Bokovoy
Sr. Principal Software Engineer / Red Hat

# Who am I?

- Software engineer at Red Hat
- Focus on identity management and authentication in Red Hat Enterprise Linux and Fedora Project
  - FreeIPA, SSSD, Samba, MIT Kerberos

# What is this talk about?

- Past
- Progress today in FreeIPA, SSSD, and MIT Kerberos
- Future (in Fedora 39 or later)

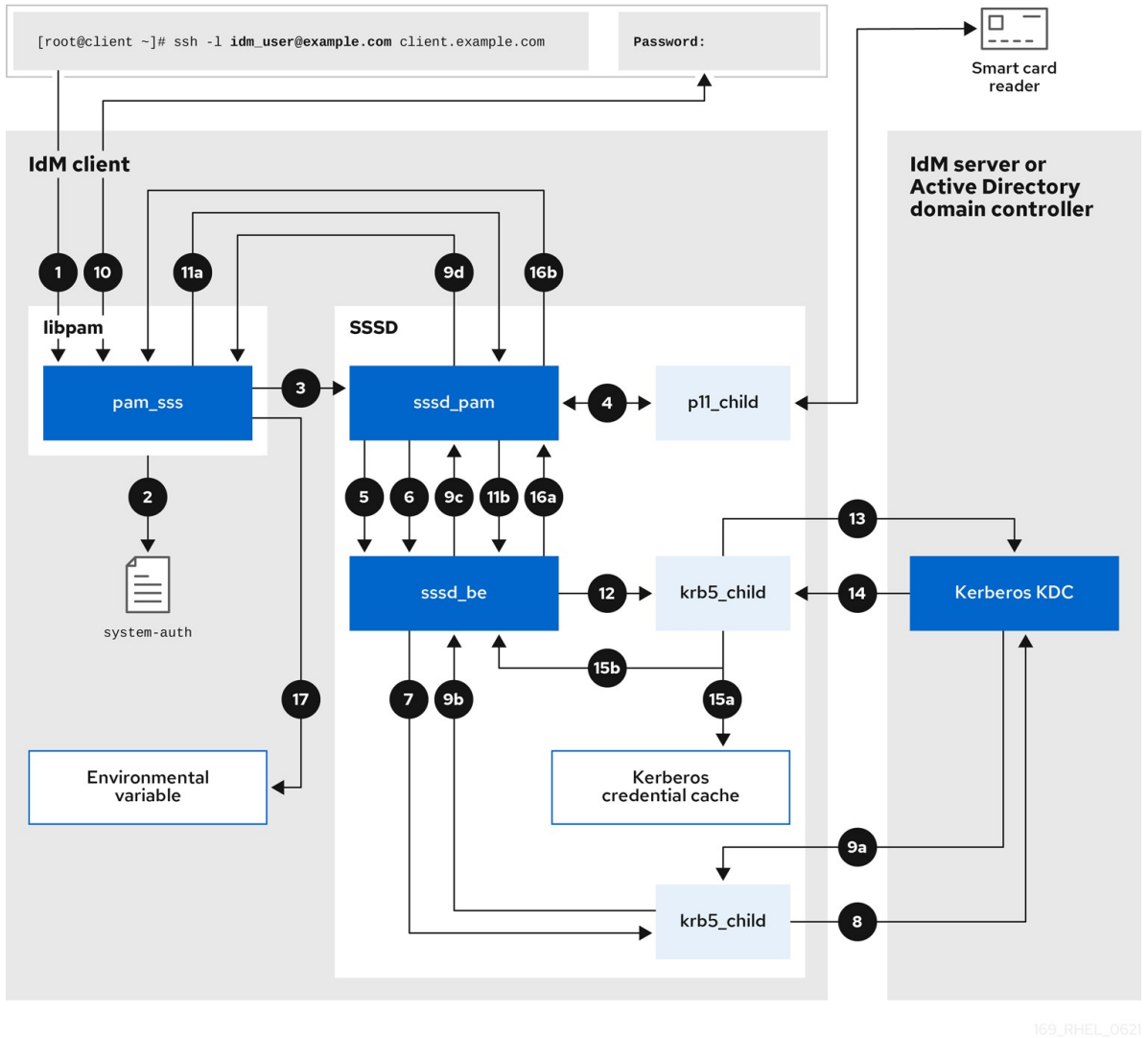
# Past?

- Assumptions
  - Compatible authentication mechanisms
  - Transferrable state of authentication
- Typical approach
  - Login to unlock secrets manager
  - Use of session authentication agent
  - Resource consumption based on the secrets' access
- Application-specific issues

# Kerberos

- 40 years of networking
- Three problems solved
  - Decouple initial authentication from the rest of use cases
  - Transferrable state of authentication
  - Uniform application-level API (GSS-API)
- Initial (pre-)authentication can be passwordless
  - PKINIT (smartcards)

# Authentication with Kerberos




Detailed description is in RHEL IdM guide 'Configuring and managing Identity Management':

8.3. Data flow when authenticating as a user with SSSD in IdM

# Blast from the past

- FOSDEM 2016
  - Fedora 22
  - FreeIPA as single sign-on enterprise environment
  - Single sign-on from GDM to web applications
  - Use of Kerberos for VPN, SSH, network file systems' access

# Change of winds

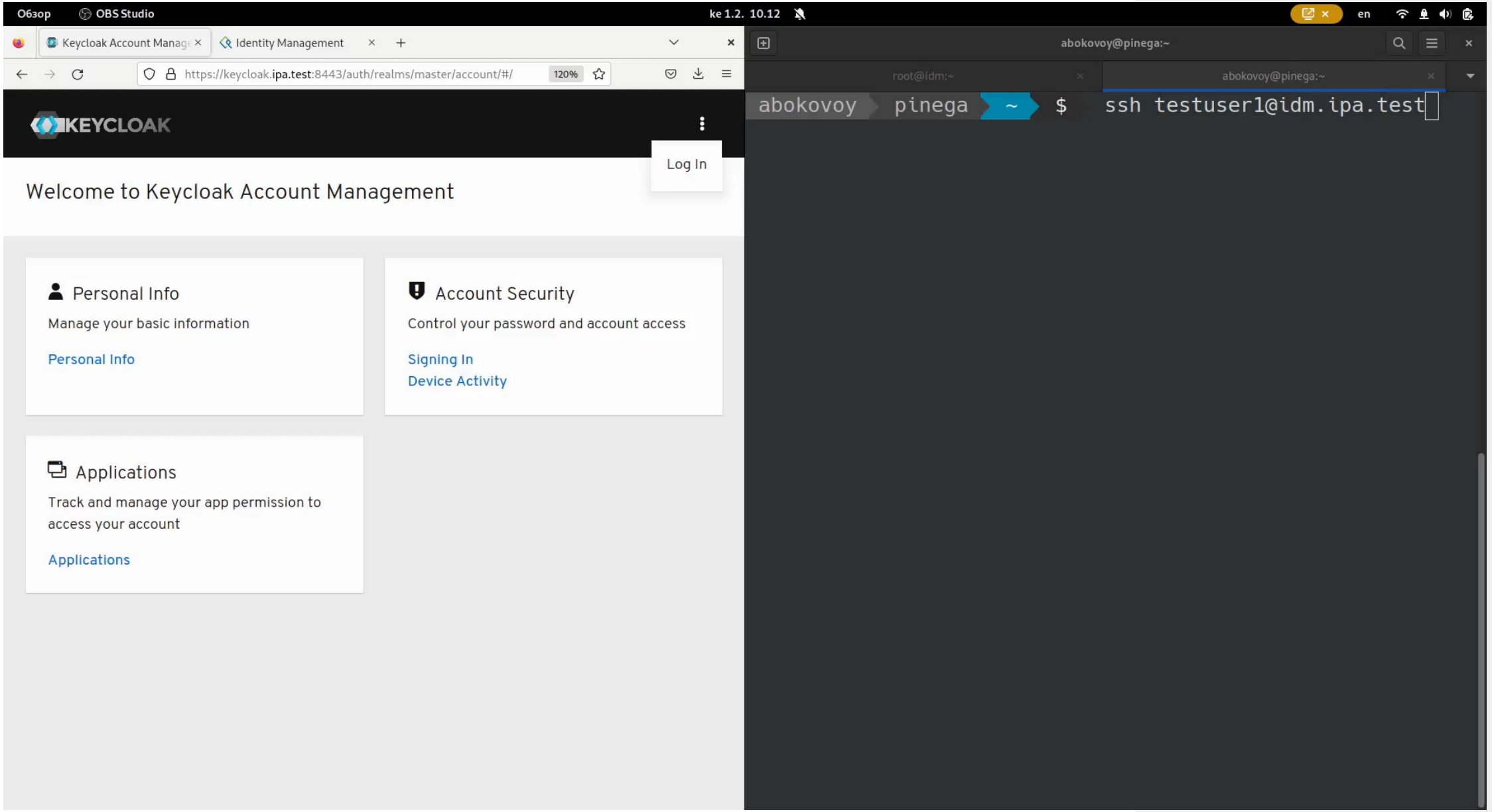
- Infrastructure for applications vs infrastructure for people
- Transition to all-web applications
  - Browser is a new mainframe
  - OAuth 2.0 is a new authentication and authorization king
- BYOA
  - Bridge your own authentication

# Browser is a new mainframe

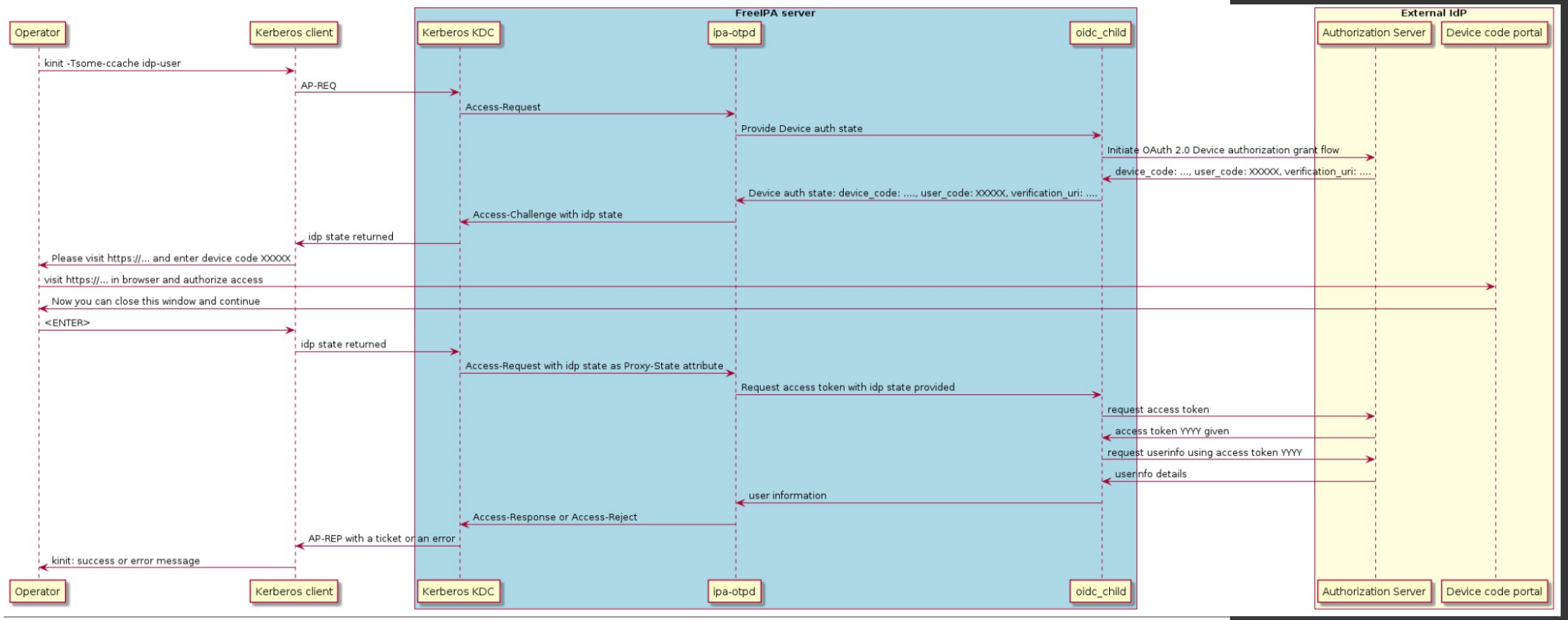
- 2016: captive portals
  - Login over network needs … network access
  - Network access needs captive portal handling
    - Before login to the desktop/laptop
- 2023: OAuth 2.0 identity provider before login
  - Login with OAuth 2.0 implies user browser interaction
  - Still no browser view access prior to GDM login
    - Security issues with untrusted content

# Somewhere else browser

- Remote access
  - We already have **other** system to run browser
  - Instruct user to visit OAuth 2.0 IdP end-point
    - Device authorization grant flow
- FreeIPA 4.9.10 or later
  - SSSD extends MIT Kerberos pre-authentication mechanism
  - Works with almost all public OAuth 2.0 IdPs
    - Requires Device authorization grant flow (RFC 8628)
- [demo]



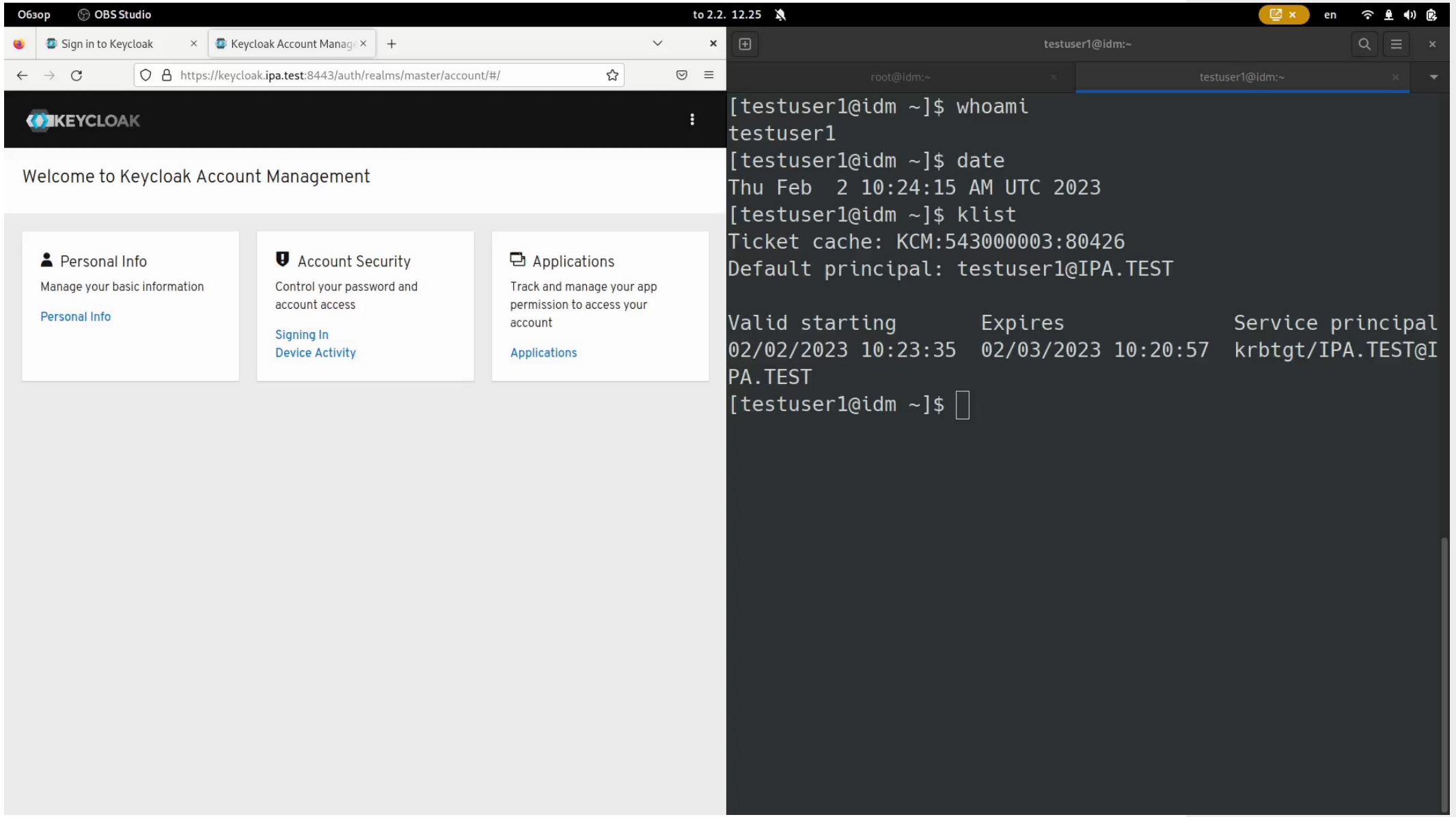
# Authentication with external IdP in Kerberos



# Webauthn/FIDO2

- OAuth 2.0 IdP
  - May already support Webauthn/FIDO2 tokens
  - May already allow login to itself with Webauthn
- FreeIPA in Fedora 37
  - Login with external IdP authentication
  - External IdP uses Webauthn tokens
  - Passwordless login to Linux console

## Passwordless Linux – where are we?

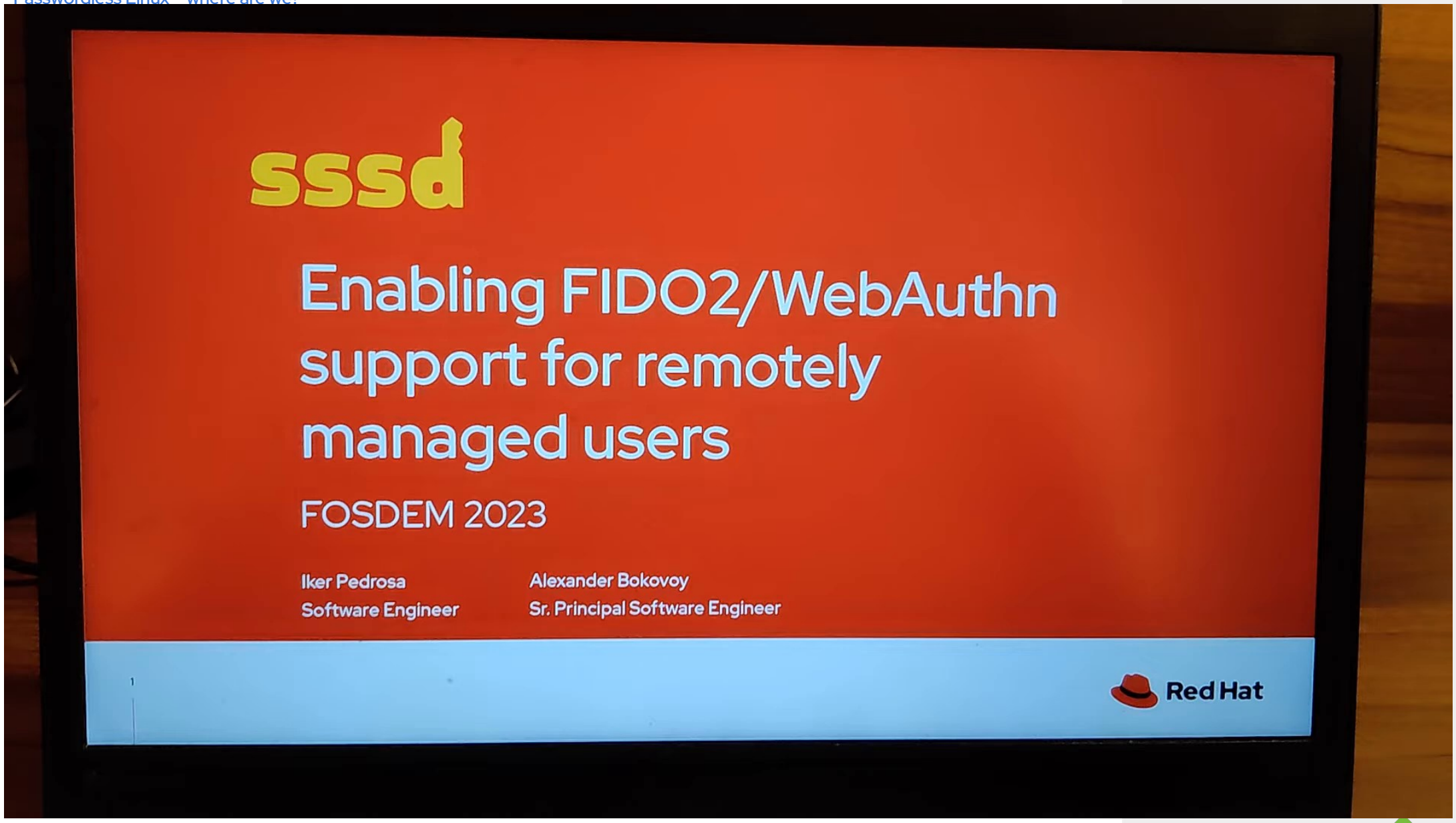





FreeIPA
Open Source Identity Management Solution

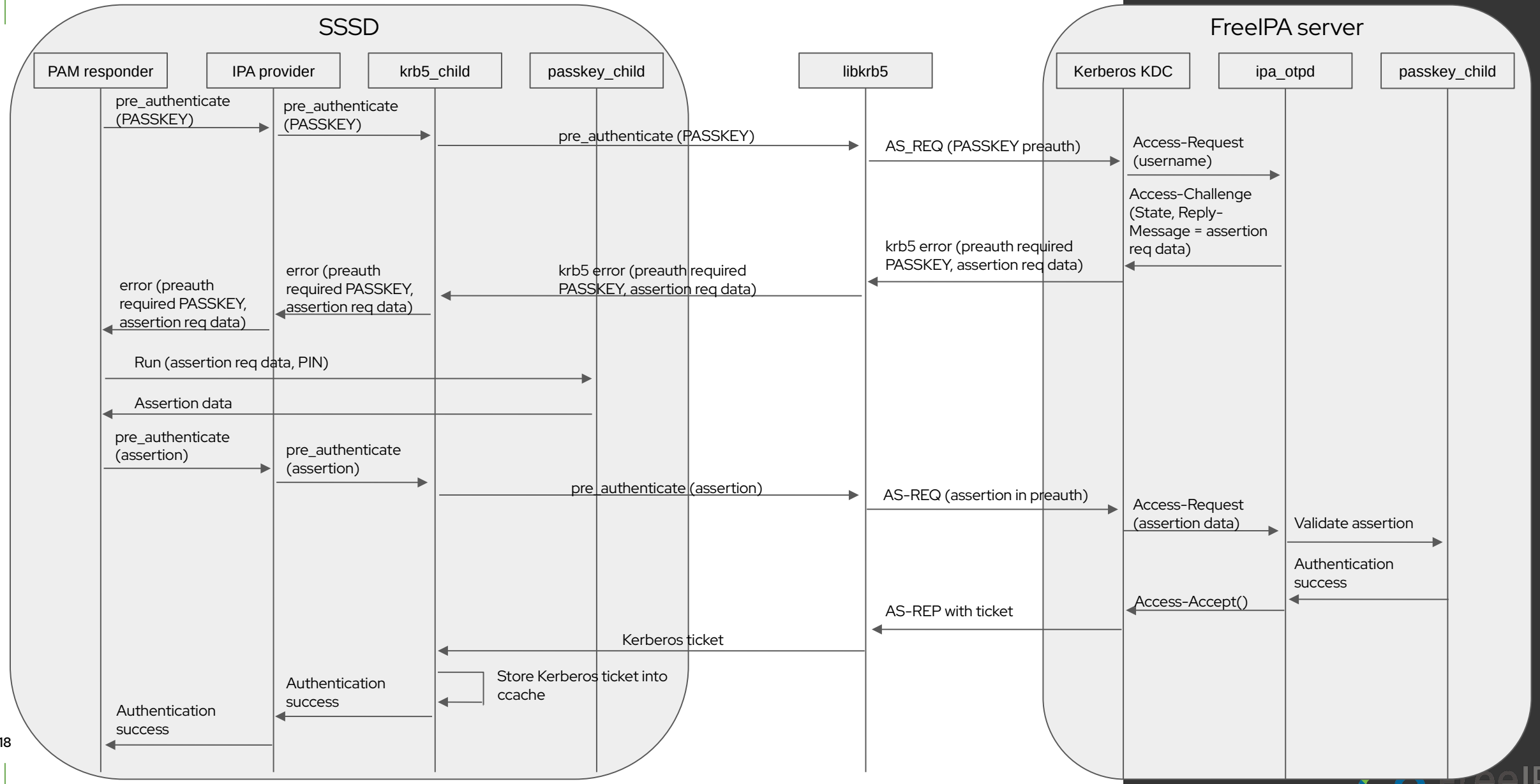
# Webauthn/FIDO2

- Can we get away from the networking services?
  - Local FIDO2 authentication
- [demo]


sssd
Enabling FIDO2/WebAuthn support for remotely managed users
FOSDEM 2023
Iker Pedrosa
Software Engineer
Alexander Bokovoy
Sr. Principal Software Engineer
1
Red Hat

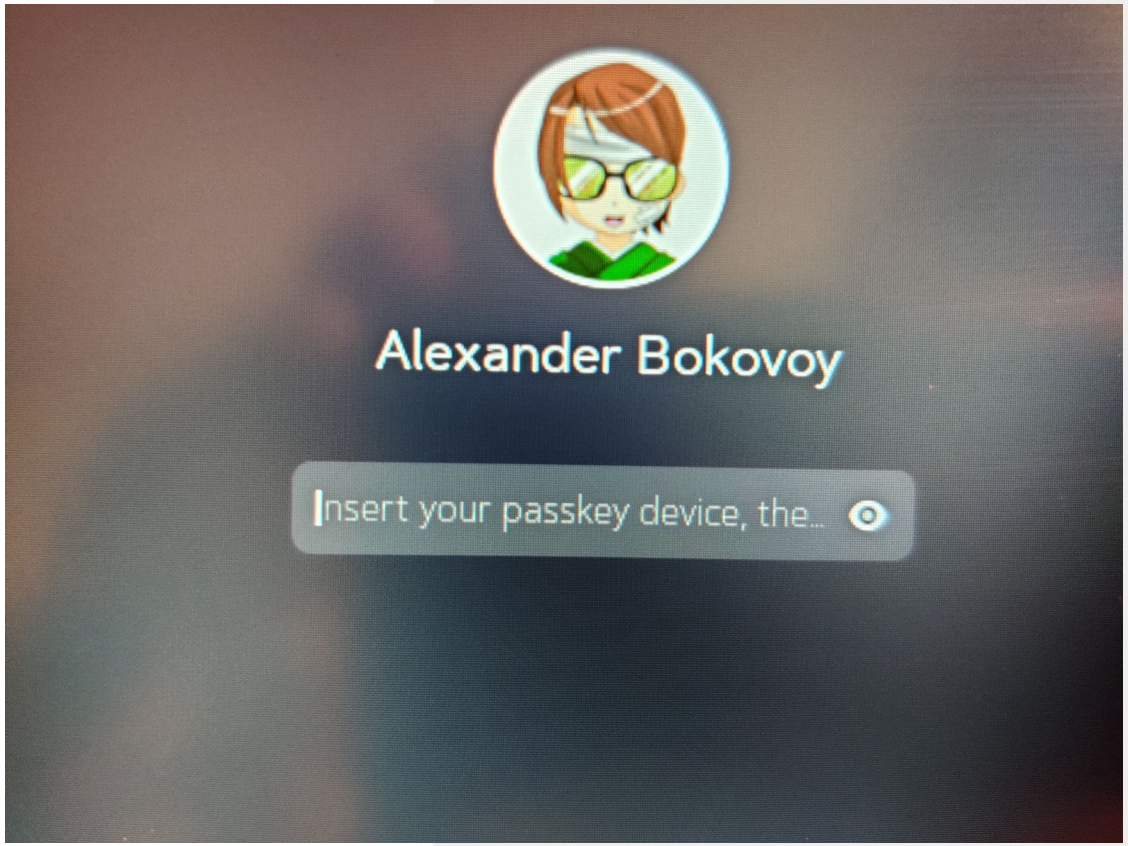
# Webauthn/FIDO2

- Combine local FIDO2 and Kerberos
  - Similar to OAuth 2.0 IdP integration
  - Work in progress at the moment


SSSD
FreeIPA server
PAM responder
IPA provider
krb5_child
passkey_child
libkrb5
Kerberos KDC
ipa_otpd
passkey_child
pre_authenticate (PASSKEY)
pre_authenticate (PASSKEY)
pre_authenticate (PASSKEY)
AS_REQ (PASSKEY preauth)
Access-Request (username)
Access-Challenge (State, Reply-Message = assertion req data)
krb5 error (preauth required PASSKEY, assertion req data)
krb5 error (preauth required PASSKEY, assertion req data)
error (preauth required PASSKEY, assertion req data)
error (preauth required PASSKEY, assertion req data)
Run (assertion req data, PIN)
Assertion data
pre_authenticate (assertion)
pre_authenticate (assertion)
pre_authenticate (assertion)
AS-REQ (assertion in preauth)
Access-Request (assertion data)
Validate assertion
Authentication success
Access-Accept()
AS-REP with ticket
Kerberos ticket
Store Kerberos ticket into ccache
Authentication success
Authentication success

# Desktop integration

- GDM login issues
  - UX issues
  - Multiple authentication methods
  - Passkeys and remote device guidance
- Other graphical environments
- Authentication state preservation



# Distribution integration

- Distribution integration effort
- Upstream projects coordination
- Parallel efforts

FreeIPA
Open Source Identity Management Solution

# Questions?

Images generated with the help of a Stable Diffusion driver using ukiyo-e style prompts