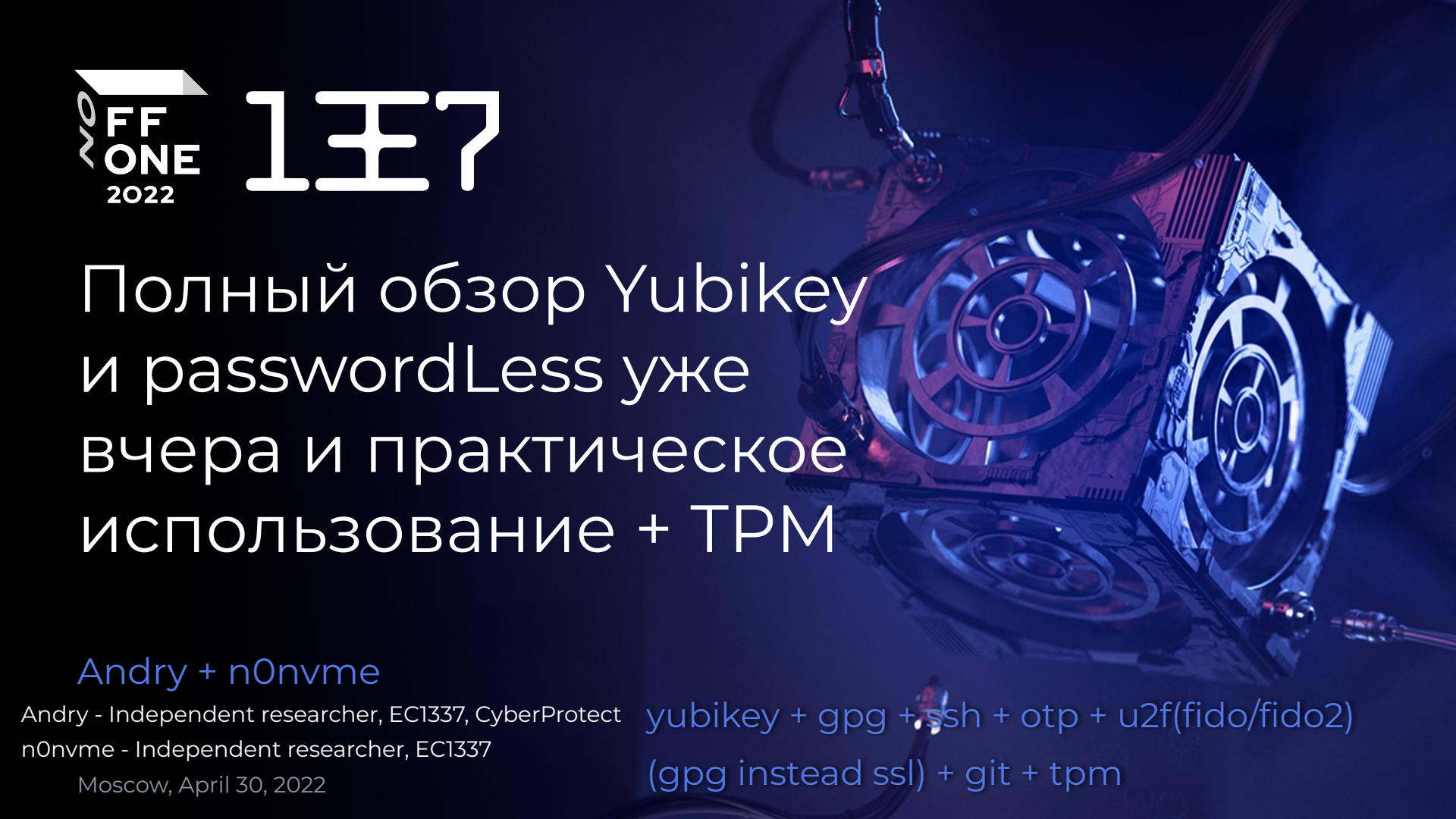


ON FF ONE 2022

1E7

Полный обзор Yubikey и passwordLess уже вчера и практическое использование + TPM

Andry + n0nvme

Andry - Independent researcher, EC1337, CyberProtect

n0nvme - Independent researcher, EC1337

Moscow, April 30, 2022

yubikey + gpg + ssh + otp + u2f(fido/fido2)

(gpg instead ssl) + git + tpm

Галопом по европе - или краткий экскурс по:

yubikey + gpg + ssh + otp + u2f(fido/fido2)
(gpg instead ssl) + git

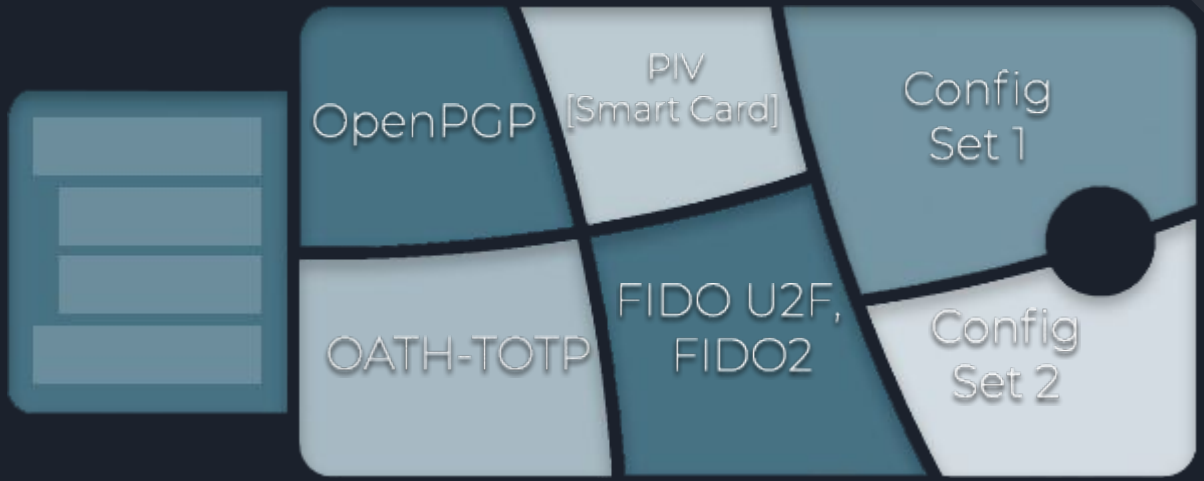
tg @Andryski
tg @n0nvme

Без углубления в:

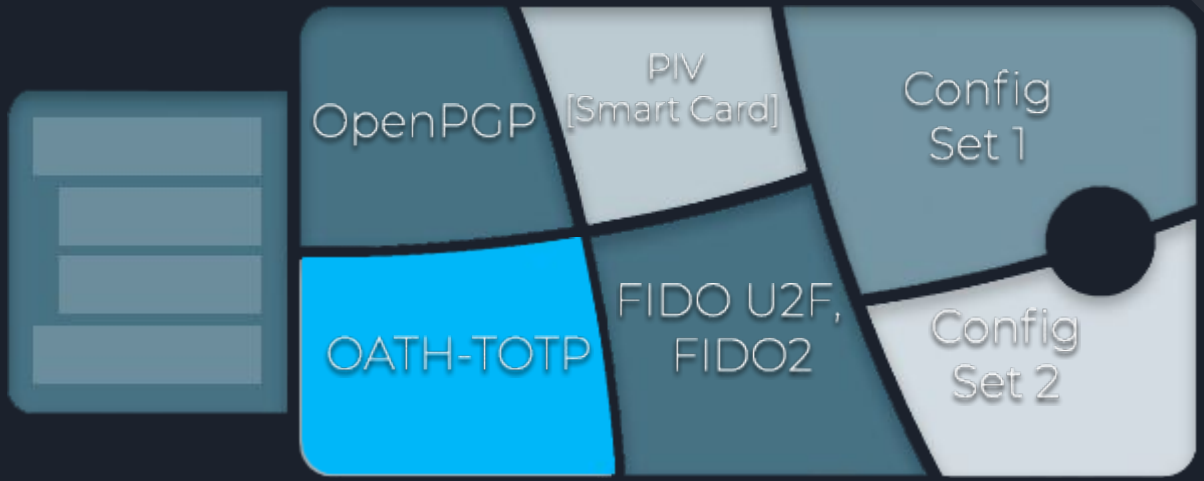
- Правовую часть
- Реализацию

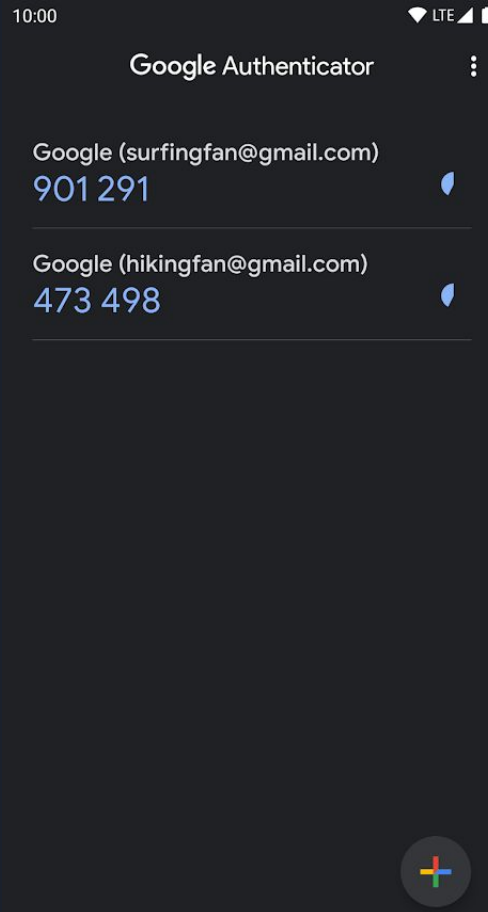
yubico & YubiKey

План/Что мы сегодня узнаем



OTP


OpenPGP
PIV
[Smart Card]
Config
Set 1
OATH-TOTP
FIDO U2F,
FIDO2
Config
Set 2


10:00
LTE
Google Authenticator
Google (surfingfan@gmail.com)
901 291
Google (hikingfan@gmail.com)
473 498

OTP - One-Time Password

HOTP - HMAC-Based One-Time Password
TOTP - Time - Based One-time Password

HOTP - HMAC-Based One-Time Password Algorithm

HMAC - hash-based message authentication code

HMAC = SHA-1(Text1 + Text2)

HOTP - HMAC-Based One-Time Password Algorithm

- Секретный ключ

- Счётчик

HOTP = HMAC( + )

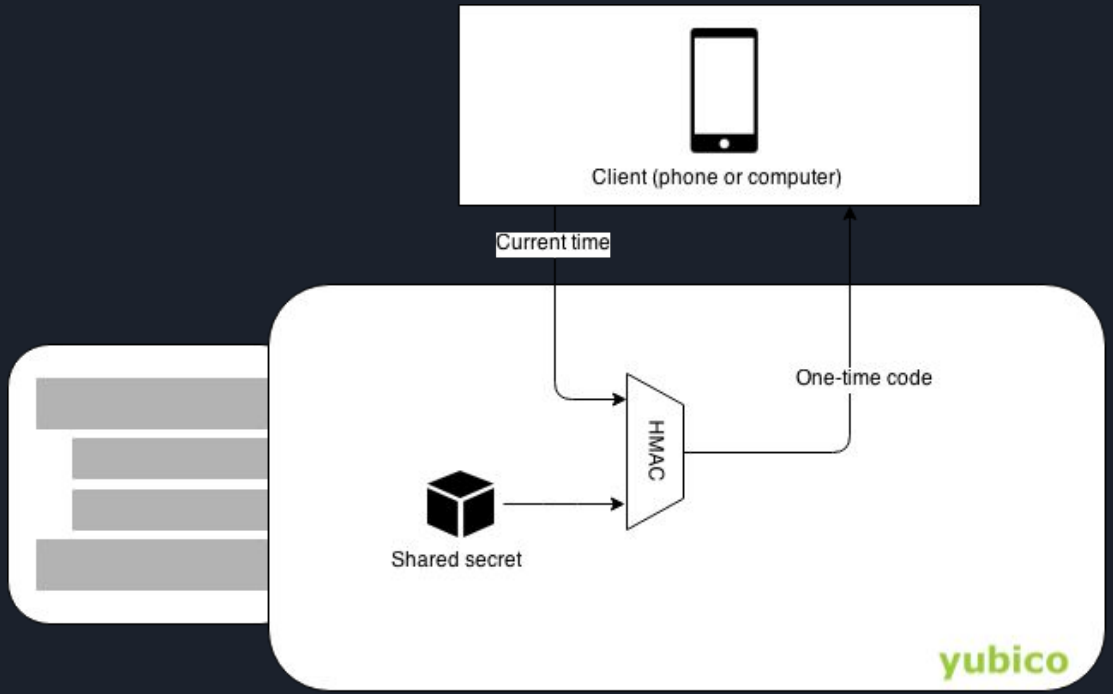
TOTP - Time-based One-time Password Algorithm

- Секретный ключ

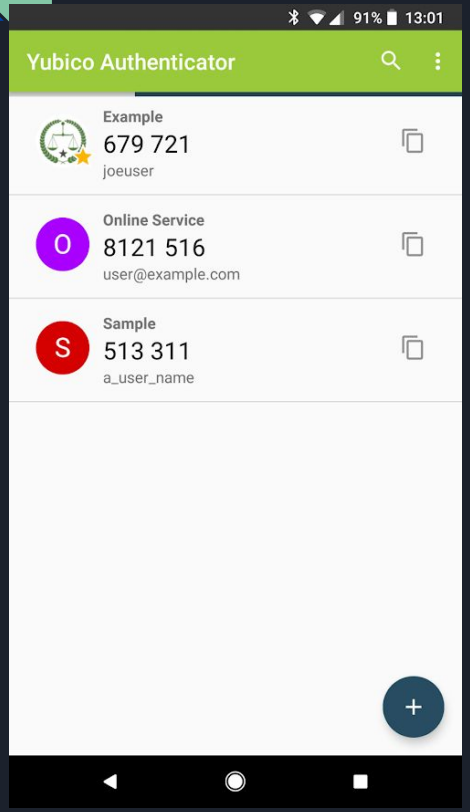
- Время

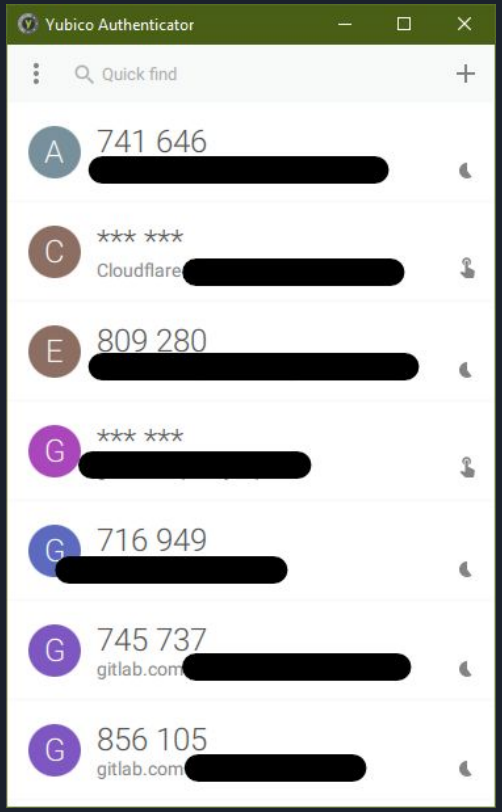
TOTP = HMAC(+)

TOTP


Client (phone or computer)
Current time
One-time code
HMAC
Shared secret
yubico

Yubikey authenticator





KEYCLOAK

Smart Card


OpenPGP
PIV
[Smart Card]
Config
Set 1
OATH-TOTP
FIDO U2F,
FIDO2
Config
Set 2

Smart Card

Slot 9a: PIV Authentication

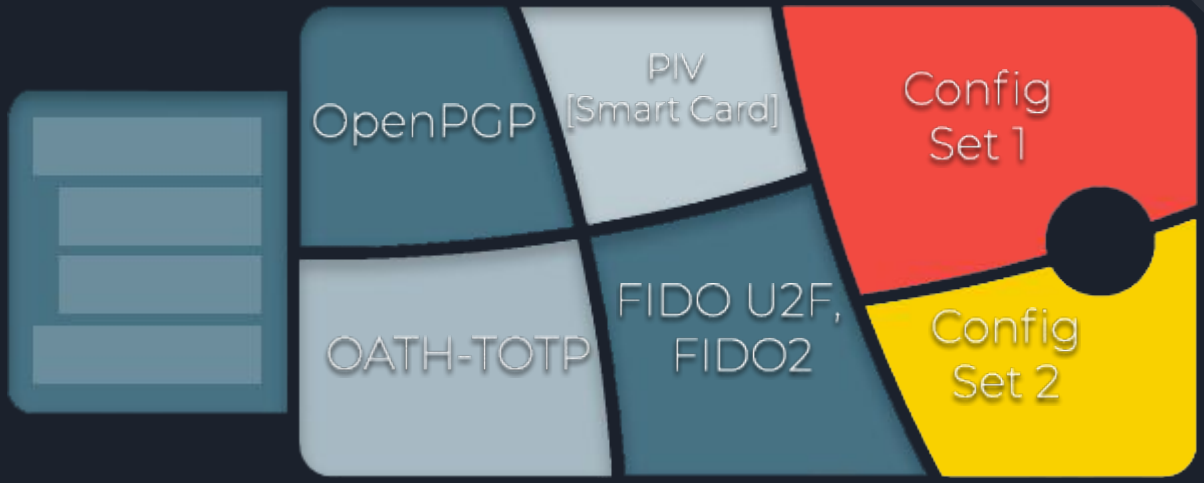
Slot 9c: Digital Signature

Slot 9d: Key Management

Slot 9e: Card Authentication

Slot 82-95: Retired Key Management

Config slots



Config Slots

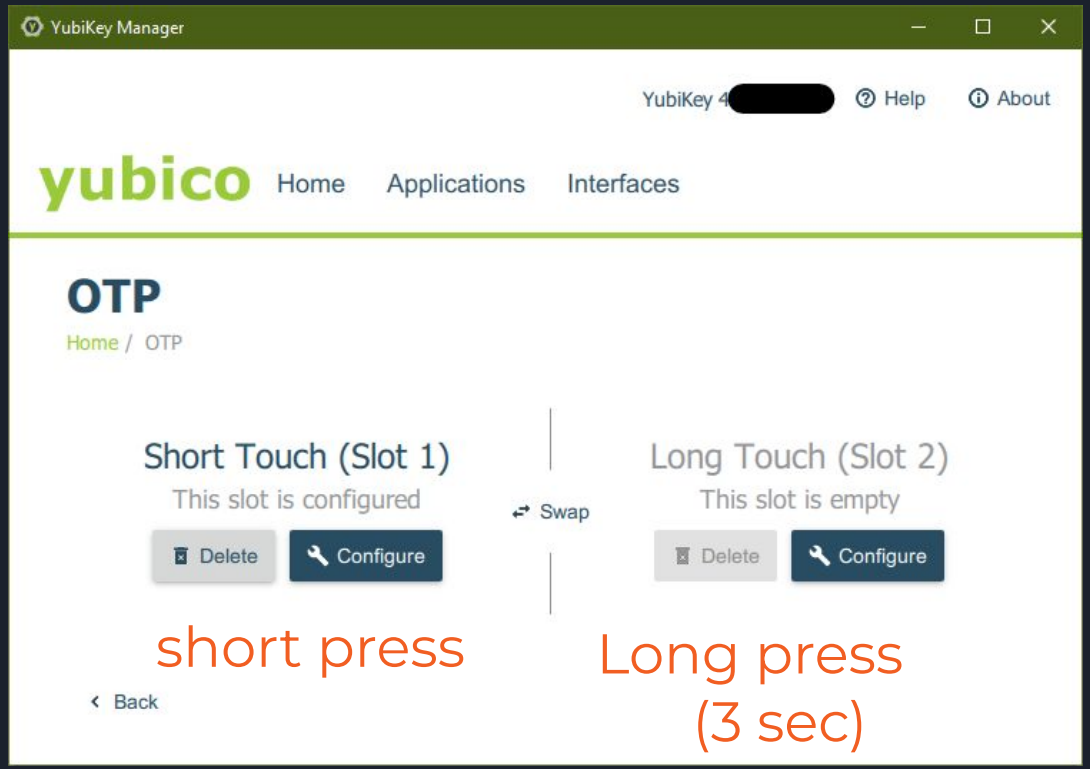
Static password

OATH-HOTP

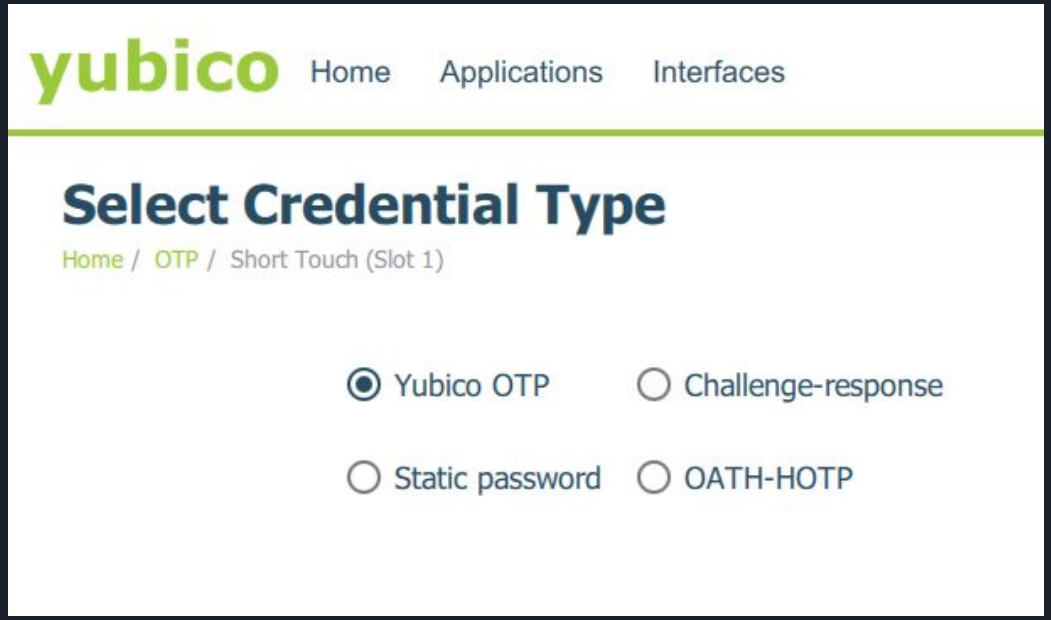
Challenge-response

Yubico OTP

Config slots

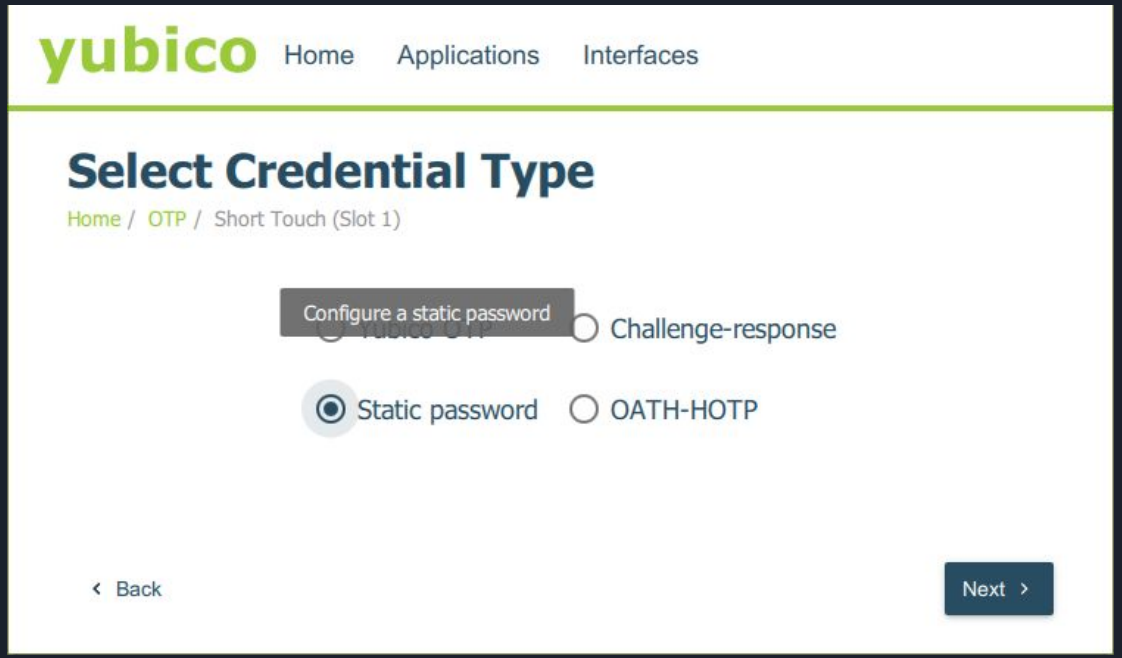


Config slots

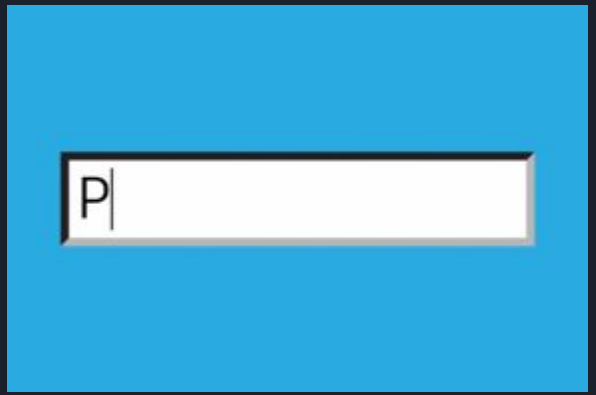


OTP ≠ Yubico OTP

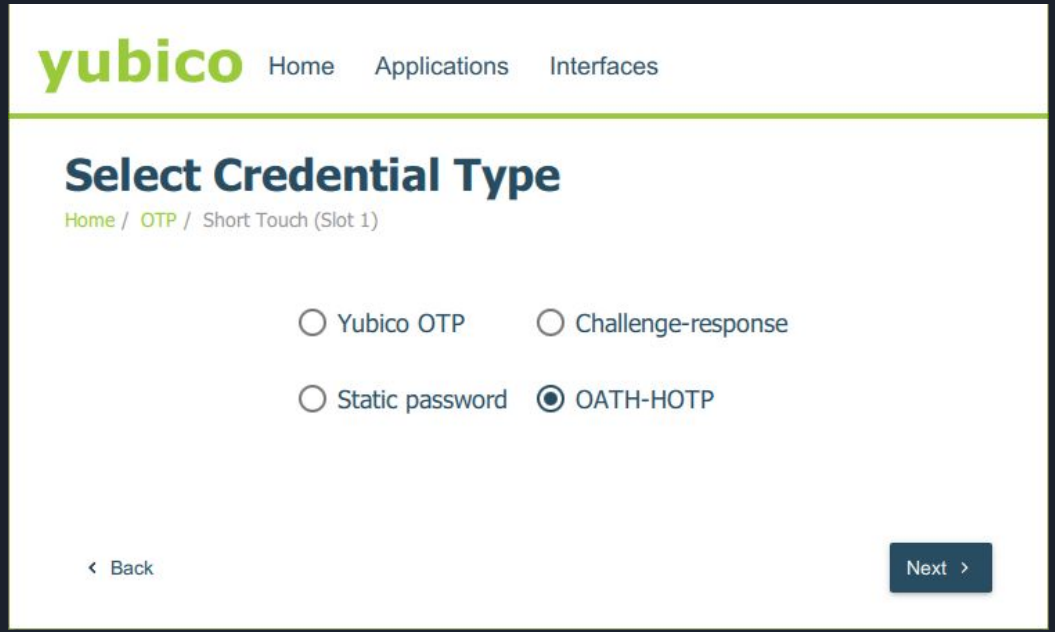
Config slots



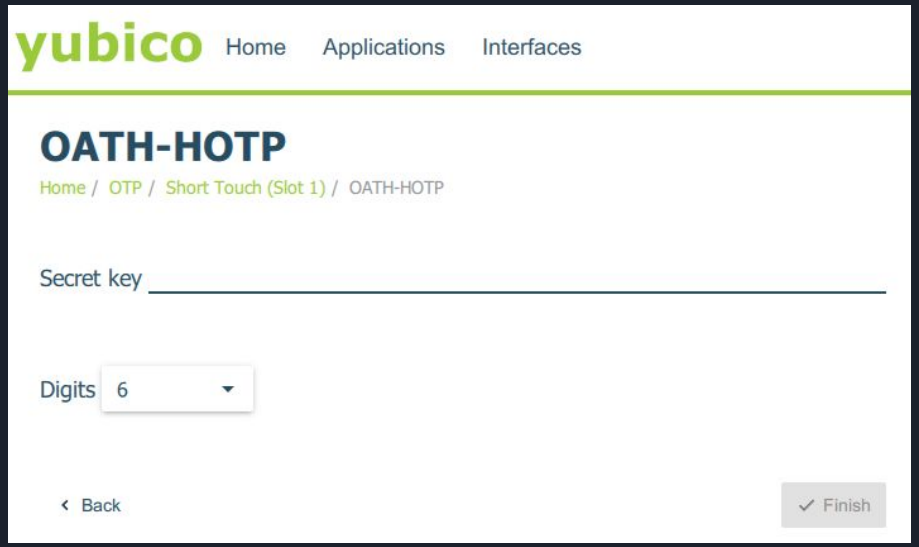
Static Password



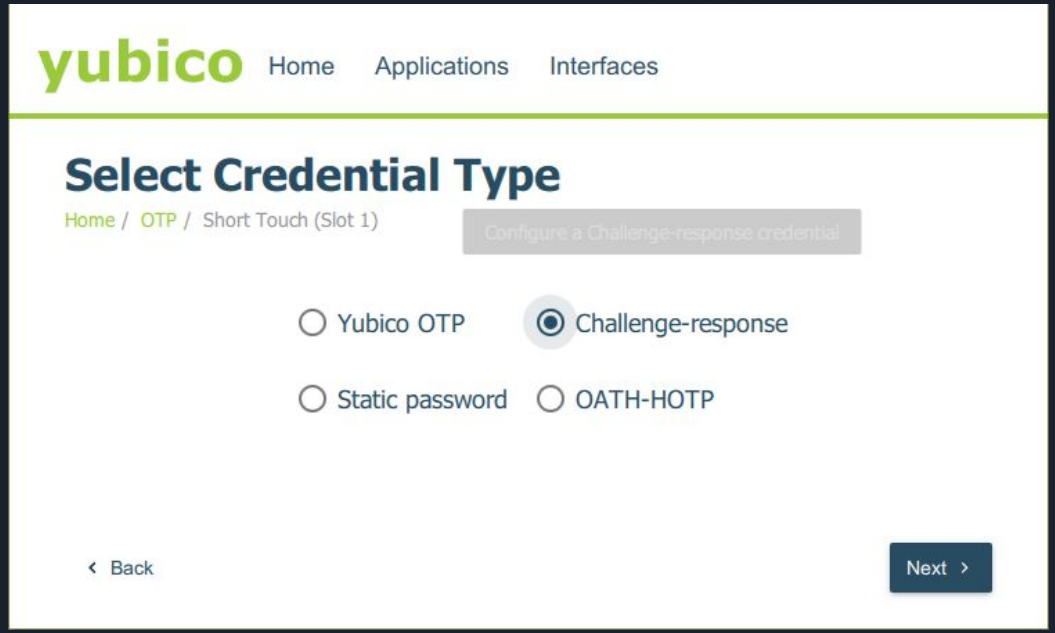
Config slots



OATH-HOTP


yubico
Home
Applications
Interfaces
OATH-HOTP
Home / OTP / Short Touch (Slot 1) / OATH-HOTP
Secret key
Digits
6
Back
Finish

Config slots



challenge-response

HMAC( + challenge)

os authentication

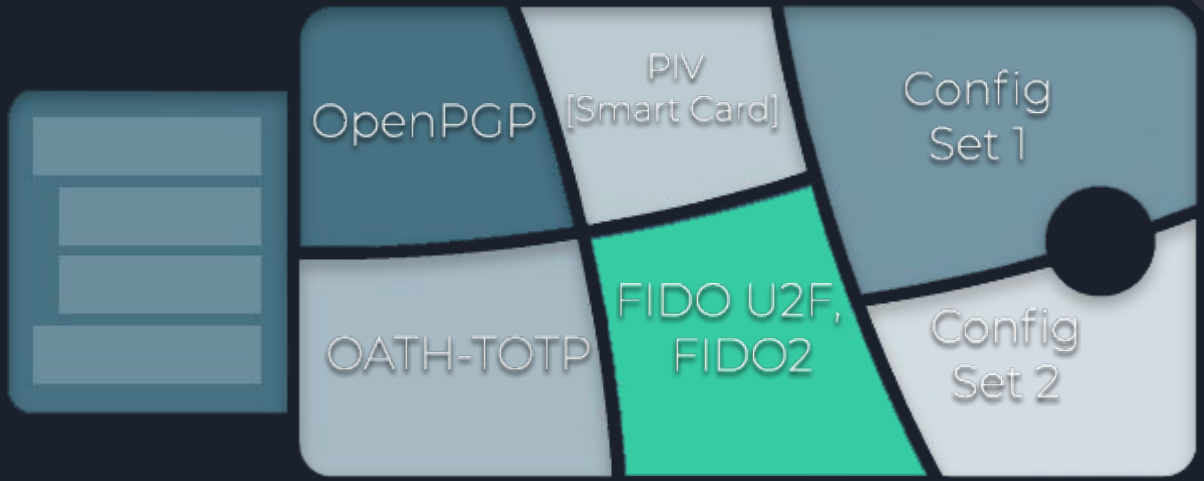
PAM (Pluggable Authentication Modules) - linux

yubikey as second factor (otp / Challenge-response / u2f / passwordless)

yubikey as second in ssh (otp / Challenge-response)

Mac OS (certificate)

FIDO “passwordless”



2FA

2FA - Two-factor authentication

SMS:

1. Нужно иметь номер телефона - я верю что их не станет также как и паролей
2. Not secure

OTP

Smart card

FIDO

Фидонет

ON
FF
ONE
2022

Фидонет

Fido

Fido

Fido - Альянс FIDO (Fast IDentity Online)-консорциум технологических компаний, которые собирались создать новый протокол беспарольной онлайновой аутентификации.

CTAP - Client to Authenticator Protocol - Низкоуровневое описание взаимодействие операционной системы с устройствами аутентификации BLE/NFC/USB

CTAP1 - A formal name of U2F protocol.

U2F - Universal 2nd Factor

CTAP2 - A name for second version of the CTAP protocol

FIDO2 - passwordless authentication protocol

WebAuthn - API Authentication

Fido

Fido

Fido - Альянс FIDO (Fast IDentity Online)-консорциум технологических компаний, которые собирались создать новый протокол беспарольной онлайновой аутентификации.

CTAP - Client to Authenticator Protocol - Низкоуровневое описание взаимодействие операционной системы с устройствами аутентификации BLE/NFC/USB

CTAP1 - A formal name of U2F protocol.

U2F - Universal 2nd Factor

CTAP2 - A name for second version of the CTAP protocol

FIDO2 - passwordless authentication protocol

WebAuthn - API Authentication

Fido

CTAP1 - A formal name of U2F protocol.

U2F - Universal 2nd Factor

2FA (U2F)

CTAP2 - A name for second version of the CTAP protocol

FIDO2 - passwordless authentication protocol

Passwordless

WebAuthn - API Authentication

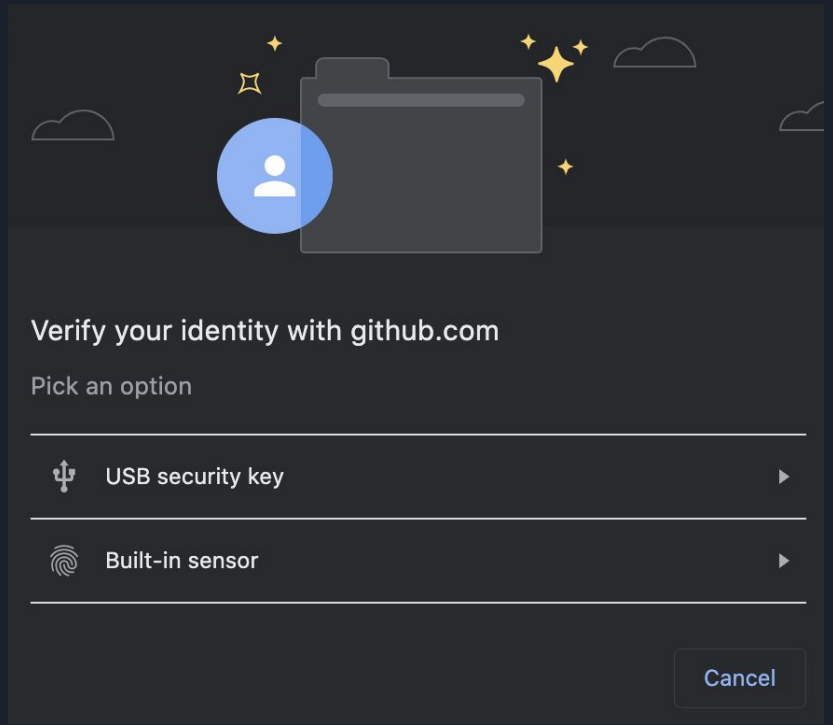
Fido2 Passwordless

Login + Token

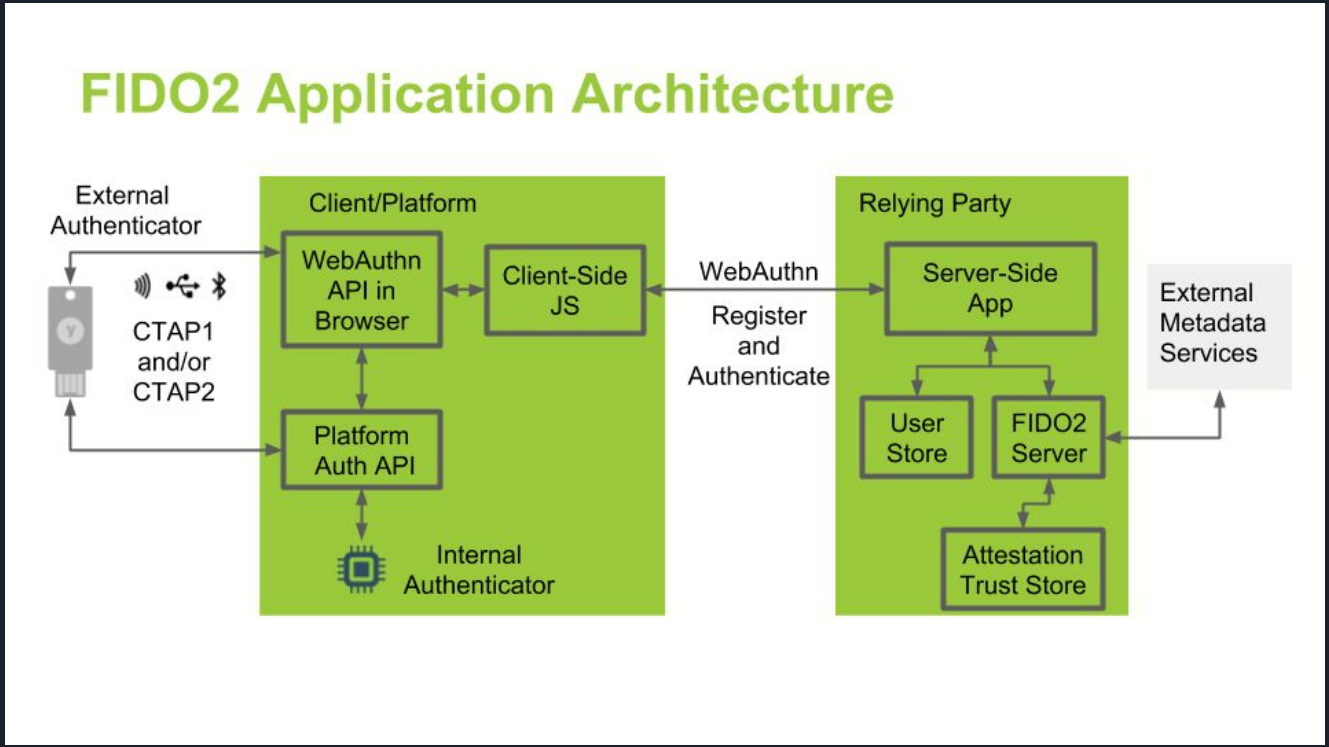
Single f: Username + FIDO2 credential

Second f: Username + password + FIDO2 credential

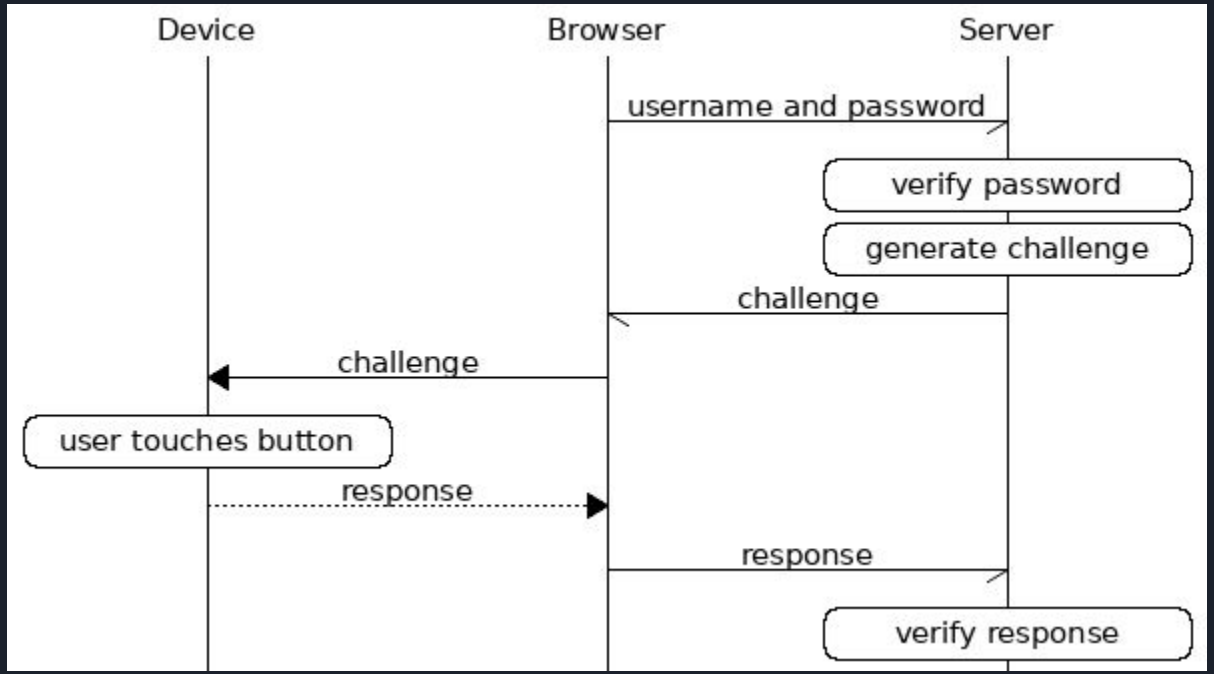
Passwordless MFA: FIDO2 resident key credential + PIN



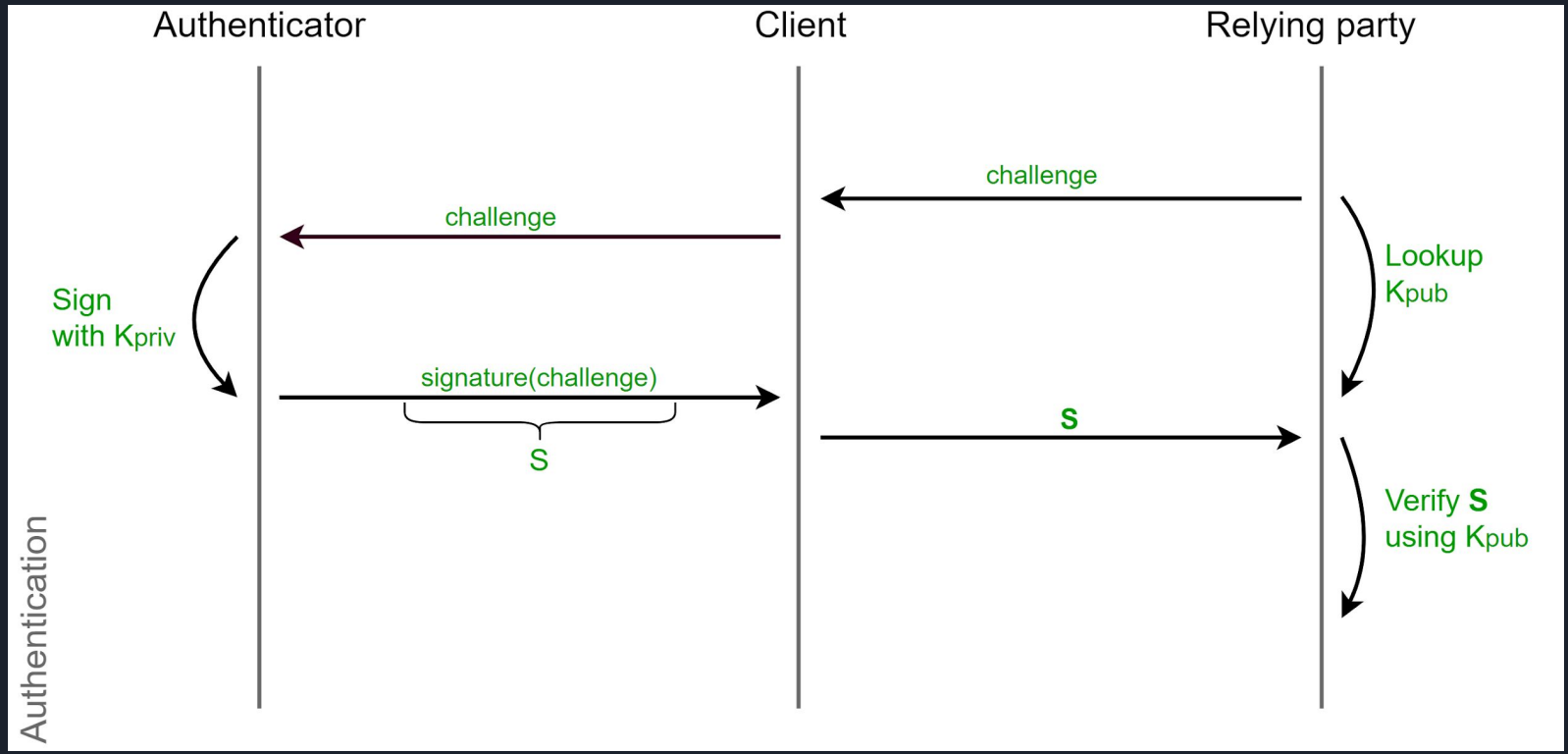
Fido


FIDO2 Application Architecture
External Authenticator
CTAP1 and/or CTAP2
Client/Platform
WebAuthn API in Browser
Client-Side JS
Platform Auth API
Internal Authenticator
WebAuthn
Register and Authenticate
Relying Party
Server-Side App
User Store
FIDO2 Server
Attestation Trust Store
External Metadata Services

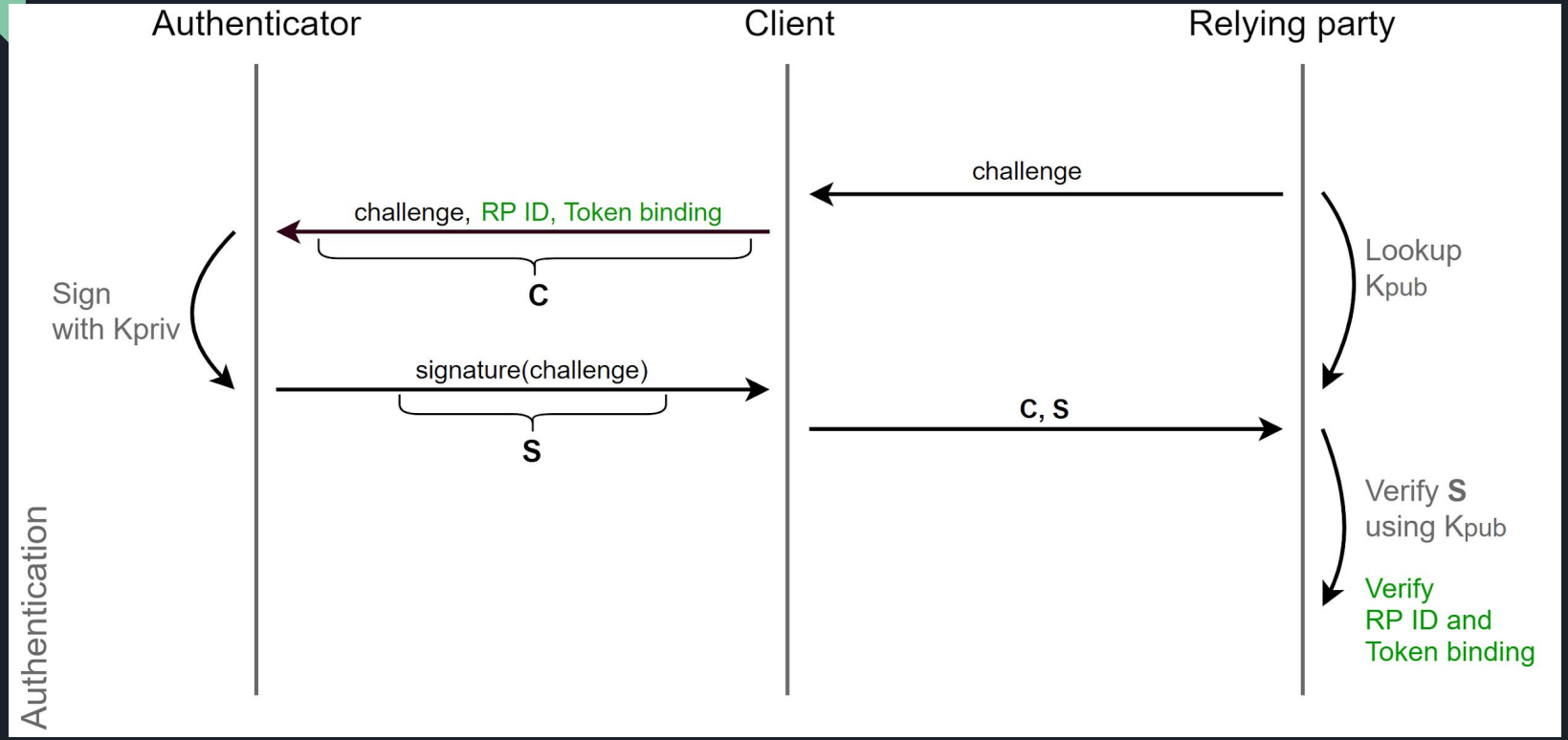
U2F / FIDO2 - 0 User presence test



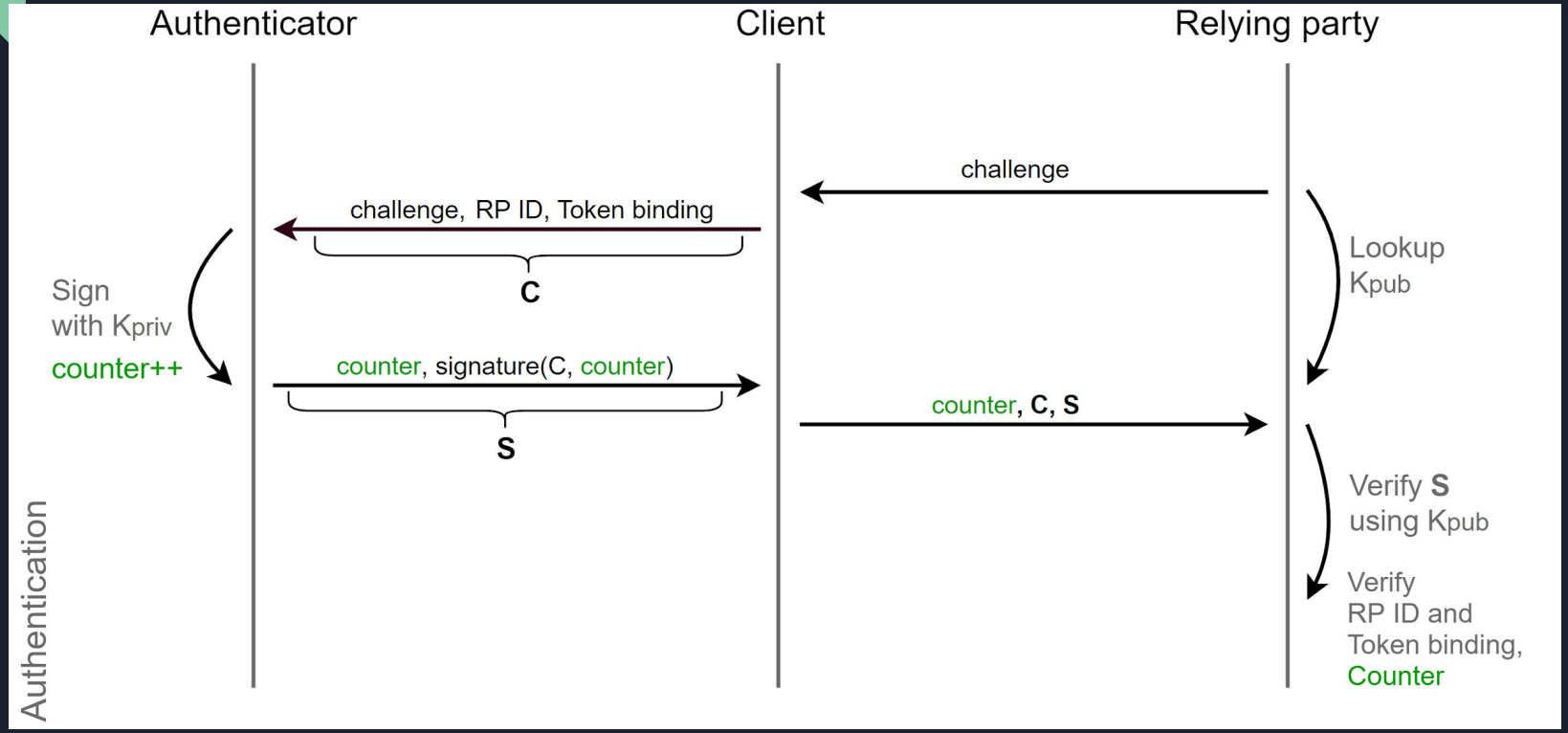
U2F / FIDO2 - 1 Challenge-response



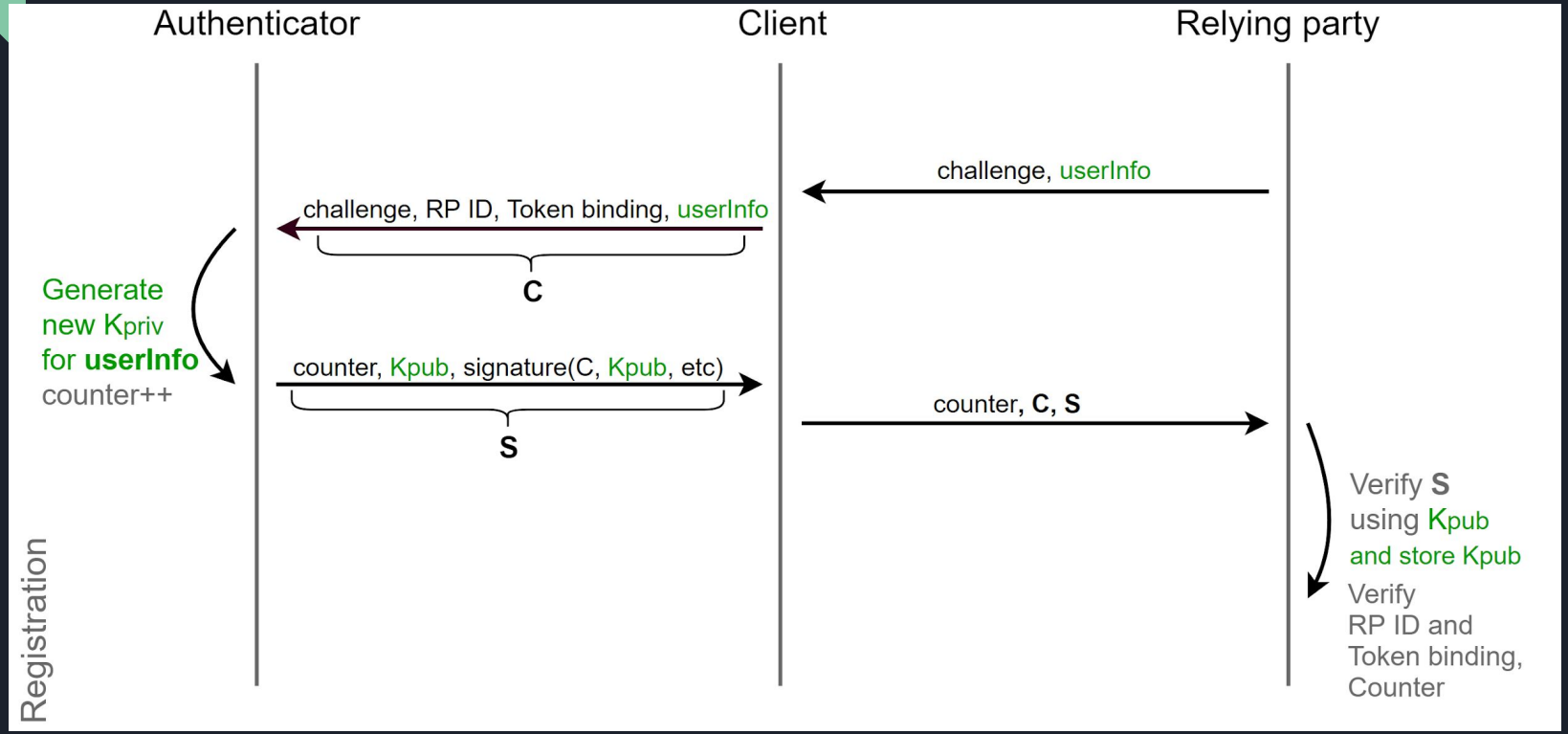
U2F / FIDO2 - 2 Phishing and MitM protection



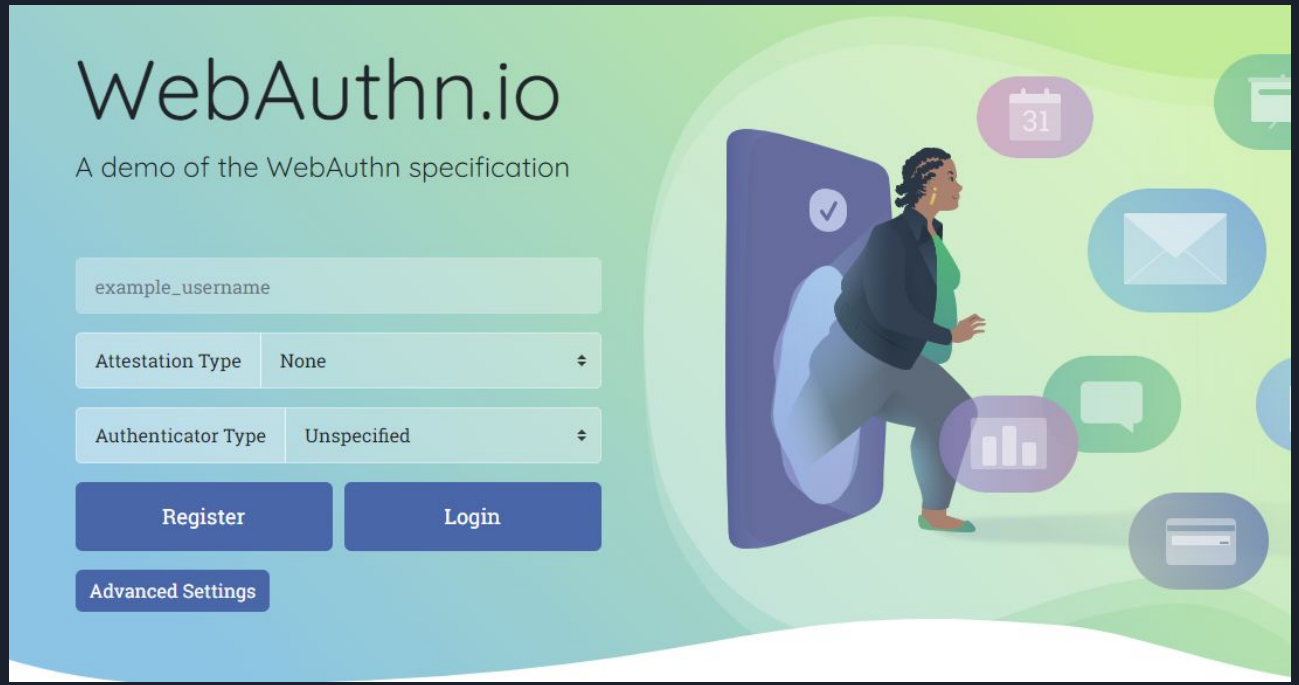
U2F / FIDO2 - 3 Device cloning detection



FIDO2 - 4 Privacy



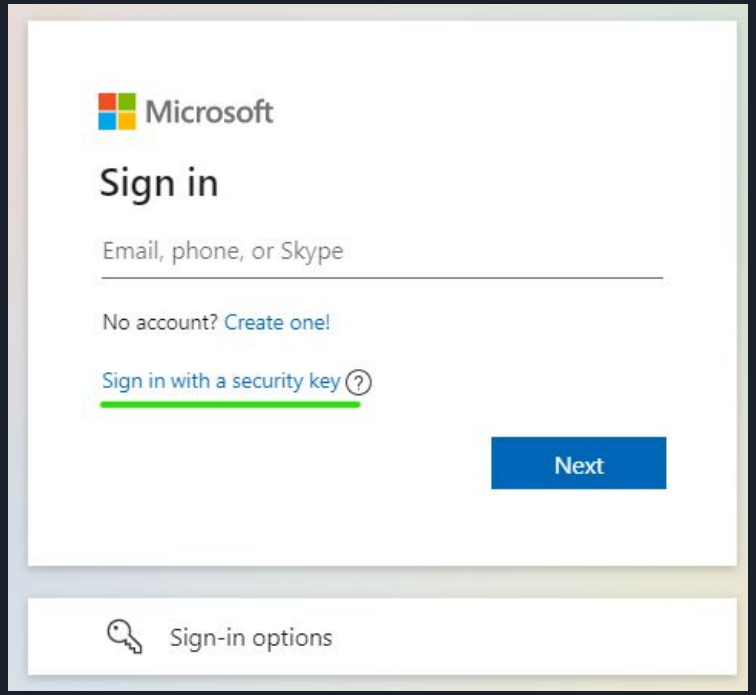
https://webauthn.io/



Windows U2f / passwordless

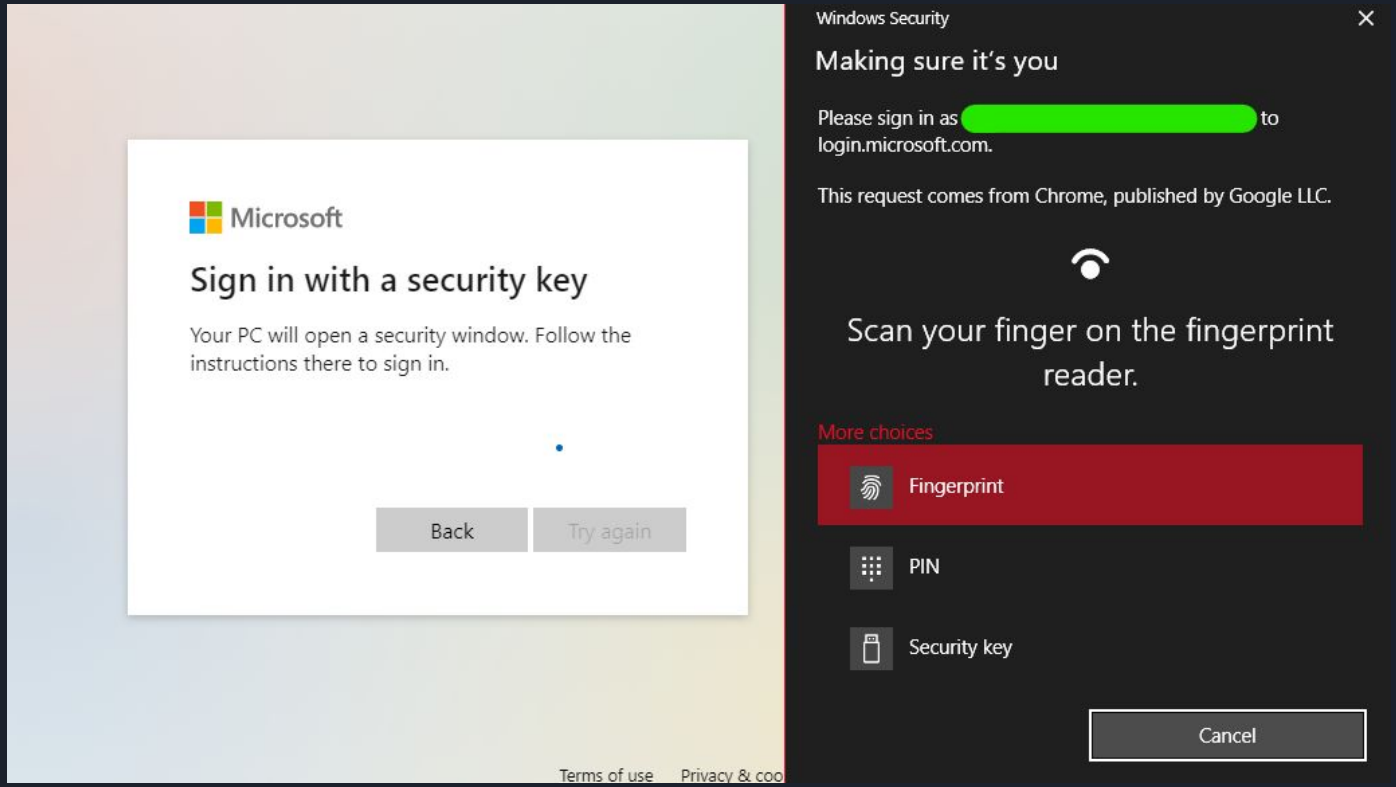
Windows passwordless - demo

https://login.live.com/



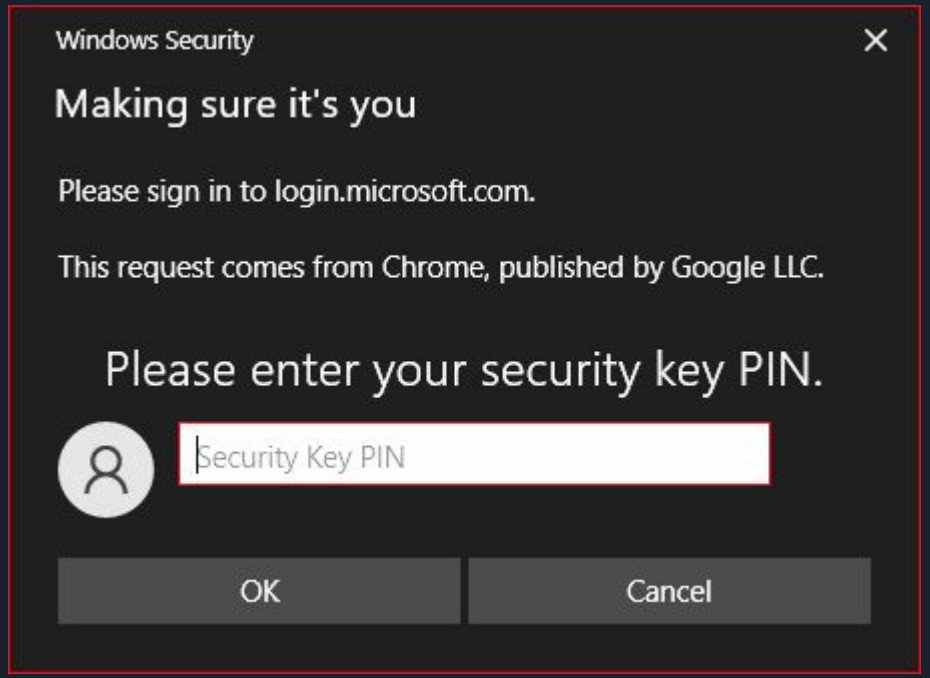
Windows passwordless - demo

https://login.live.com/



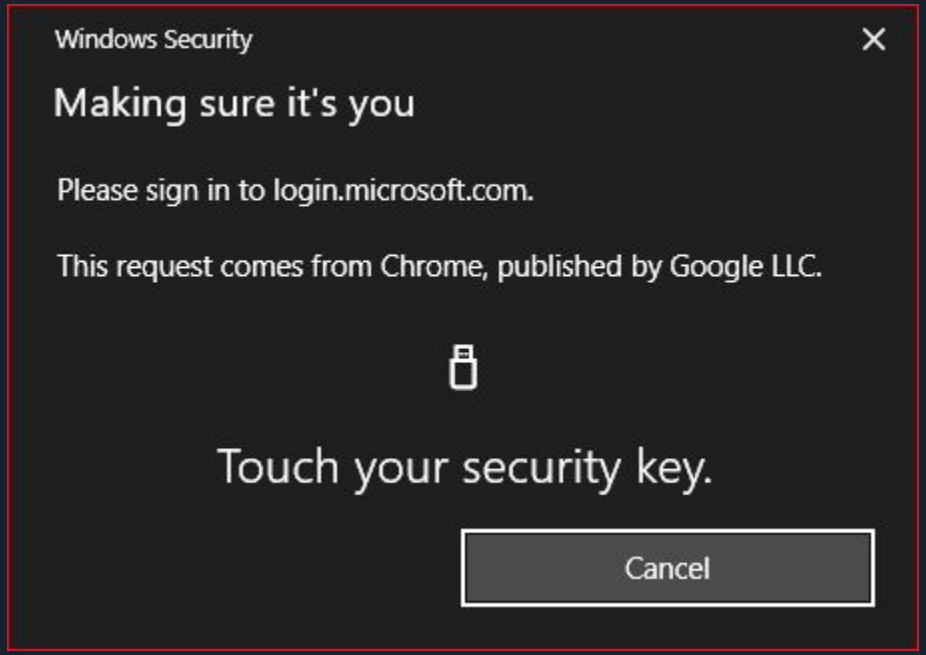
Windows passwordless - demo

https://login.live.com/



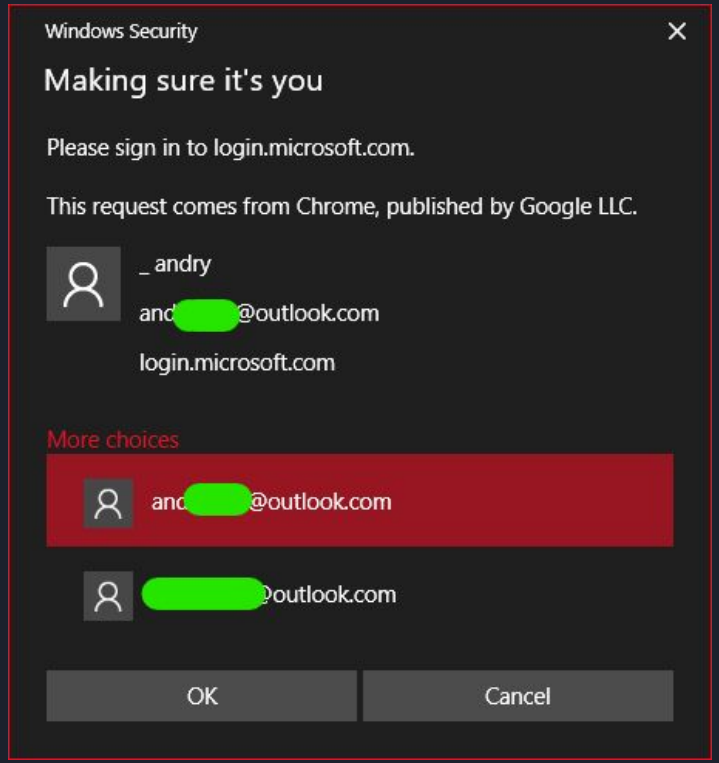
Windows passwordless - demo

https://login.live.com/

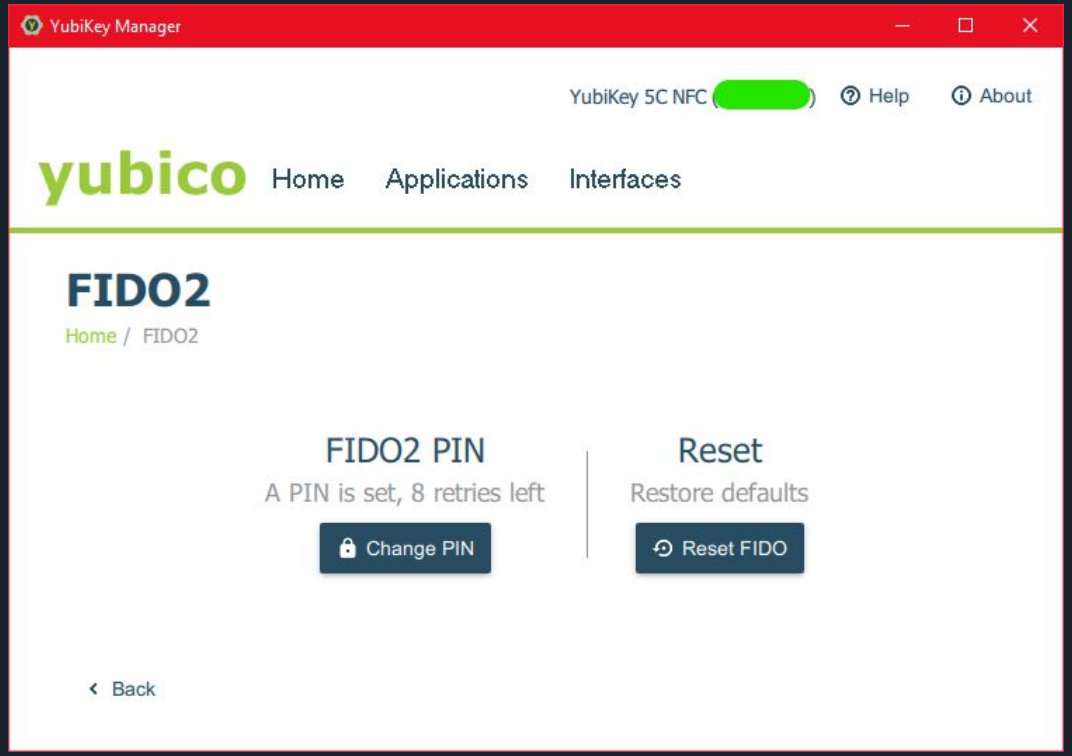


Windows passwordless - demo

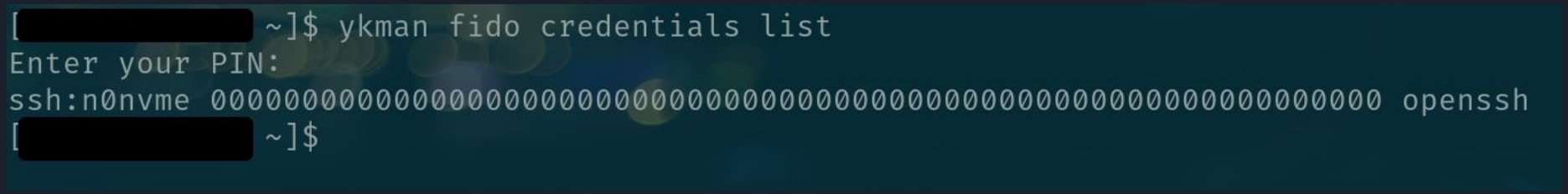
https://login.live.com/



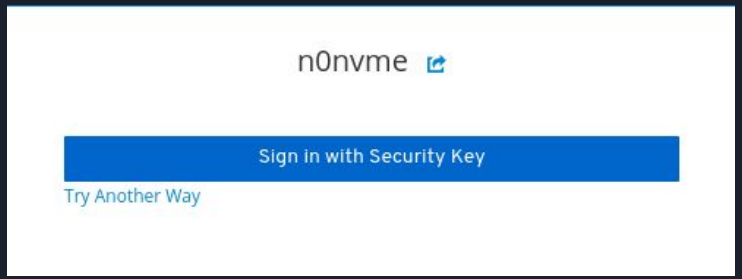
Manage FIDO2 passwordless



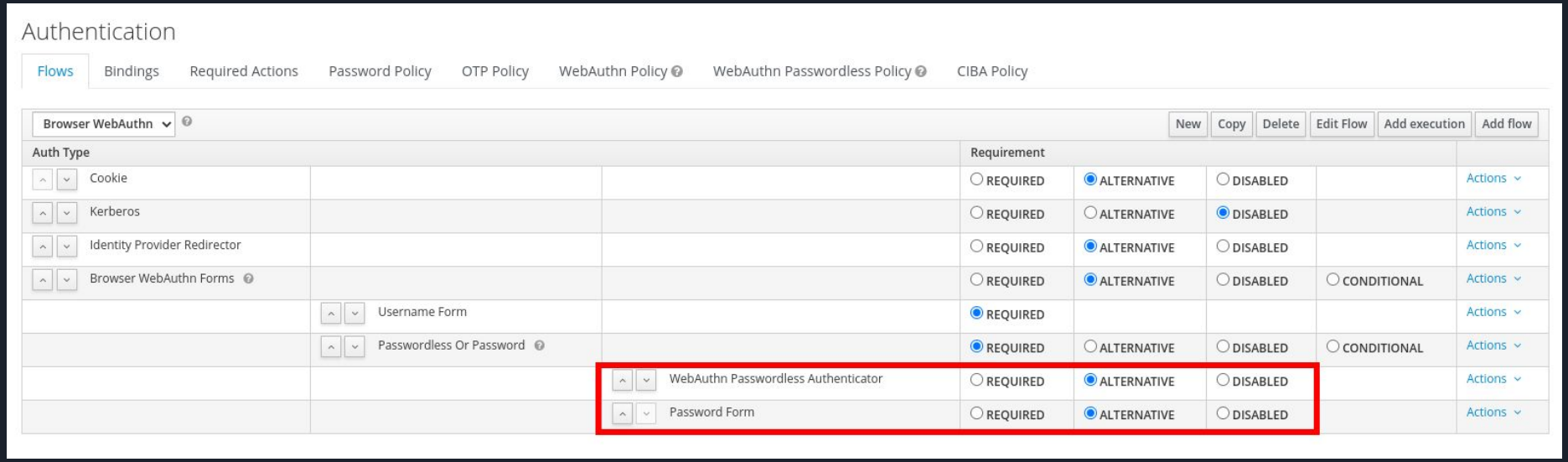
Manage FIDO2 passwordless - YKman



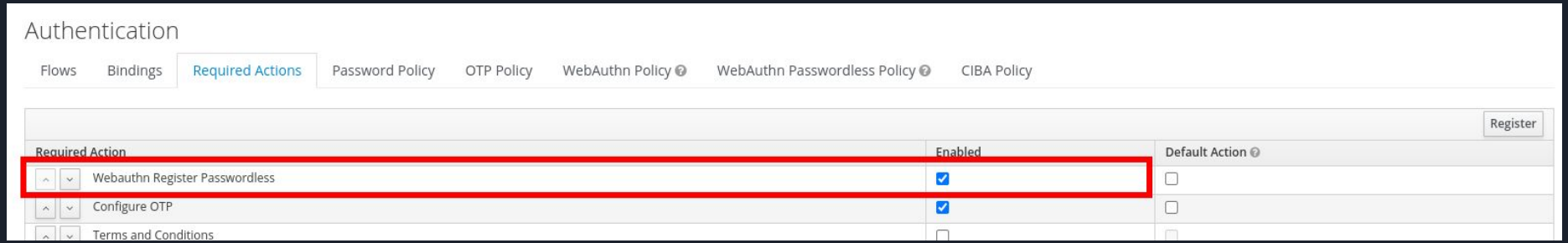
FIDO2 passwordless - SelfHosted - keycloak



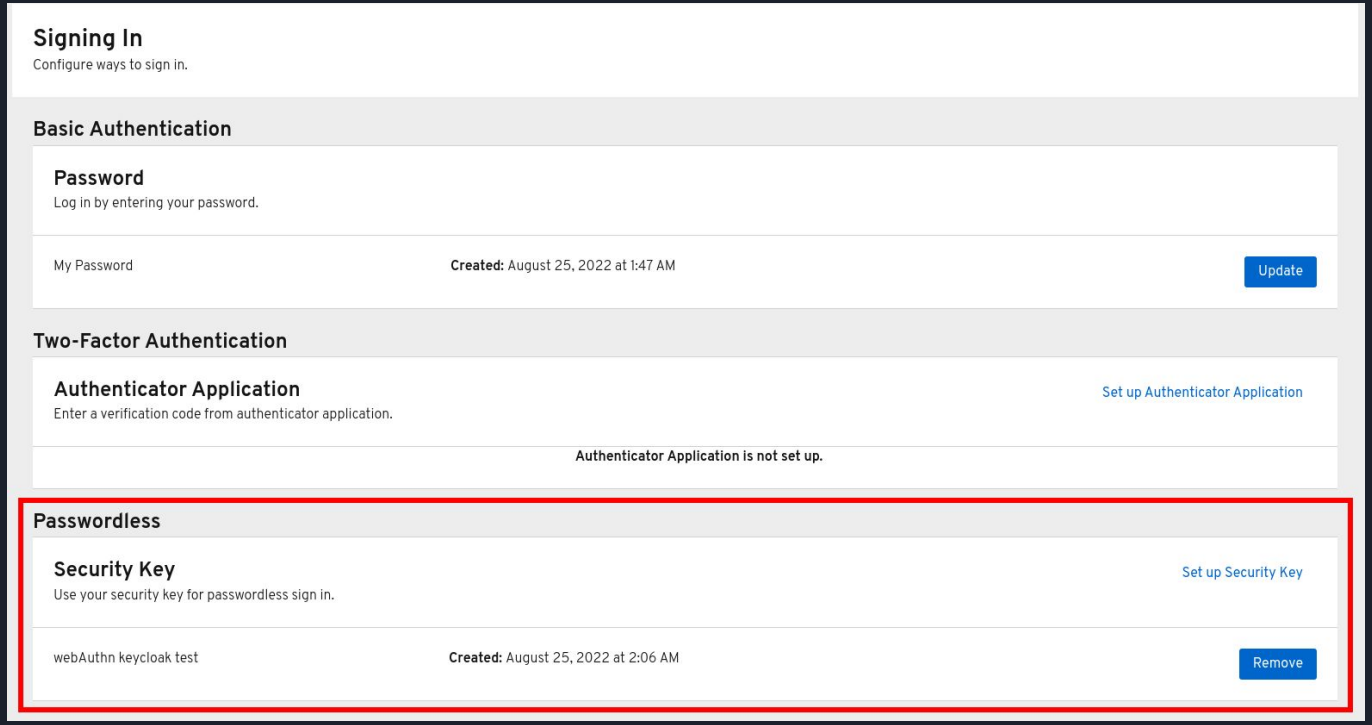
FIDO2 passwordless - SelfHosted - keycloak



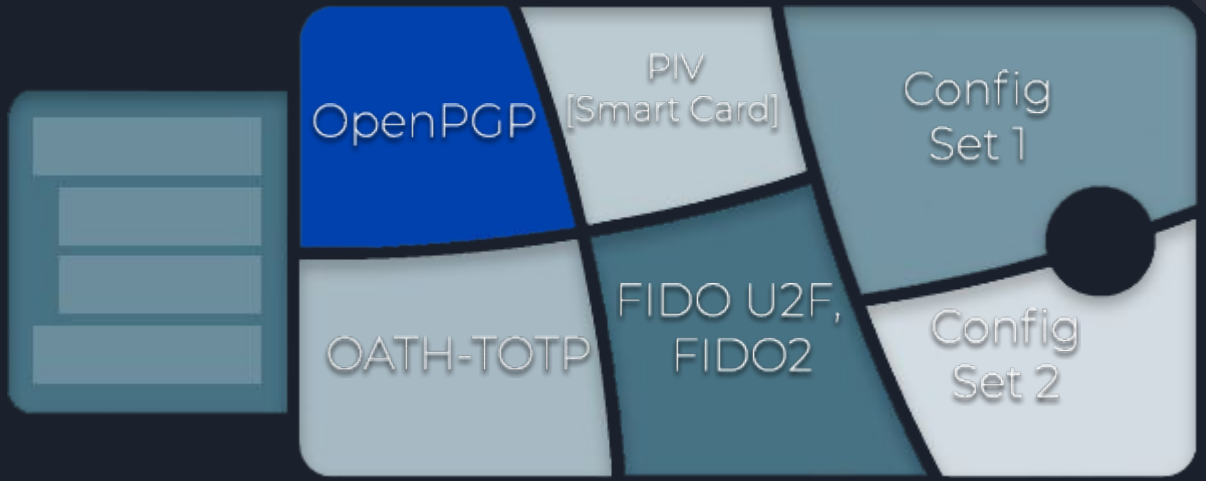
FIDO2 passwordless - SelfHosted - keycloak



FIDO2 passwordless - SelfHosted - keycloak



PGP


OpenPGP
PIV
[Smart Card]
Config
Set 1
OATH-TOTP
FIDO U2F,
FIDO2
Config
Set 2

GPG or PGP or OpenPGP ?!

PGP

GPG

PGP - Pretty Good Privacy

GPG

PGP - Pretty Good Privacy - openpgp - proprietary

GPG

PGP - Pretty Good Privacy - openpgp - proprietary

GPG - GnuPG

PGP - Pretty Good Privacy - openpgp - proprietary

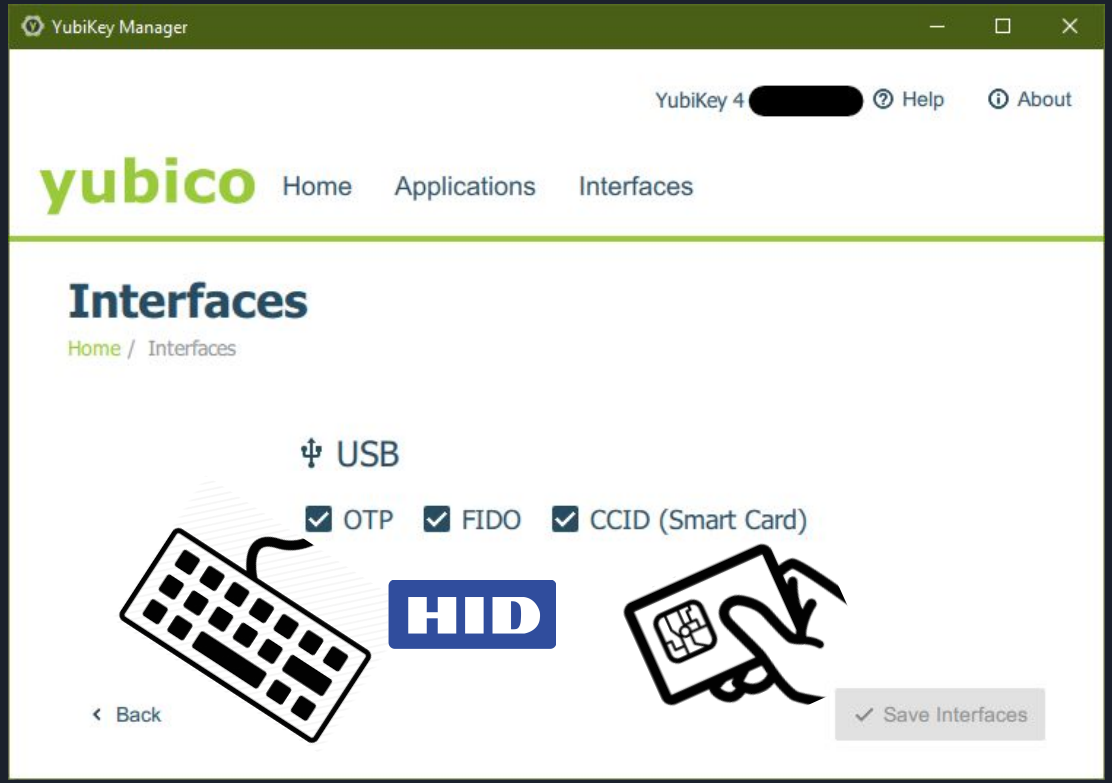
GPG - GnuPG - GNU Privacy Guard

PGP - Pretty Good Privacy - openpgp - proprietary

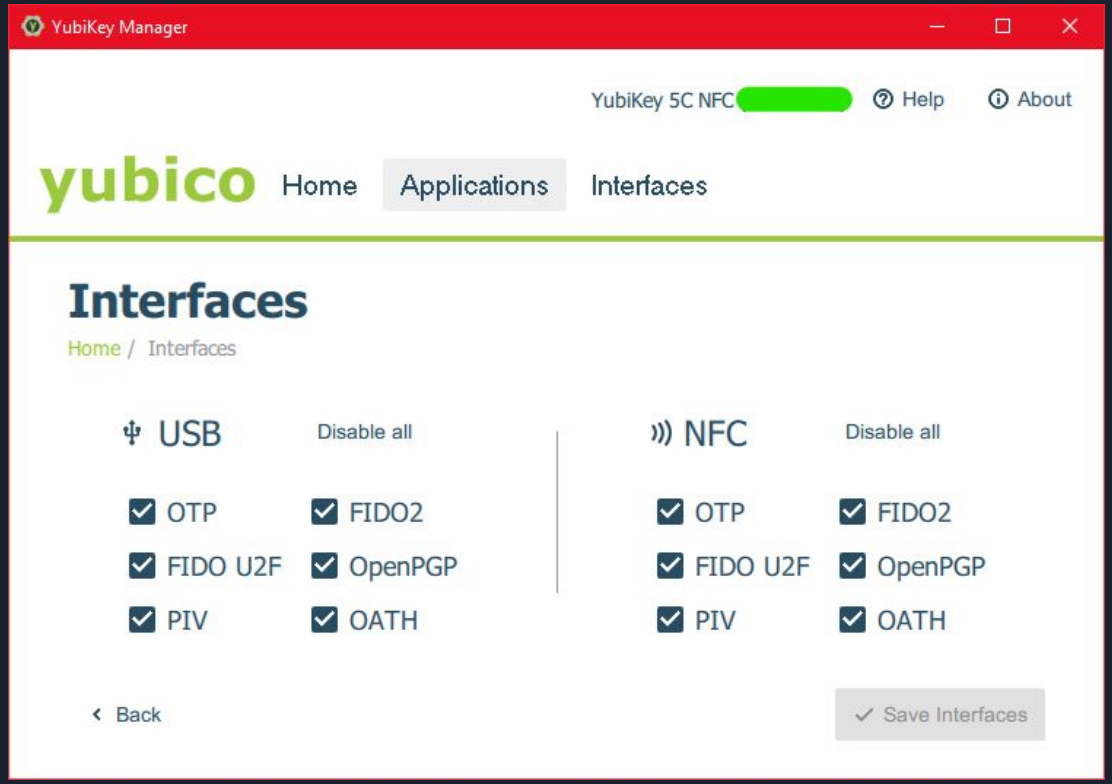
GPG - GnuPG - GNU Privacy Guard - open source

PGP ~ GPG

Что есть yubikey для компа?


YubiKey Manager
YubiKey 4 [redacted]
Help
About
yubico
Home
Applications
Interfaces
Interfaces
Home / Interfaces
USB
OTP
FIDO
CCID (Smart Card)
HID
Back
Save Interfaces

https://docs.yubico.com/hardware/yubikey/yk-5/tech-manual/index.html



Yubikey/SmartCard backed TLS servers

Yubikey/SmartCard(GPG) backed TLS servers

Yubikey/SmartCard(GPG) backed TLS servers

GPG и SSL(TLS)

в основе лежит RSA/ECC (Elliptic Curve Cryptography)

Та почему не использовать?)

Yubikey/Smartcard backed TLS servers

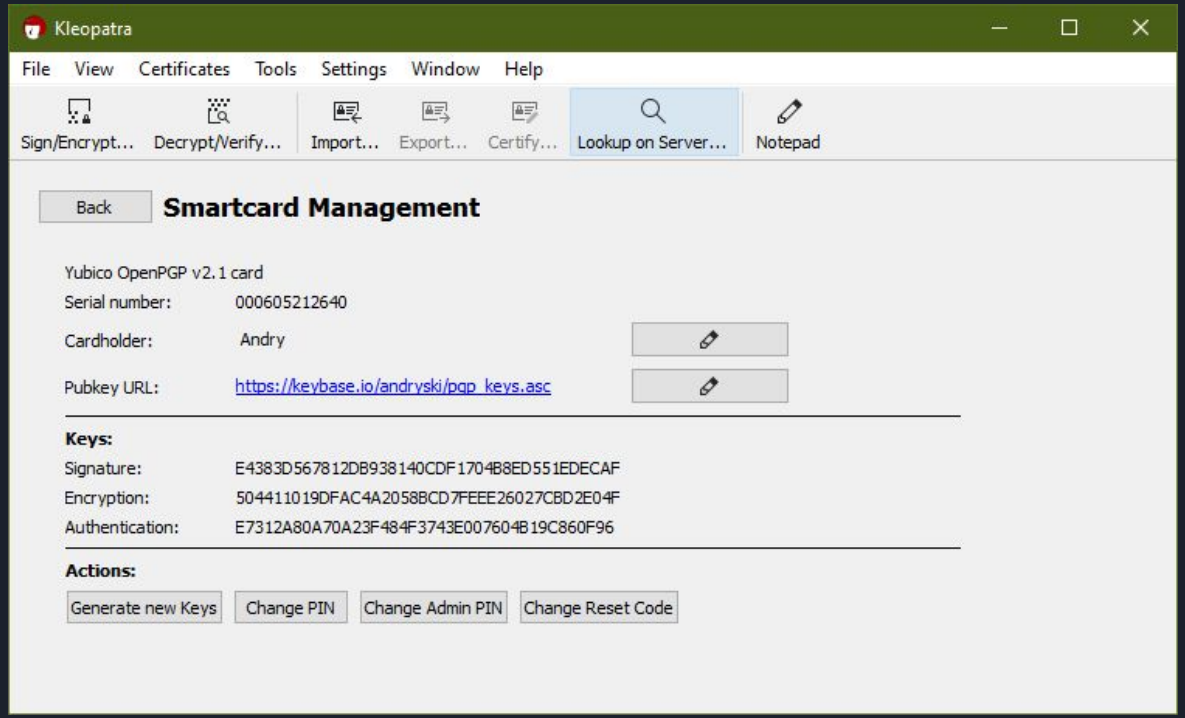
YubiKey & ssh (?)

YubiKey & ssh(certificate)

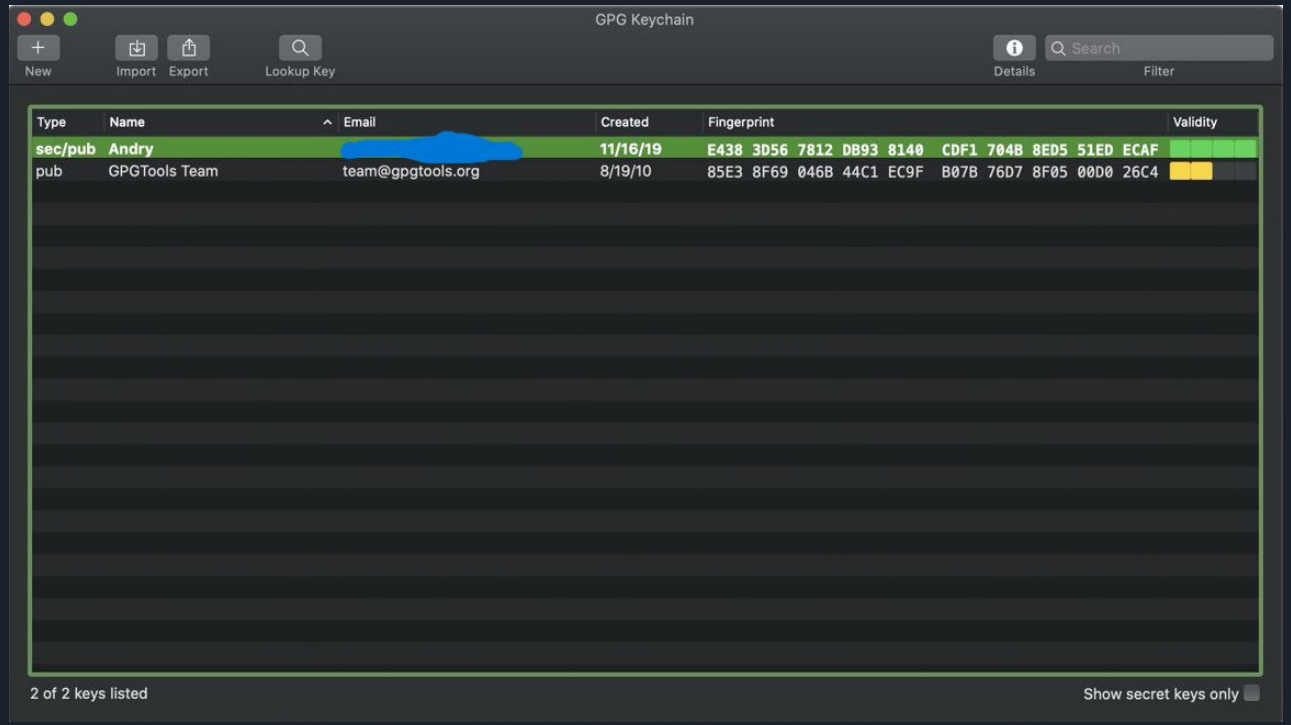
home work

YubiKey & ssh (gpg)

Common

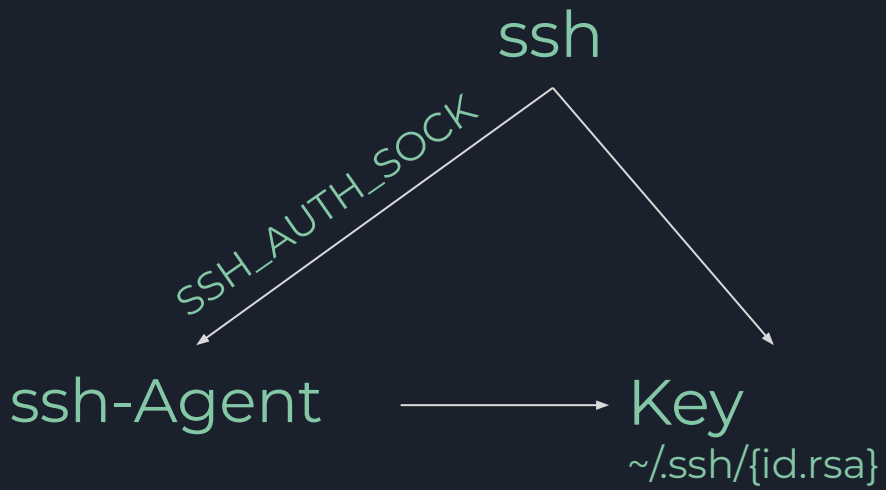

Kleopatra
File View Certificates Tools Settings Window Help
Sign/Encrypt... Decrypt/Verify... Import... Export... Certify... Lookup on Server... Notepad
Back
Smartcard Management
Yubico OpenPGP v2.1 card
Serial number: 000605212640
Cardholder: Andry
Pubkey URL: https://keybase.io/andryski/pgp_keys.asc
Keys:
Signature: E4383D567812DB938140CDF1704B8ED551EDECAF
Encryption: 504411019DFAC4A2058BCD7FEEE26027CBD2E04F
Authentication: E7312A80A70A23F484F3743E007604B19C860F96
Actions:
Generate new Keys
Change PIN
Change Admin PIN
Change Reset Code

Common

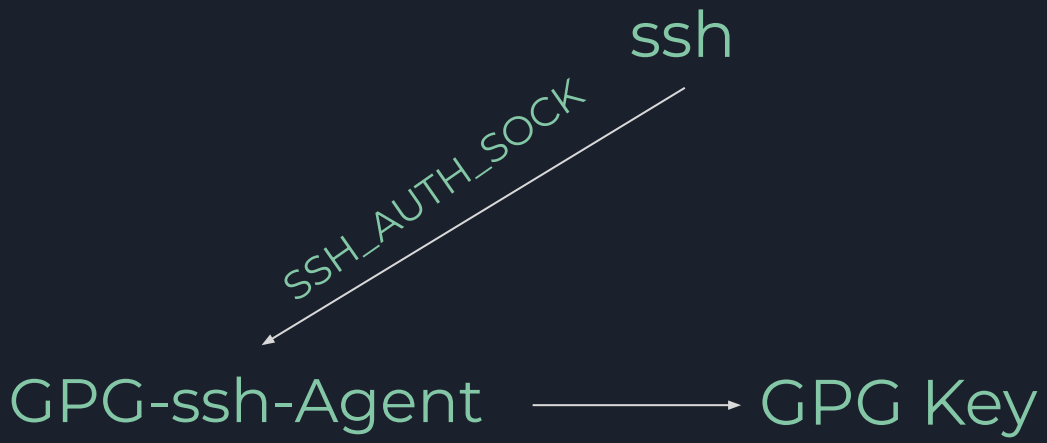

GPG Keychain
New
Import
Export
Lookup Key
Details
Search
Filter
Type
Name
Email
Created
Fingerprint
Validity
sec/pub
Andry
11/16/19
E438 3D56 7812 DB93 8140 CDF1 704B 8ED5 51ED ECAF
pub
GPGTools Team
team@gpgtools.org
8/19/10
85E3 8F69 046B 44C1 EC9F B07B 76D7 8F05 00D0 26C4
2 of 2 keys listed
Show secret keys only

ssh

environment variable - SSH_AUTH_SOCK



ssh



ssh

```
export SSH_AUTH_SOCK="${HOME}/.gnupg/S.gpg-agent.ssh"
```

```
$ ssh-add -L
```

```
ssh-rsa AAAA.....DmL cardno:000605212640
```

```
echo <KEY> > ./.ssh/authorized_keys
```

```
ssh-copy-id
```

ssh + GPG in Linux/Mac

~/.bashrc
~/.zshrc

```
ssg(){ ~/scripts/gssh.sh $@; }
export -f ssg
```

ssh + GPG in Linux/Mac

```
#!/bin/bash
if [ ! -S "${HOME}/.gnupg/S.gpg-agent.ssh" ]; then
    gpg-agent --daemon > /dev/null
fi

gpg --card-status > /dev/null 2>/dev/null
if [ $? -eq 0 ]
then
  echo "Success: gpg token fund"
else
  echo "Failure: gpg token NOT fund" >&2

  exit 1
fi
```

ssh + GPG in Linux/Mac

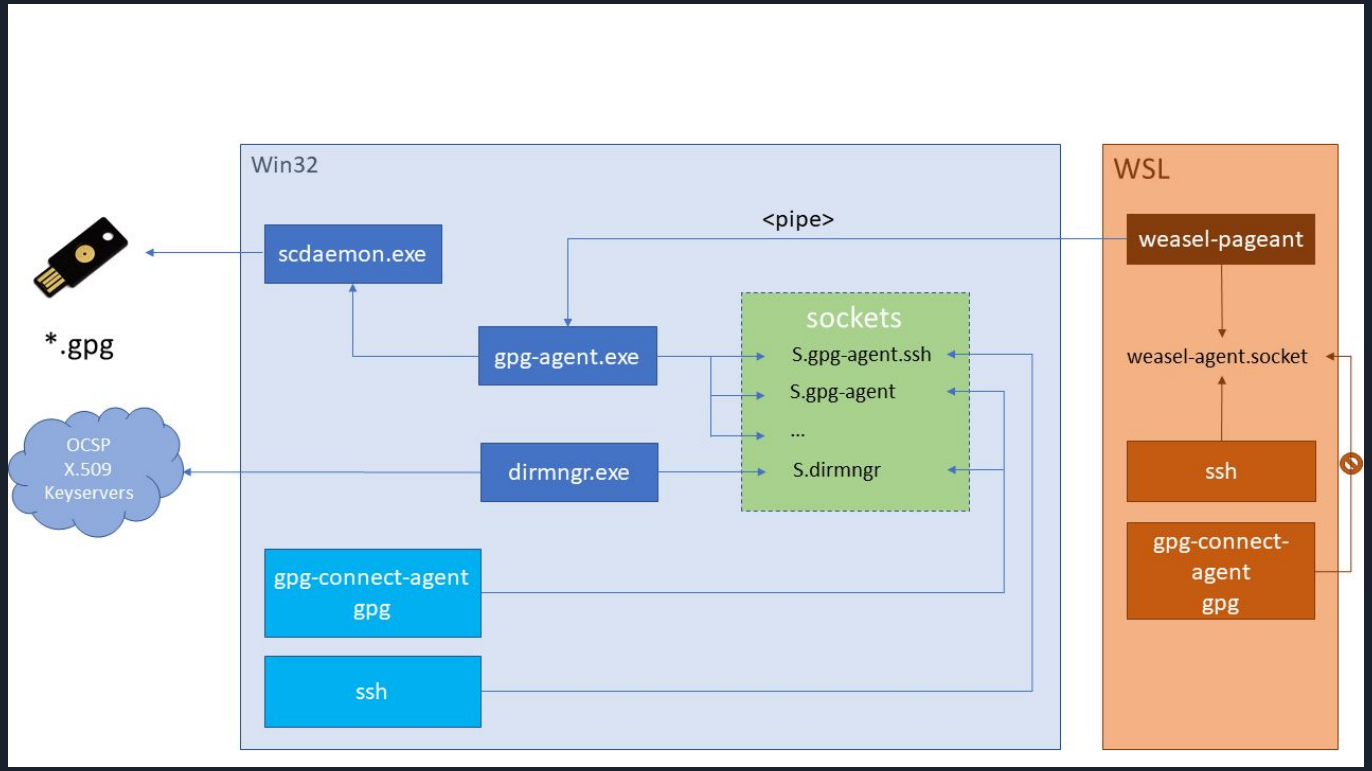
```
export SSH_AUTH_SOCK_old=$SSH_AUTH_SOCK
export SSH_AUTH_SOCK="${HOME}/.gnupg/S.gpg-agent.ssh"
ssh $@
export SSH_AUTH_SOCK=$SSH_AUTH_SOCK_old
unset SSH_AUTH_SOCK_old
```

ssh + GPG in Mac

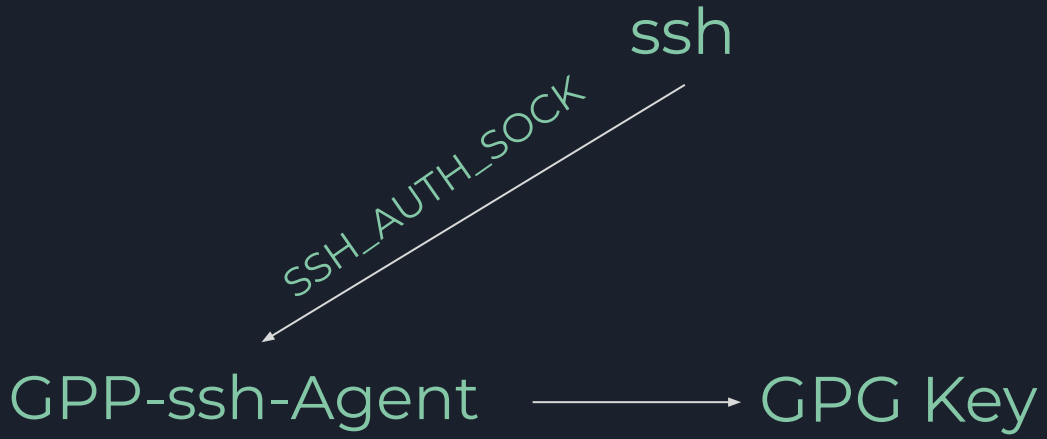
```
#!/bin/bash
if [ ! -S "${HOME}/.gnupg/S.gpg-agent.ssh" ]; then
    gpg-agent --daemon > /dev/null
fi

gpg --card-status > /dev/null 2>/dev/null
if [ $? -eq 0 ]
then
  echo "Success: gpg token fund"
else
  echo "Failure: gpg token NOT fund" >&2
  gpgconf --kill gpg-agent                    export GPG_TTY=$(tty)
  exit 1
fi
```

https://www.gnupg.org/(it)/documentation/manuals/gnupg/Common-Problems.html

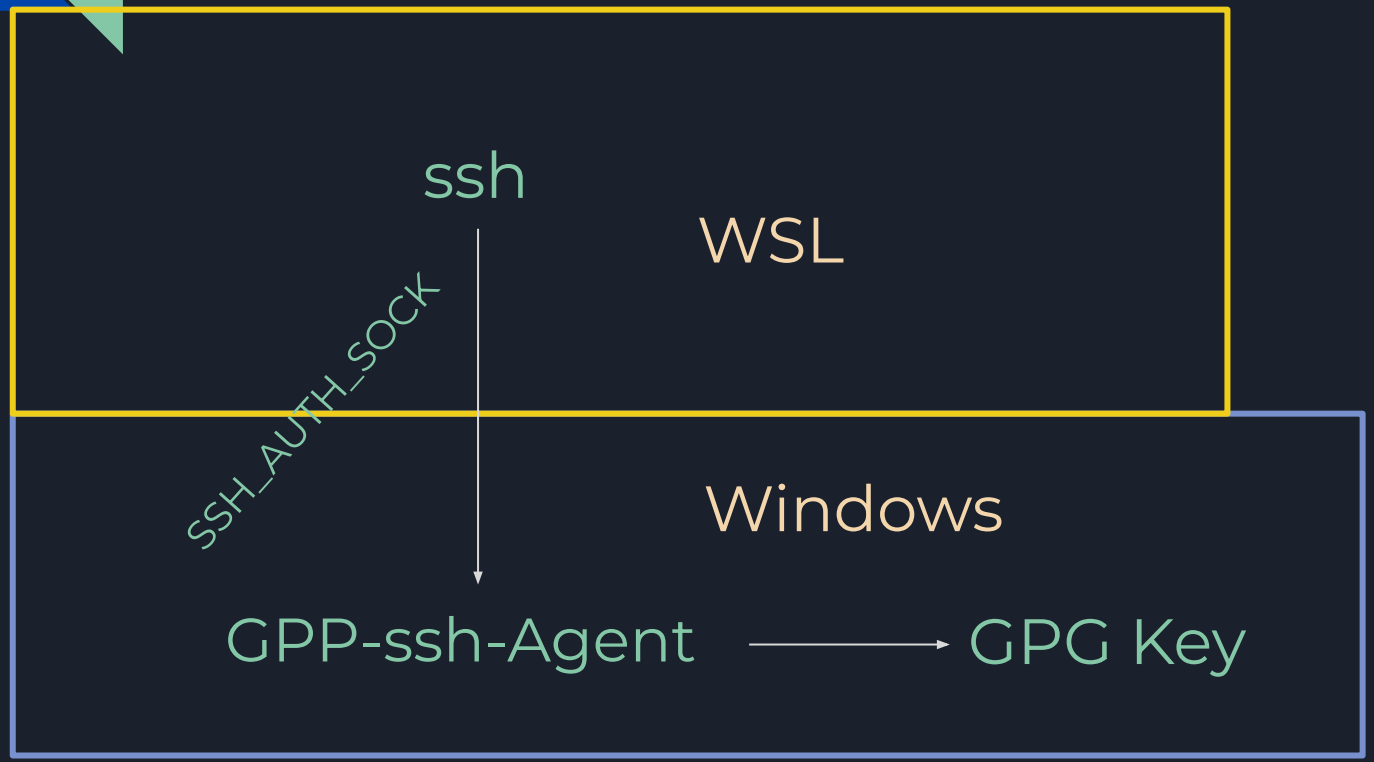
ssh + GPG in Windows (WSL SSH bridge)



ssh + GPG in Windows (WSL SSH bridge)



ssh + GPG in Windows (WSL SSH bridge)



ssh + GPG in Windows (WSL SSH bridge)

%APPDATA%\gnupg\gpg-agent.conf -> enable-putty-support

shell:startup

"C:\Program Files (x86)\GNU\GnuPG\gpg-connect-agent.exe" /bye

WSL

https://github.com/benpye/wsl-ssh-pageant/releases

ssh + GPG in Windows (WSL SSH bridge)

C:\wsl-ssh\ssg.sh

```
#!/bin/bash
export SSH_AUTH_SOCK_old=$SSH_AUTH_SOCK
eval $(/mnt/c/wsl-ssh/weasel-pageant -r -a /tmp/S.weasel-pageant)
ssh $@
export SSH_AUTH_SOCK=$SSH_AUTH_SOCK_old
unset SSH_AUTH_SOCK_old
```

ssh + GPG in Windows (WSL SSH bridge)

```
export SSH_AUTH_SOCK_old=$SSH_AUTH_SOCK
export SSH_AUTH_SOCK="${HOME}/.gnupg/S.gpg-agent.ssh"

wsl
export SSH_AUTH_SOCK_old=$SSH_AUTH_SOCK
eval $(/mnt/c/wsl-ssh/weasel-pageant -r -a /tmp/S.weasel-pageant)
```

ssh + GPG in Windows (SSH)

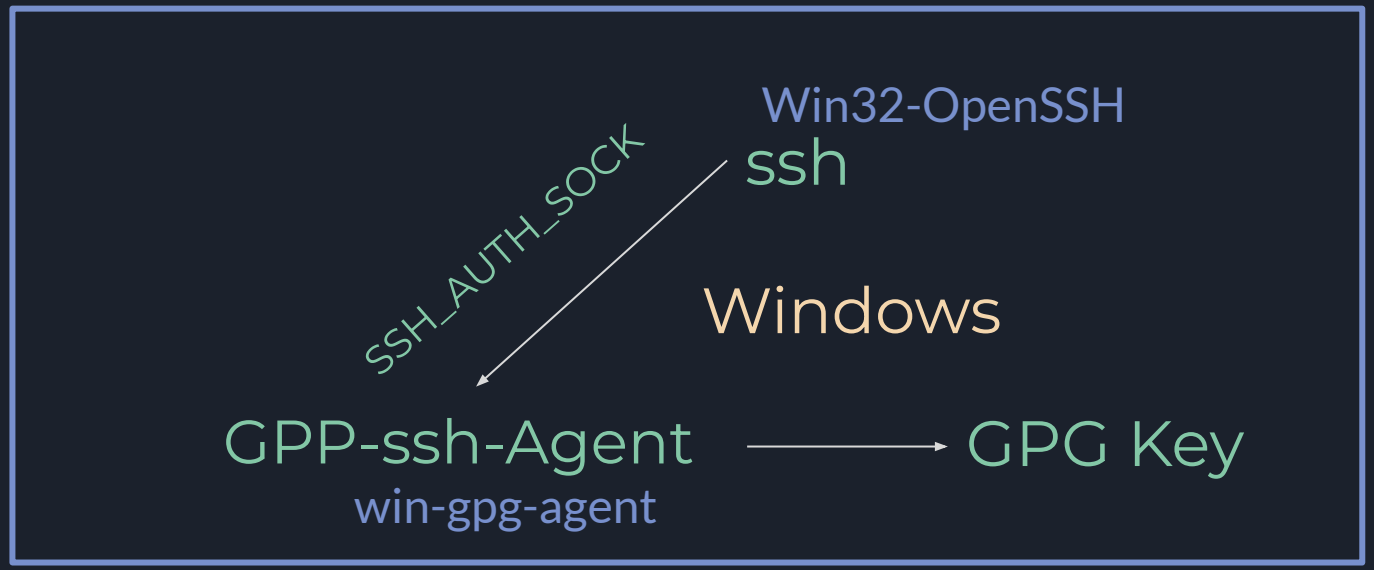
win-gpg-agent

ssh

https://github.com/rupor-github/win-gpg-agent

https://github.com/PowerShell/Win32-OpenSSH

ssh + GPG in Windows (SSH)



ssh Deeper

ssh Deeper

Релиз OpenSSH 8.2 с поддержкой токенов двухфакторной аутентификации FIDO/U2F

https://www.openssh.com/txt/release-8.2
https://www.opennet.ru/opennews/art.shtml?num=52369
https://about.gitlab.com/blog/2022/03/03/how-to-protect-gitlab-connected-ssh-key-with-yubikey/
https://cryptsus.com/blog/how-to-configure-openssh-with-yubikey-security-keys-u2f-otp-authentication-ed25519-sk-ecdsa-sk-on-ubuntu-18.04.html

FIDO2 SSH

FIDO 2
ssh-keygen -t ed25519-sk -O resident -O application=ssh:n0nvme -C n0nvme@yubikey2

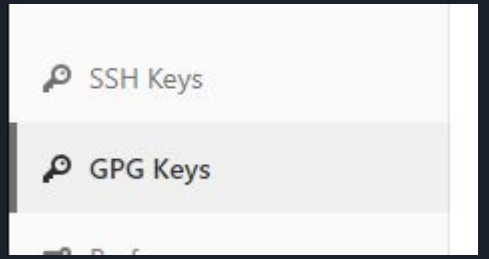
ssh-add -K - add resident key to agent
ssh-keygen -K - restore resident key to File system

windows bug - https://github.com/PowerShell/Win32-OpenSSH/issues/1915

U2F
нужно выполнять без "-O resident" , и сгенеренные файлы не потерять.

GIT

GIT



GIT

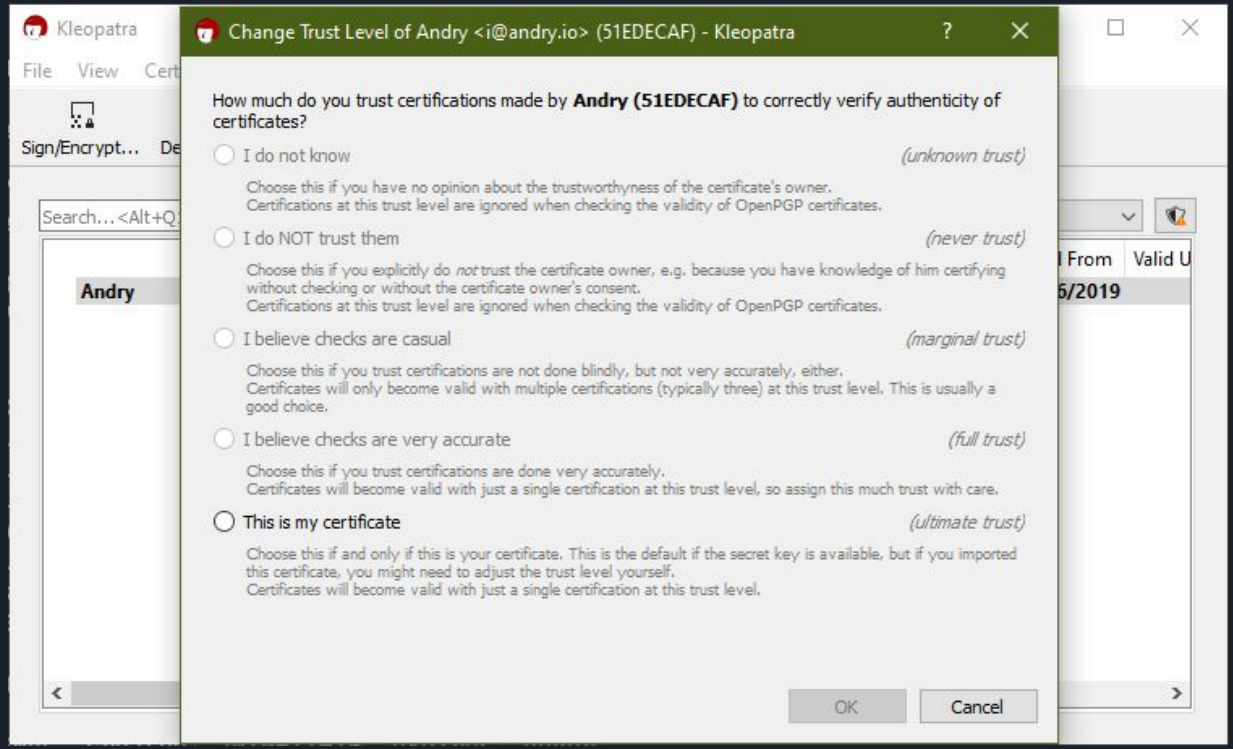
(Signature key)

```
git config --global user.signingkey 51EDECAF

git config --global commit.gpgsign true
```

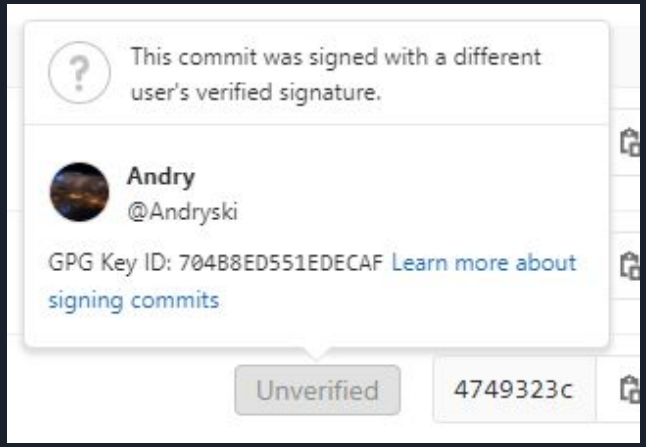
windows

```
git config --global gpg.program "C:\Program Files (x86)\GnuPG\bin\gpg.exe"
```

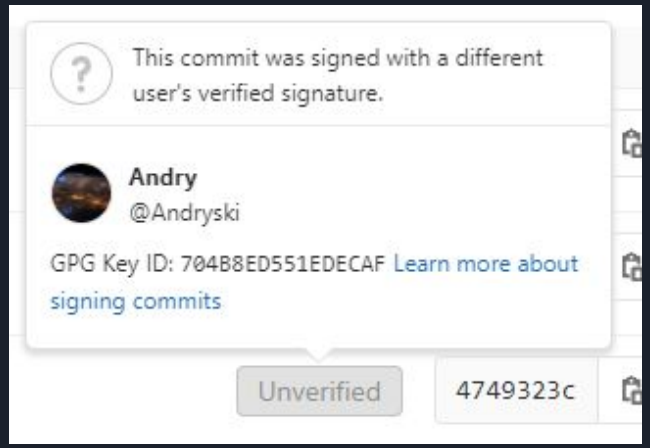
GIT


Kleopatra
File View Cert
Sign/Encrypt... De
Search... <Alt+Q>
Andry
From
Valid U
5/2019
Change Trust Level of Andry <i@andry.io> (51EDECAF) - Kleopatra
How much do you trust certifications made by Andry (51EDECAF) to correctly verify authenticity of certificates?
I do not know (unknown trust)
Choose this if you have no opinion about the trustworthyness of the certificate's owner.
Certifications at this trust level are ignored when checking the validity of OpenPGP certificates.
I do NOT trust them (never trust)
Choose this if you explicitly do not trust the certificate owner, e.g. because you have knowledge of him certifying without checking or without the certificate owner's consent.
Certifications at this trust level are ignored when checking the validity of OpenPGP certificates.
I believe checks are casual (marginal trust)
Choose this if you trust certifications are not done blindly, but not very accurately, either.
Certificates will only become valid with multiple certifications (typically three) at this trust level. This is usually a good choice.
I believe checks are very accurate (full trust)
Choose this if you trust certifications are done very accurately.
Certificates will become valid with just a single certification at this trust level, so assign this much trust with care.
This is my certificate (ultimate trust)
Choose this if and only if this is your certificate. This is the default if the secret key is available, but if you imported this certificate, you might need to adjust the trust level yourself.
Certificates will become valid with just a single certification at this trust level.
OK
Cancel

GIT

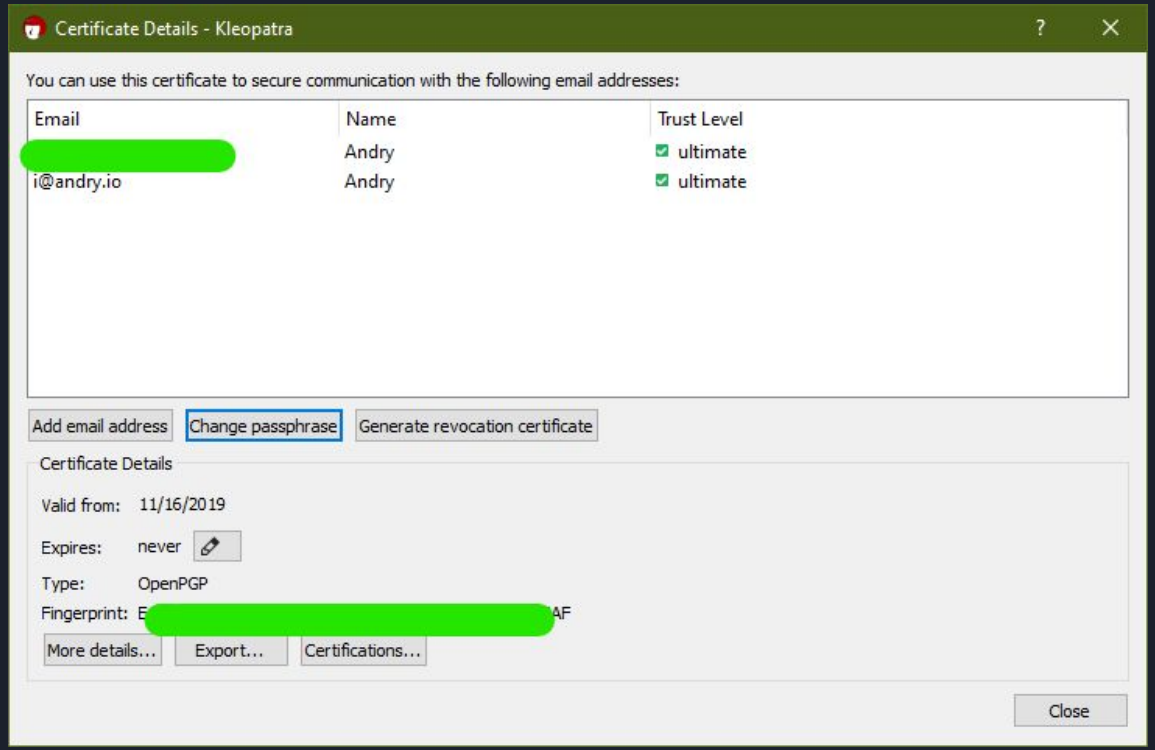

This commit was signed with a different user's verified signature.
Andry
@Andryski
GPG Key ID: 704B8ED551EDECAF Learn more about signing commits
Unverified
4749323c

GIT

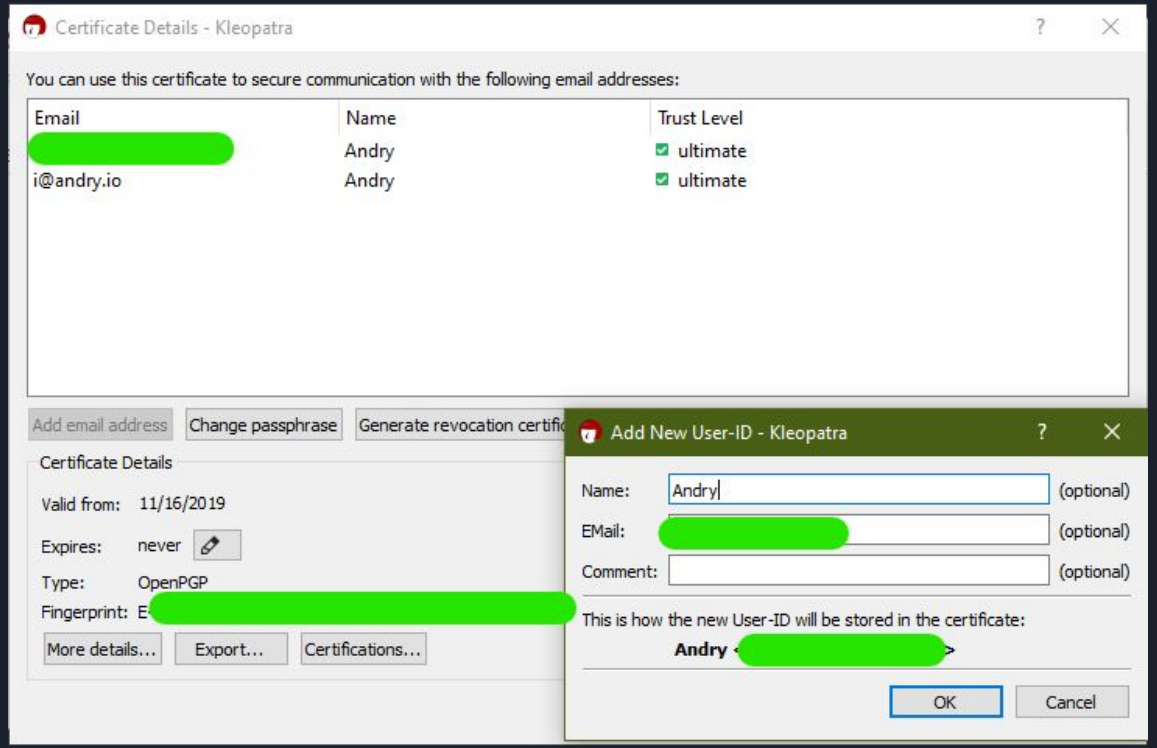



```
      created ....: 2019-11-16 14:41:33
Authentication key: E731 2A80 A70A 23F4 84F3  743E 0076 04B1 9C86 0F96
General key info..: pub  rsa2048/704B8ED551EDECAF 2019-11-16 Andry <i@andry.io>
sec>  rsa2048/704B8ED551EDECAF  created: 2019-11-16  expires: never
                                card-no: 0006 05212640
```

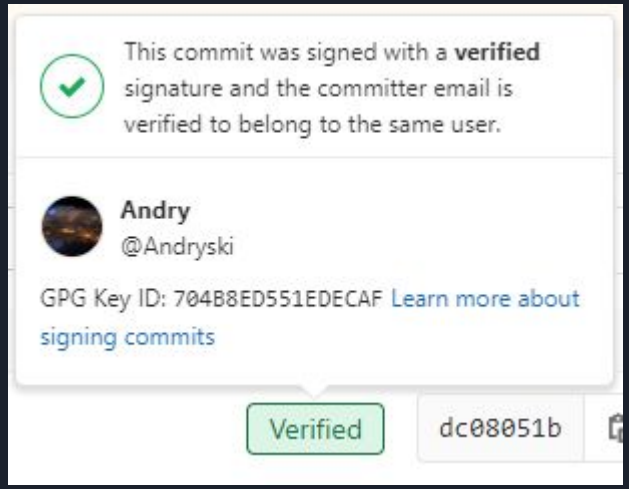
GIT


Certificate Details - Kleopatra
You can use this certificate to secure communication with the following email addresses:
Email
Name
Trust Level
[redacted]
Andry
ultimate
i@andry.io
Andry
ultimate
Add email address
Change passphrase
Generate revocation certificate
Certificate Details
Valid from: 11/16/2019
Expires: never
Type: OpenPGP
Fingerprint: E[redacted]AF
More details...
Export...
Certifications...
Close

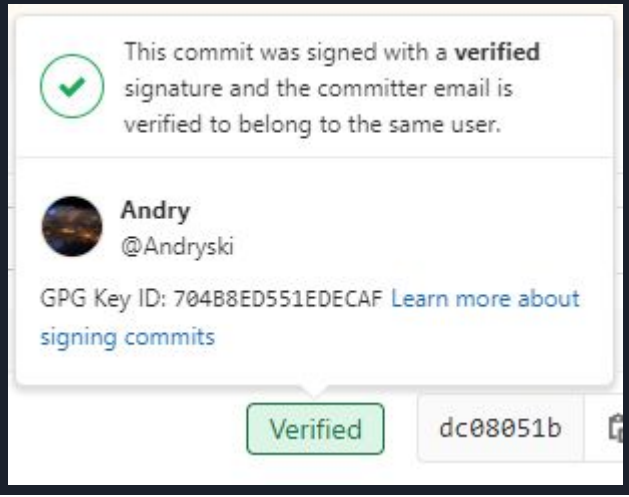
GIT


Certificate Details - Kleopatra
You can use this certificate to secure communication with the following email addresses:
Email
Name
Trust Level
Andry
ultimate
i@andry.io
Andry
ultimate
Add email address
Change passphrase
Certificate Details
Valid from: 11/16/2019
Expires: never
Type: OpenPGP
Fingerprint:
More details...
Export...
Certifications...
Add New User-ID - Kleopatra
Name: Andry
(optional)
EMail:
(optional)
Comment:
(optional)
This is how the new User-ID will be stored in the certificate:
Andry
OK
Cancel

GIT


This commit was signed with a **verified** signature and the committer email is verified to belong to the same user.
Andry
@Andryski
GPG Key ID: 704B8ED551EDECAF Learn more about signing commits
Verified
dc08051b

GIT

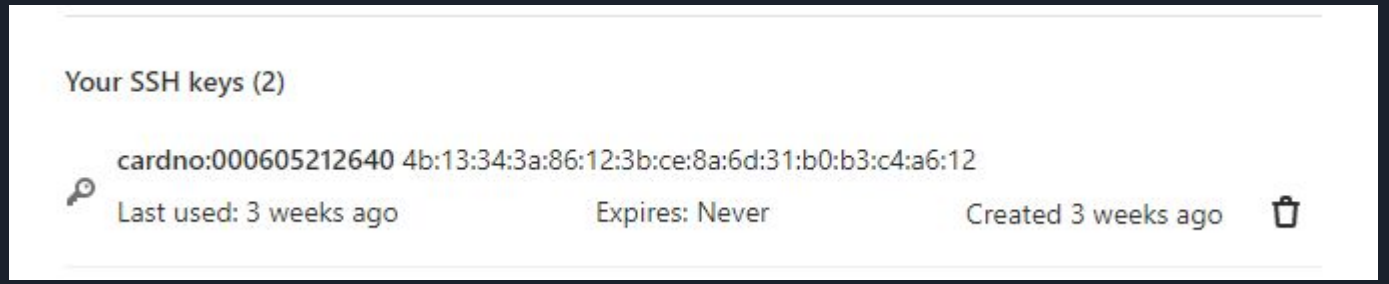

This commit was signed with a **verified** signature and the committer email is verified to belong to the same user.
Andry
@Andryski
GPG Key ID: 704B8ED551EDECAF Learn more about signing commits
Verified
dc08051b


Your GPG keys (1)
Verified
i@andry.io
Unverified

GIT + SSH ?!

GIT + SSH ?!

GIT + SSH + GPG


Your SSH keys (2)
cardno:000605212640 4b:13:34:3a:86:12:3b:ce:8a:6d:31:b0:b3:c4:a6:12
Last used: 3 weeks ago
Expires: Never
Created 3 weeks ago

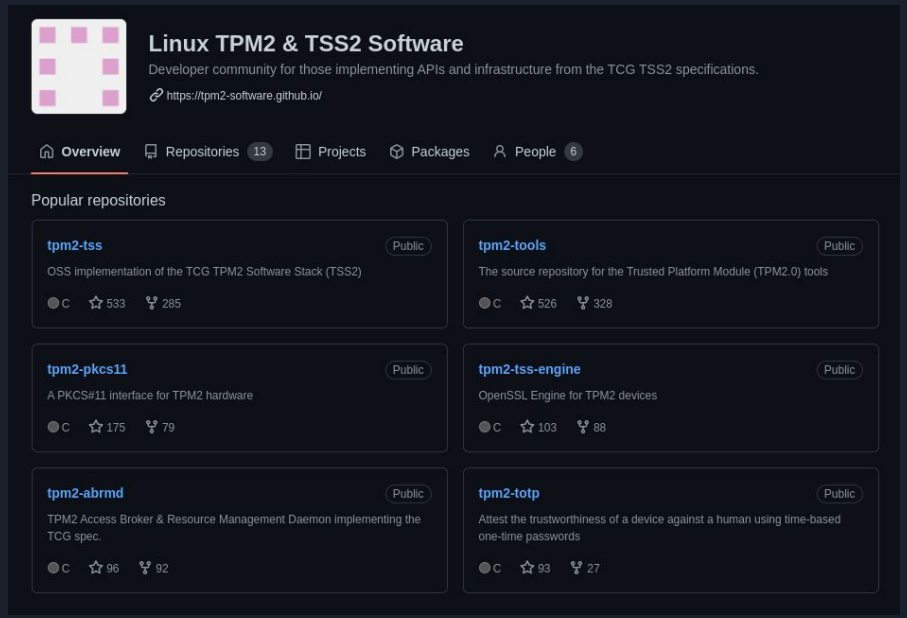
GIT + SSH + GPG

Your SSH keys (2)

cardno:000605212640 4b:13:34:3a:86:12:3b:ce:8a:6d:31:b0:b3:c4:a6:12

Last used: 3 weeks ago Expires: Never Created 3 weeks ago

SSH + TPM + Linux (Dependencies)



https://github.com/tpm2-software

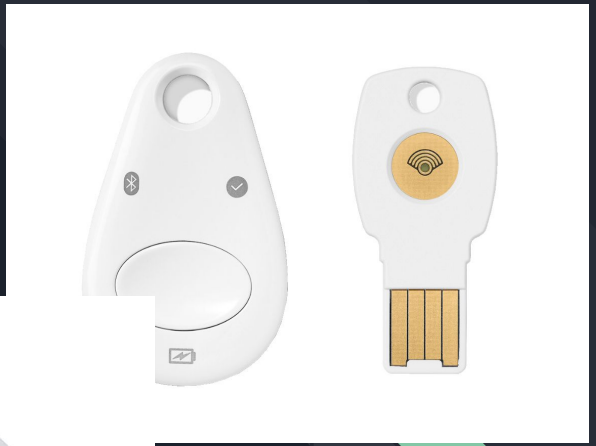
$ sudo pacman -S tpm2-tss tpm2-abrmd tpm2-pkcs11 tpm2-tools

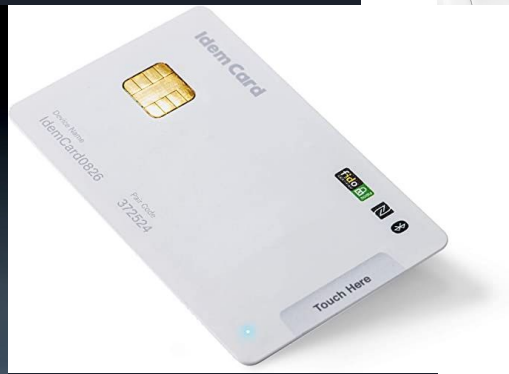
SSH + TPM + Linux

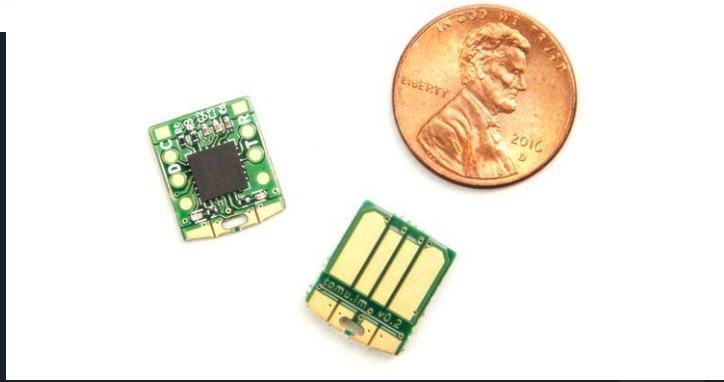
```
# Создаем новый слот в tpm
tpm2_ptool init
# Генерируем ключ
tpm2_ptool addtoken --pid=1 --label=ssh-key --sopin=XXXX --userpin=YYYY
tpm2_ptool addkey --label=ssh-key --userpin=YYYY --algorithm=ecc256

ssh-keygen -D /usr/lib/pkcs11/libtpm2_pkcs11.so
# Проверка что все ок
ssh -I /usr/lib/pkcs11/libtpm2_pkcs11.so user@example.com
# Добавляем в ssh-agent
ssh-add -s /usr/lib/pkcs11/libtpm2_pkcs11.so

# PROFIT
ssh user@example.com
```

Аналоги?

Есть !)







NO FF ONE 2022

аналоги

name	FIDO	FIDO2	OTP	GPG		opensource
Nitrokey						
Thetis Fido						
Trezor						
Tomu						
OnlyKey						
Librem Key						
U2F Zero						
GOOGLE TITAN						
HyperFIDO						
OpenSK						
solokey						

Yubikey-Analogues

name	FIDO	FIDO2	OTP	GPG	static password	static password manager	PIV	Challenge-response	version with nfc	can use self FW	opensource
Yubikey 5	+	+	+	+	+	-	+	+	+	-	-
Nitrokey	+	+	+	+		+?	+	-	-	-	+
Thetis Fido	+	-	-						-	-	-
Trezor	+	+	-						-	-	-
ledger	+	-	-						-	-	-
Tomu	+	+	?						-	+	+
OnlyKey	+	+	+						-	-	-
Librem Key	-	-	+						-	-	+?
U2F Zero	+	-	-						-	+	+
GOOGLE TITAN	+	-	-						-	-	-
HyperFIDO	+	+	-						-	-	-
OpenSK	+	+	-	-	-	-	-	-	-	+	+
solokey	+	+	-	-	-	-	-	-	+	+(on unlock wersion)	+
solokey 2	+	+	+	+?	?	?	+	?	+	+(on unlock wersion)	+

https://habr.com/ru/post/329648/
https://tomu.im/womu.html
https://habr.com/ru/company/globalsign/blog/500356/
https://www.yubico.com/products/compare-products-series/
https://www.yubico.com/solutions/passwordless/
https://www.nitrokey.com/
https://github.com/danstiner/rust-u2f
https://romannurik.github.io/SlidesCodeHighlighter/
https://developers.yubico.com/WebAuthn/
https://www.yubico.com/
https://raymii.org/s/tutorials/Three_New_Nitrokeys_Pro_2_Storage_2_and_Fido_u2f.html#toc_11
https://github.com/drduh/YubiKey-Guide#verify-yubikey
https://codingnest.com/how-to-use-gpg-with-yubikey-wsl/
https://habr.com/ru/post/354638/
https://habr.com/ru/post/305508/
https://webauthn.io/
https://codingnest.com/how-to-use-gpg-with-yubikey-wsl/
https://github.com/NZSmartie/npiperelay (not work - analog wsl-ssh-pagean)

Q&A

Спасибо за внимание

Feel free to contact us:
Tg: @Andryski @n0nvme

Telegram

Offtop &)