

17 FEB 2022

# MATILLION SECURITY CHAMPS

WEBAUTHN, WEBPKI, CRYPTO TOPICS AND STUFF

Hi, I'm

J.C. JONES

- Cryptography Engineer & SRE @ Let's Encrypt

BUT NOT HERE  
ON THEIR BEHALF

- Formerly:  
Mozilla Crypto Engineering

[HTTPS://INSUFFICIENT.COFFEE/](https://insufficient.coffee/)

@CIPHERCOFFEE

JC@ [INSUFFICIENT.COFFEE](https://insufficient.coffee)  
[LETSENCRYPT.ORG](https://letsencrypt.org)



# ETIQUETTE

I can talk about this stuff all day. **DON'T LET ME**

Unclear? Question? **INTERRUPT ME!**

**Q&A IS MY GOAL**

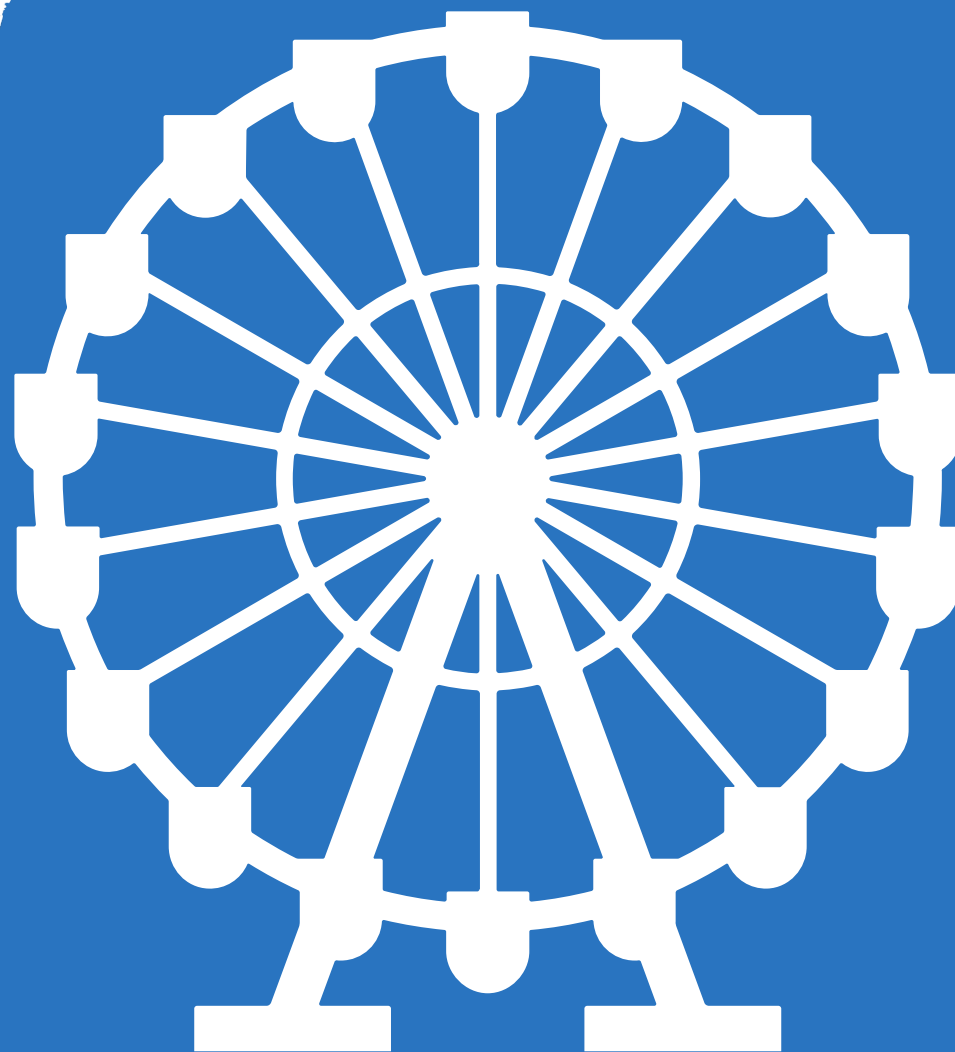
**I HAVE LOTS OF SLIDES...  
BUT BONUS IF WE USE THEM DURING Q&A!**

# MAP

War Stories

Web PKI

Combating  
Phishing:  
WebAuthn



# THE WEB PKI

ALAS, FOR THE GLOSS HAS LONG AGO GONE DARK

# WEB PKI: GRANDIOSE BEGINNINGS

THE "DIRECTORY" WILL SAVE US

# A BRIEF HISTORY OF THE WEB PKI

- 1988-1990 - X.500 shall be everything and X.509 shall define certificates
- 1995 - Netscape creates SSL, shoehorns in X.509 as the guiding format
- 1995 to ~1998 - Certificate Authorities that pay Netscape get added Navigator's trust anchors
- Aug 2000 - AICPA and CICA publish first WebTrust standard
- 2001 - Microsoft requires WebTrust for all included CAs
- March 2004 - Mozilla CA Certificate Policy public drafts begin, first public audit reviews
- Nov 2005 - First Mozilla CA Certificate Policy published, referencing WebTrust, ETSI, and ANSI X9.79-1
- 2005 - CA/Browser Forum formed, partly to homogenize Microsoft and Mozilla audits

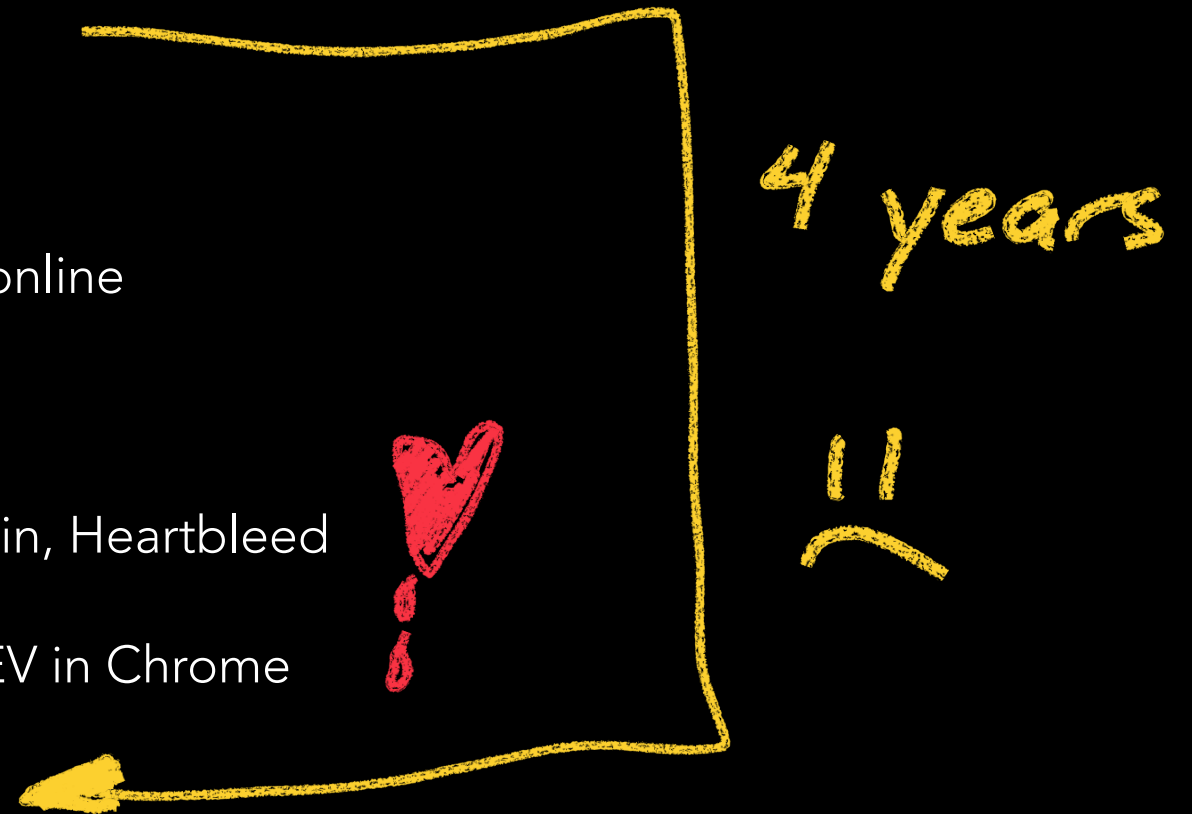
# WEB PKI AFTER 2010: BATTEN DOWN THE HATCHES

WAIT, BEING A CA IS NOT JUST A MONEY-PRINTING PRESS?



# WEB PKI IN THE 2010S

- Sept 2011 - DigiNotar gehackt!
- Nov 2011 - First Baseline Requirements published
- 2012 - Breaking SHA1 signatures becomes feasible
- Dec 2012 - Turktrust MITM intermediate
- March 2013 - First Certificate Transparency log goes online
- Dec 2013 - ANSSI MITM intermediate
- 2014 - Common CA Database, Mozilla CA audits begin, Heartbleed
- January 2015 - Certificate Transparency required for EV in Chrome
- 2016 - SHA1 retired from signatures in the Web PKI
- April 2018 - Certificate Transparency required for all certs in Chrome
- Sept 2019 - Chrome removes EV information from the nav bar



# WHAT DOESN'T SUCK ABOUT THE WEB PKI

IT WORKS SO WELL FOR HOW BROKEN IT IS

# WEB PKI STUFF WHAT AIN'T BROKE

- Certificate Transparency
  - Public Auditing
- Tightened Validation Mechanisms
- Increased Agility
  - Cert lifetimes
  - Automation
- Revocation

# WHAT IS WEB AUTHENTICATION

PHISHING SUCKS, LET'S FIX IT

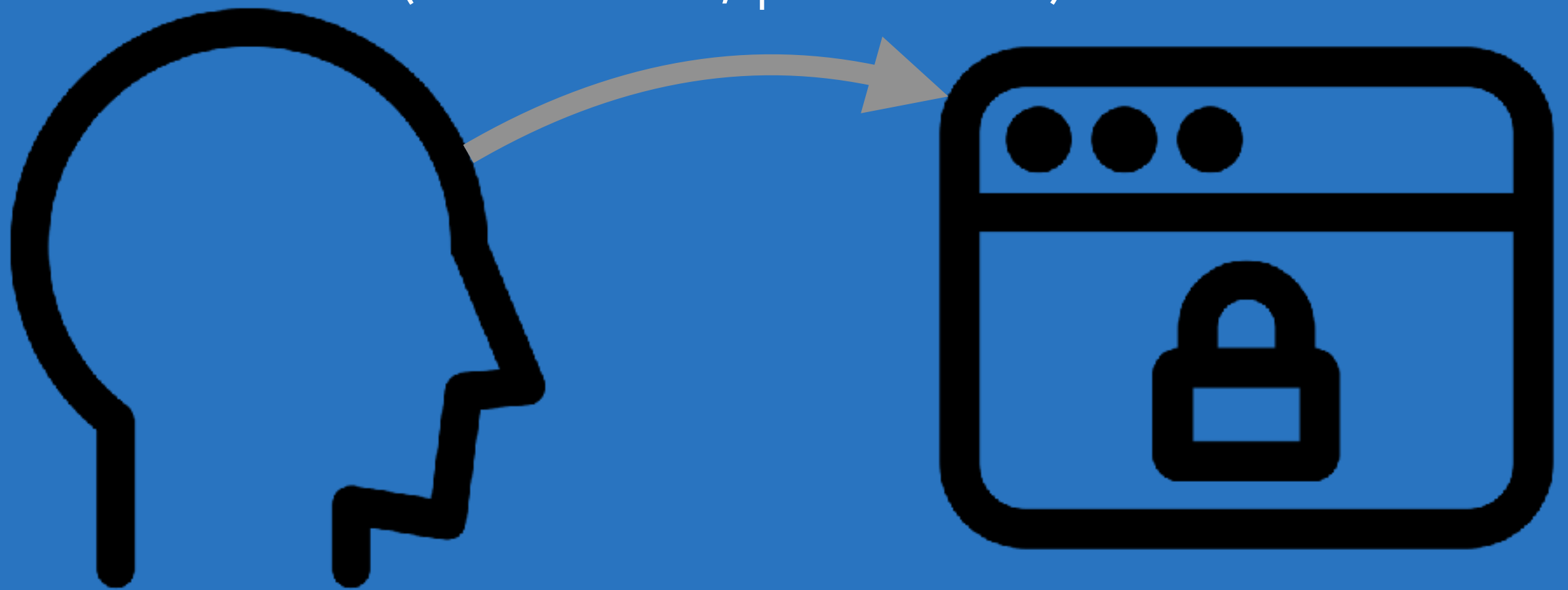
# STANDARD AUTHENTICATION

A yellow sticky note with a torn, textured edge is positioned in the bottom right corner. It contains two lines of blue handwritten text. The first line reads 'USER: GOND' and the second line reads 'P/W: LOG1234'.

USER: GOND  
P/W: LOG1234

# LOGIN

(username, password)

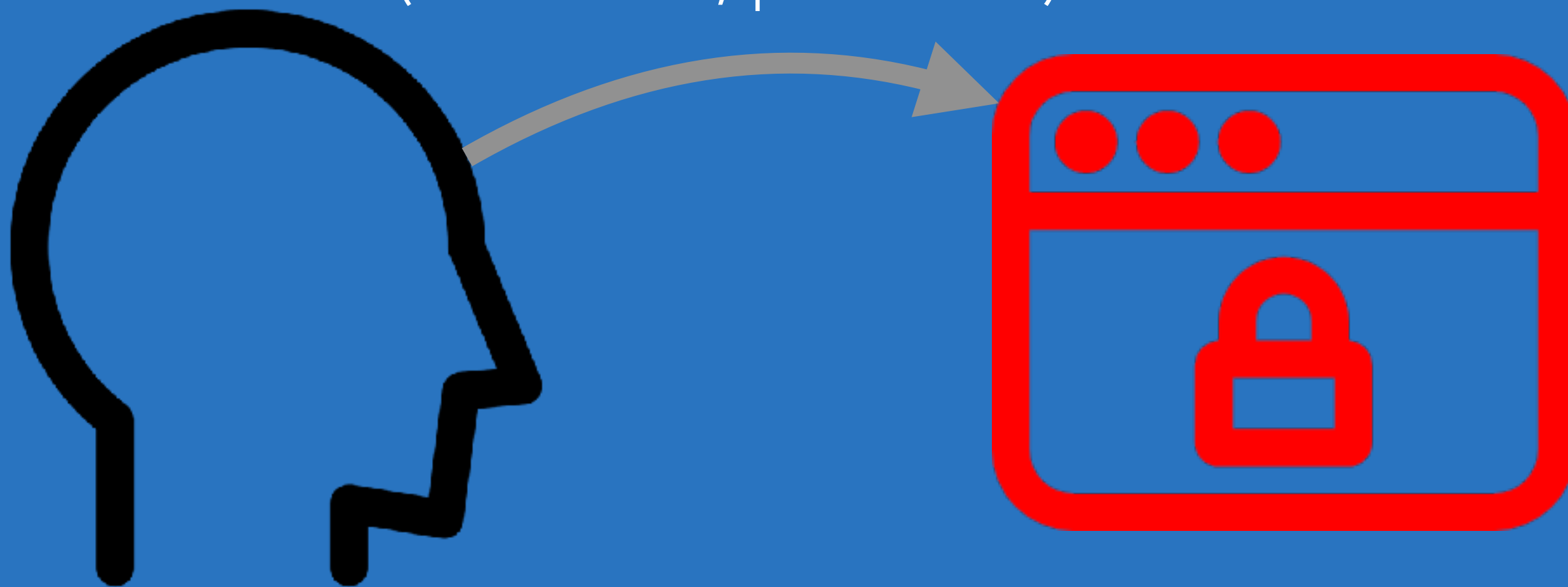


# PHISHING

DUPING SOMEONE INTO GIVING UP THEIR SECRETS

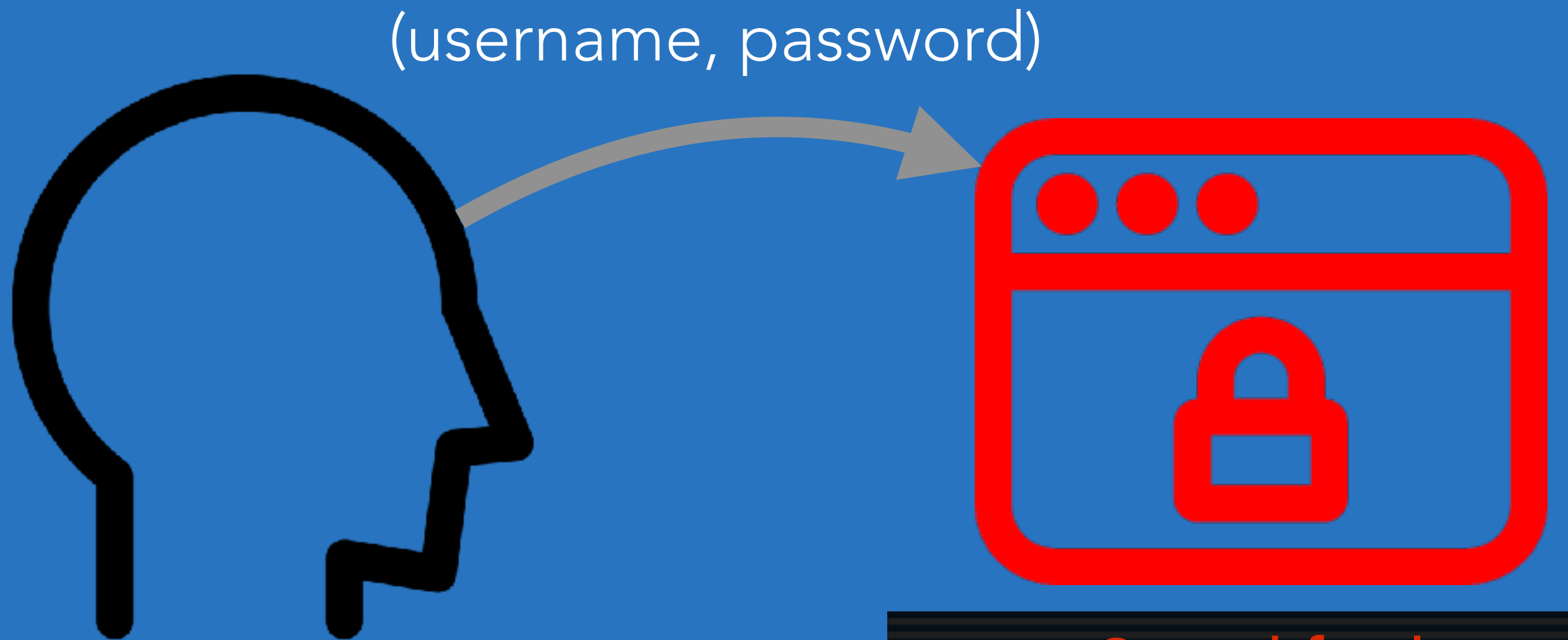
# PHISHING

(username, password)





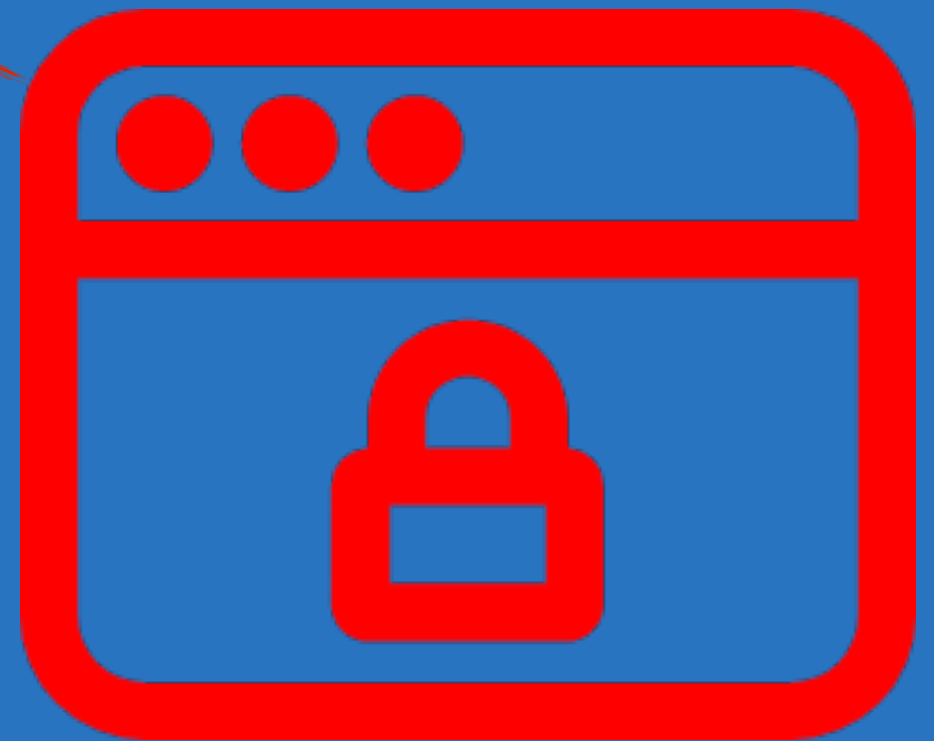
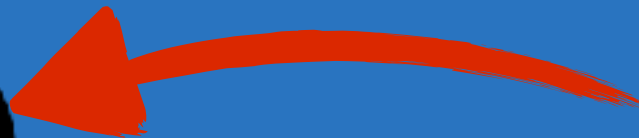
# PHISHING



Saved for later:  
(username, password)

# PHISHING

Stolen (username, password)



It's a replay attack.

BUT WAIT, WHAT ABOUT SMS OR TOTP 2FA?

# PHISHING WITH CODES

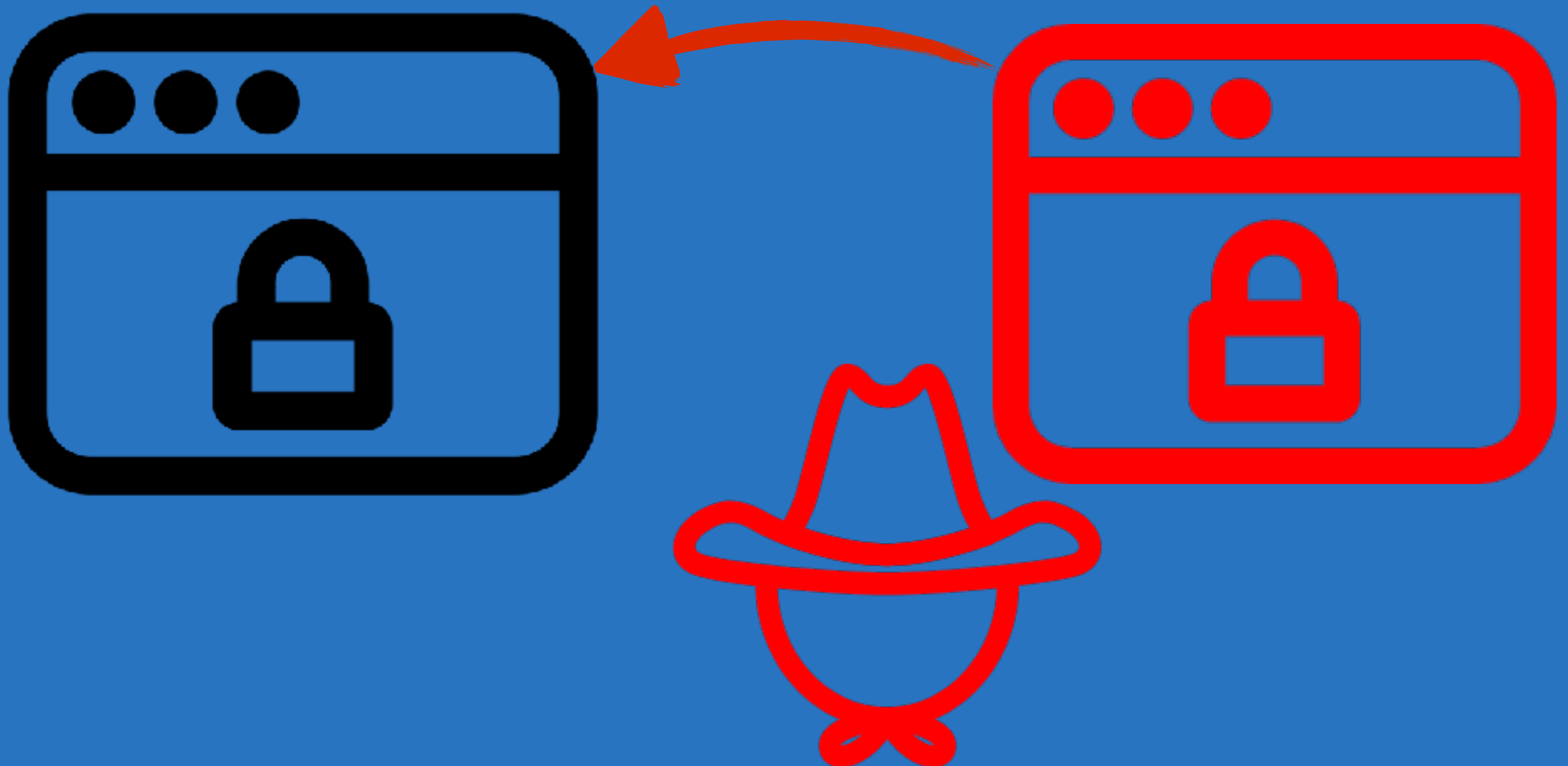
(username, password, code)



*Use immediately* via automated tools:  
(username, password, code)

# PHISHING WITH CODES

Stolen (username, password, code)



TOTP codes, SMS codes, etc. are still subject  
to replay

# ADD DIGITAL SIGNATURES

AKA: WEB AUTHENTICATION



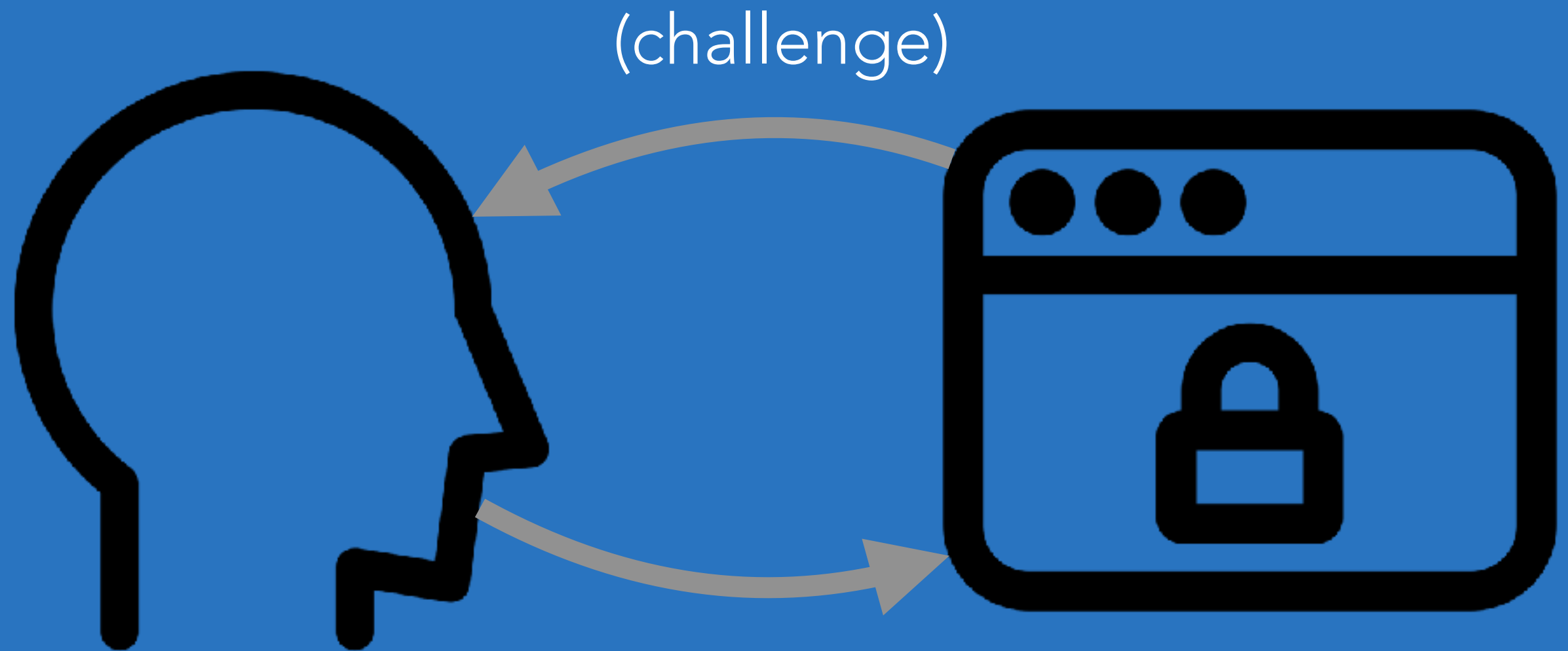
NOTE: THIS FIDO U2F TOKEN IS  
ONE EXAMPLE. THERE ARE  
MORE, AND MORE COMING.





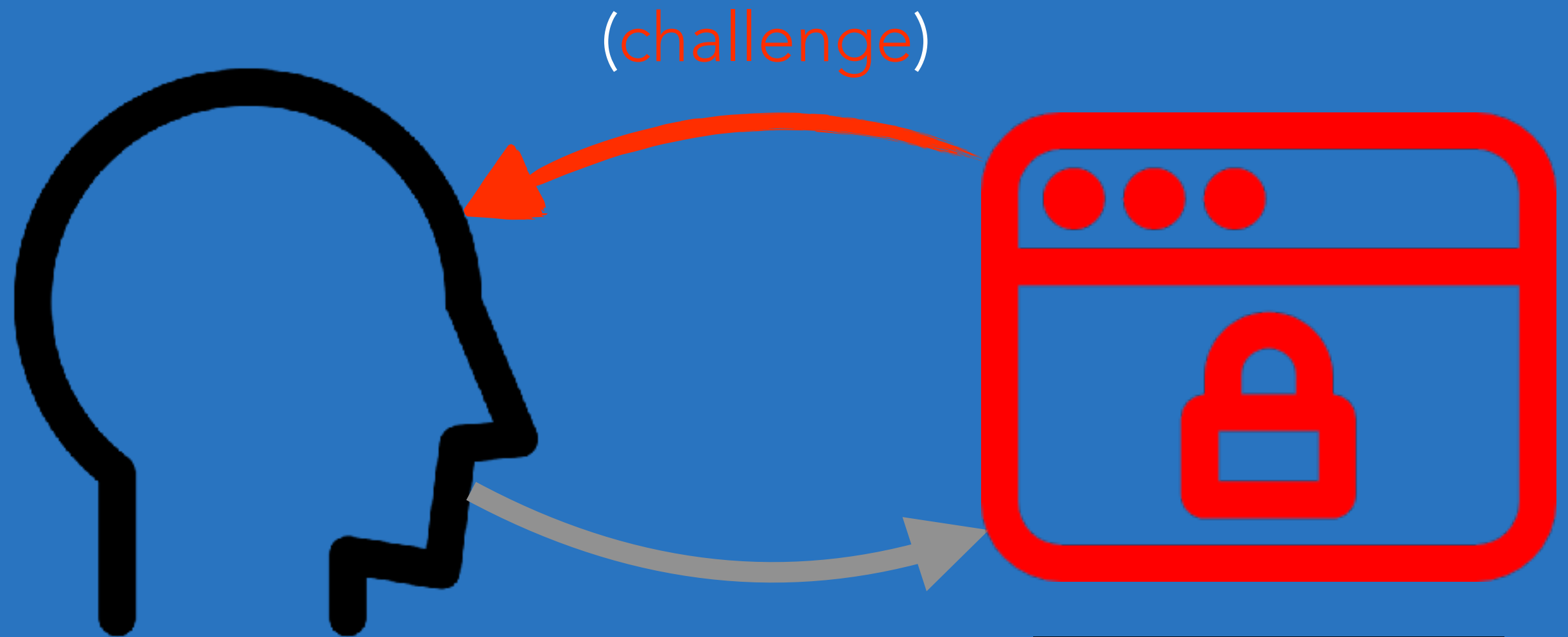
# AUTHENTICATION WITH DIGITAL SIGNATURES

# LOGIN WITH DIGITAL SIGNATURES



(username, password,  
**digital signature(origin, challenge)**)

# PHISHING WITH DIGITAL SIGNATURES

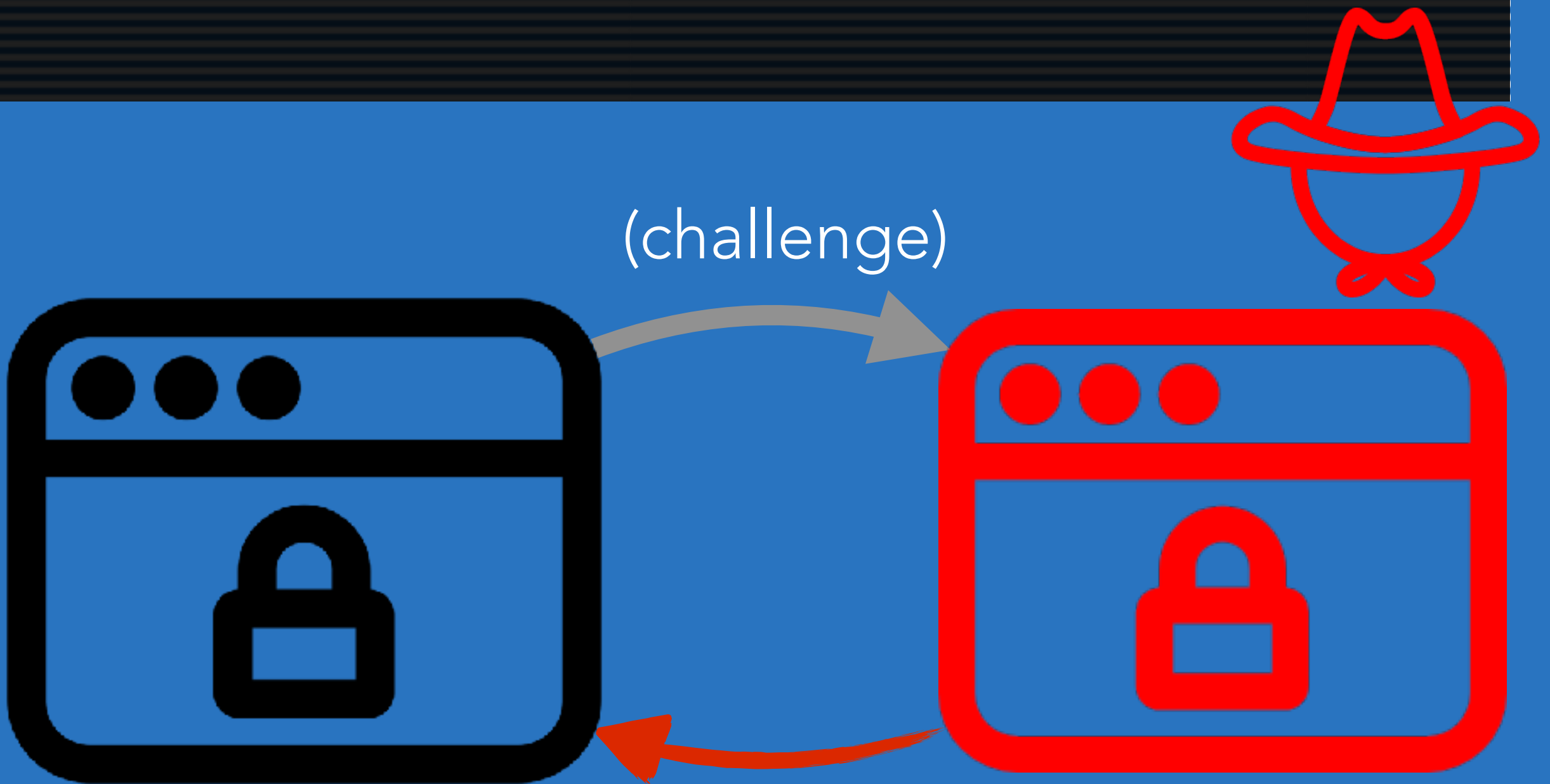


(username, password,

**digital signature(phishing origin, challenge))**

Saved for later

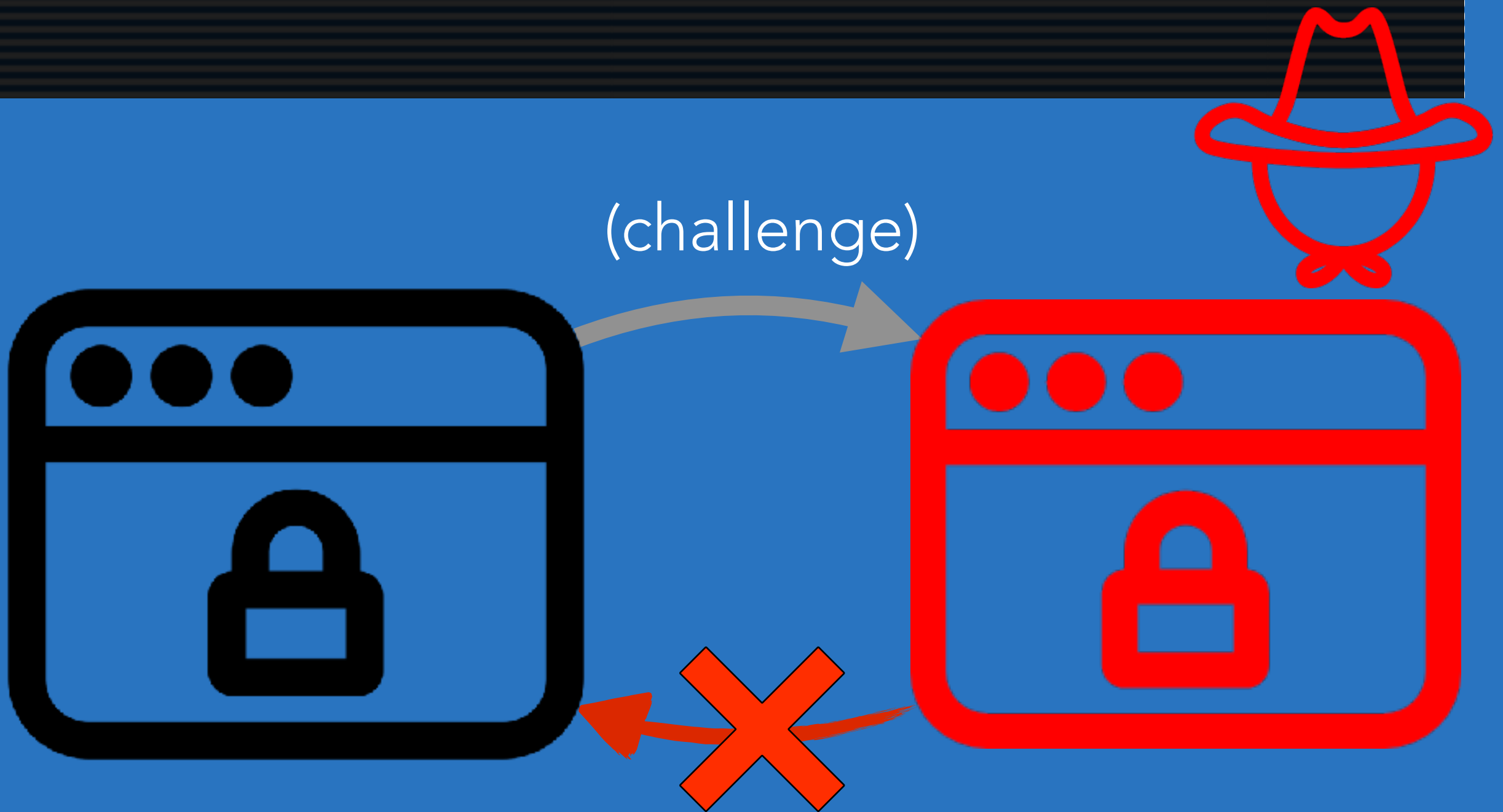
# PHISHING WITH DIGITAL SIGNATURES



(username, password,  
**digital signature(phishing origin,**  
**mismatched challenge))**



# PHISHING WITH DIGITAL SIGNATURES



Unexpected origin, unexpected challenge!

WHERE DOES WEB  
AUTHENTICATION GO FROM HERE?

INQUIRING MINDS WANT TO KNOW...

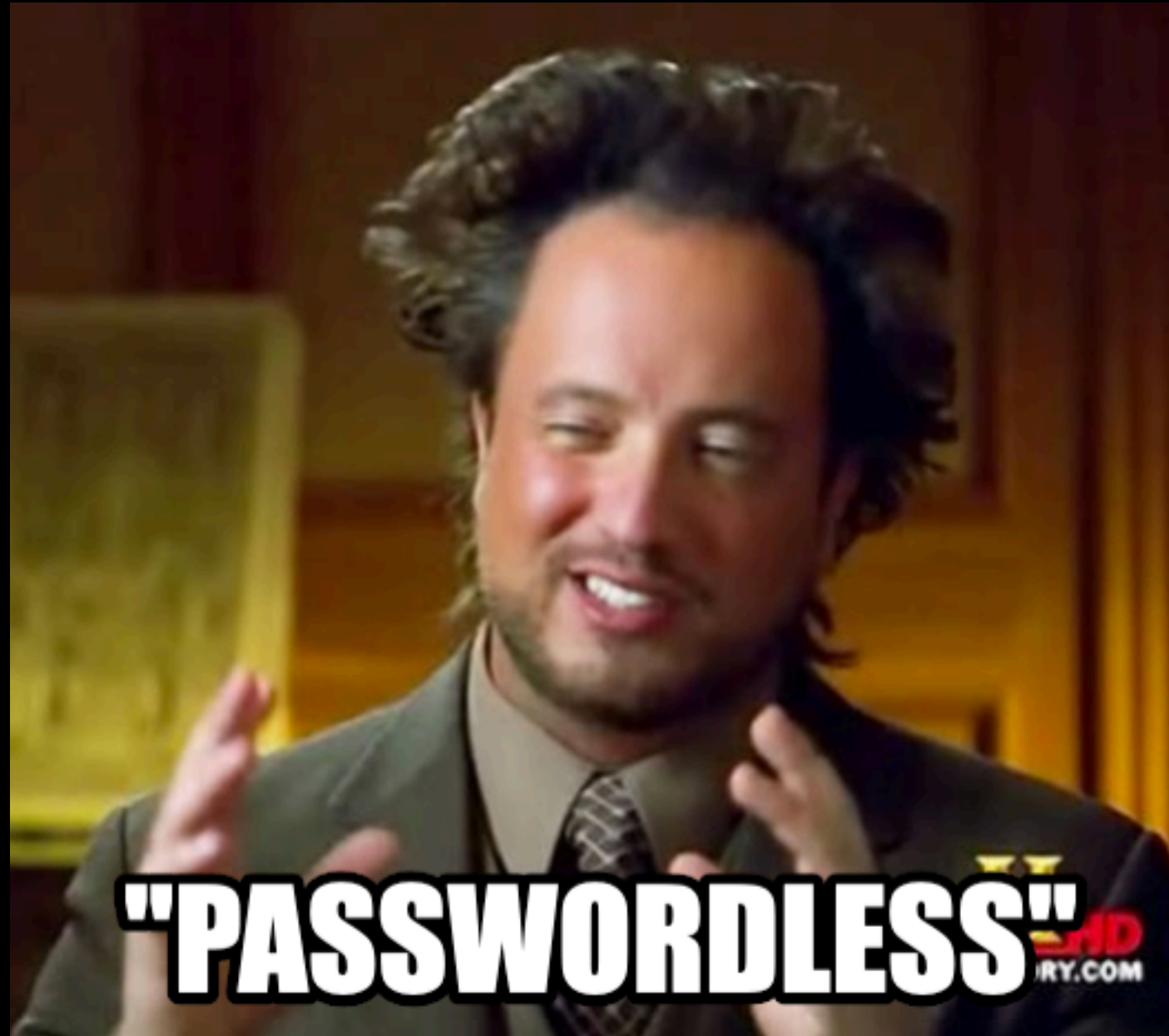
TODAY

WEBAUTHN IS  
USED FOR  
SECOND  
FACTORS

"SOMETHING YOU HAVE"



WHAT HAPPENS NEXT?

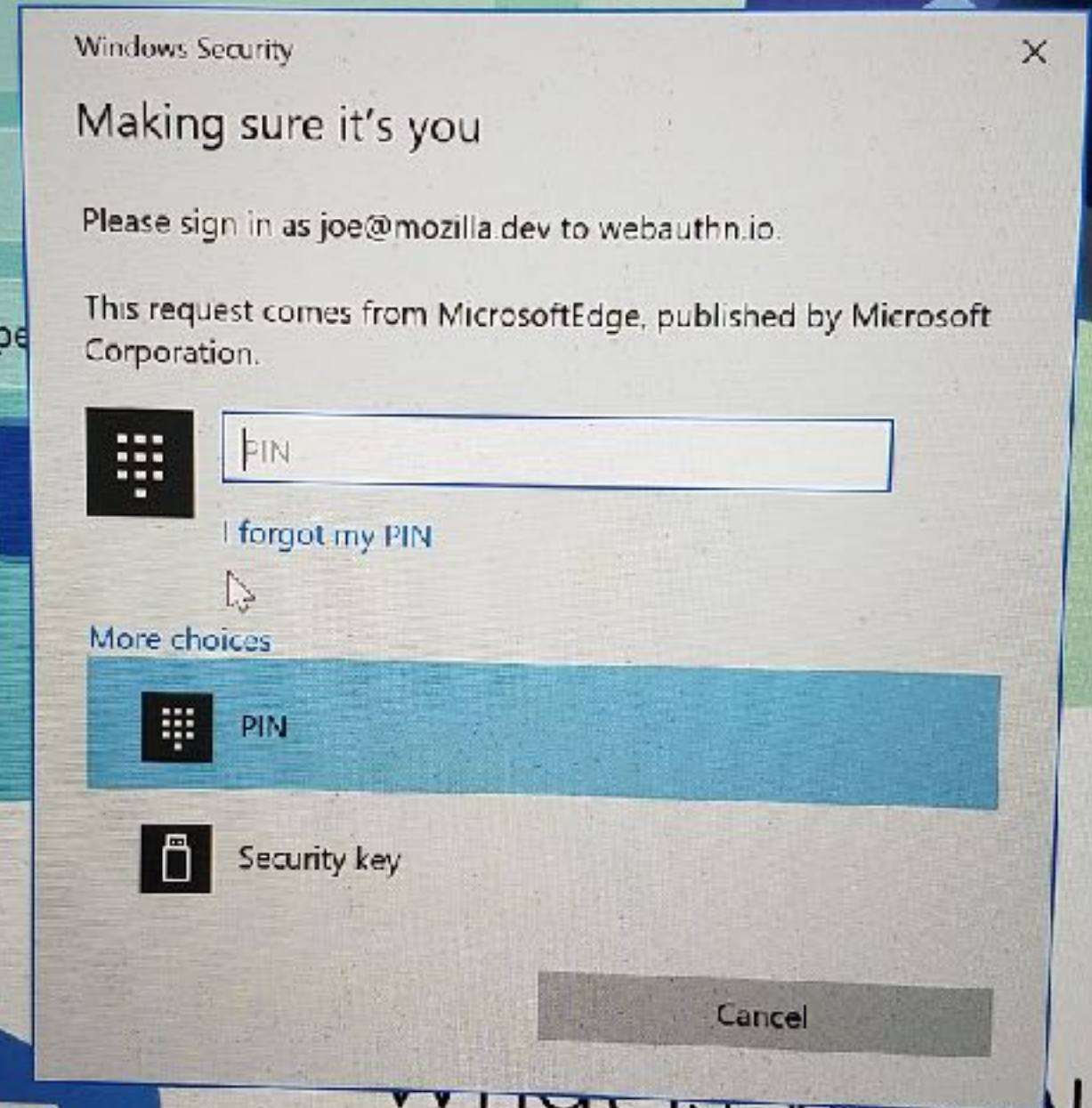




THE PUBLIC KEY IS THE  
ONLY AUTHENTICATION  
NEEDED

# FIRST FACTOR USE CASES

- Something you have, and either:
  - Something you are, or
  - Something you know





webauthn.io


WebAuthn specification

Windows Security

Making sure it's you


Please sign in as joe@mozilla.dev to webauthn.io.

This request comes from MicrosoftEdge, published by Microsoft Corporation.




[I forgot my PIN](#)

More choices



PIN



Security key

Cancel

Welcome to webauthn.io! This site is designed new W3C Specification Web Authentication. W in the Chrome, Firefox, and Edge browsers to d support for credential creation and assertion u those provided by Yubico and Feitian, is suppo code for this demo can be found here on GitHub

how you want to use your key

## Verify your identity

Confirm your fingerprint so Fennec WebAuthn Debug can verify it's you.



Touch sensor

Cancel

Use screen lock



Google Chrome

Touch ID to verify your Identity on webauthn.io →

8

9

0

-

+

delete

U

I

O

P

{

}

|

J

K

L

:

"

return



AUTHENTICATION NOW:  
CAN YOU UNLOCK THAT  
PUBLIC KEY?

Windows Security

Making sure it's you

Please sign in as joe@mozilla.dev to webauthn.io

Verify your identity

Confirm your fingerprint so Fennec  
WebAuthn Debug can verify it's you.



Google Chrome

Touch ID to verify your identity on webauthn.io →

Cancel

Welcome to webauthn.io! This site is designed  
new W3C Specification Web Authentication. W  
in the Chrome, Firefox, and Edge browsers to d  
support for credential creation and assertion u  
those provided by Yubico and Feitian, is suppo  
code for this demo can be found here on GitHub



This is called CTAP2.

Also known as FIDO2.

(CLEARLY RENAMED BY A CAT.)



Not just for built-in authenticators.

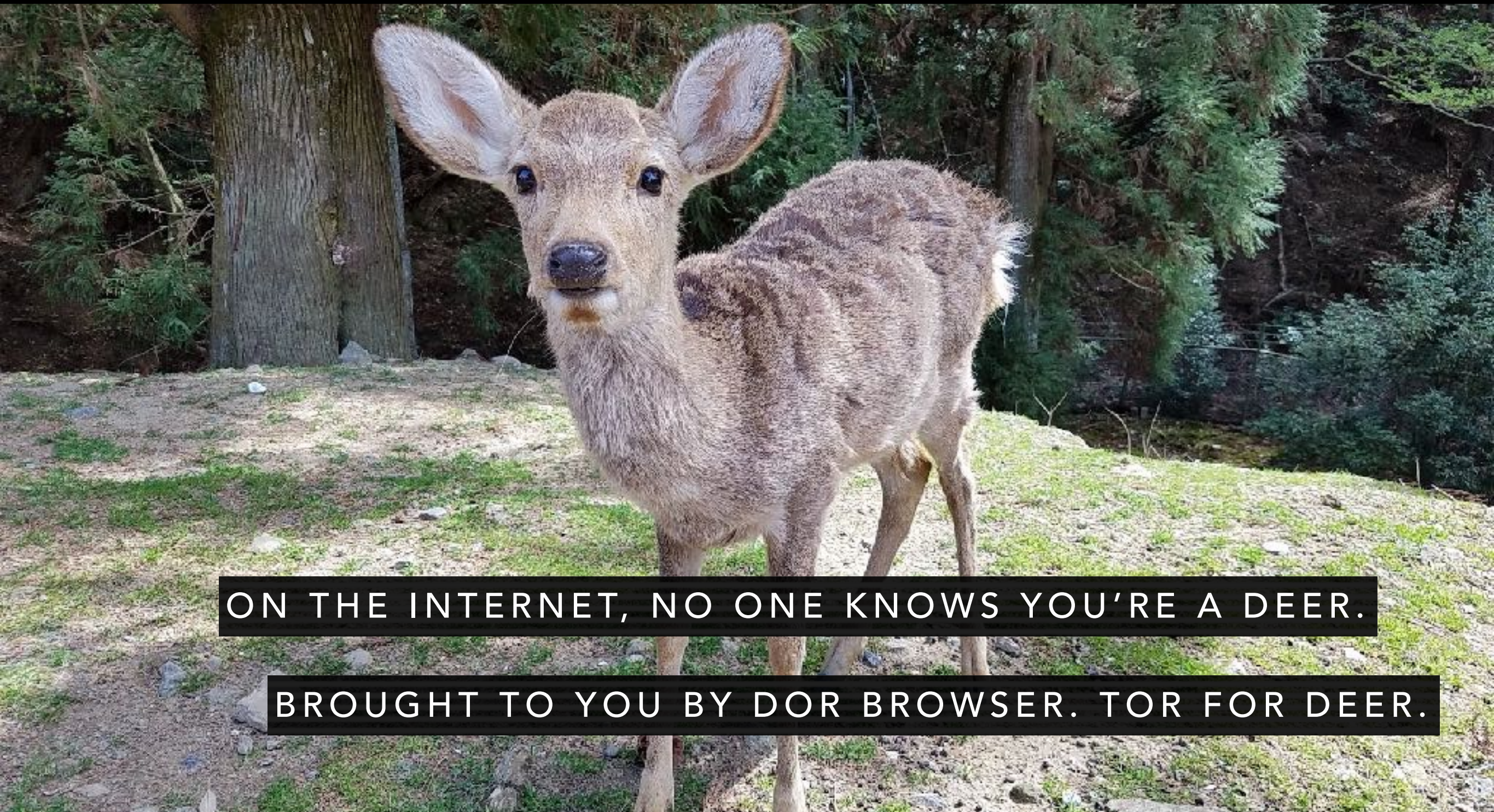
Fingerprint-reader and/or PIN-  
accepting security keys now on  
market.

...WHAT ELSE?



# ANONYMIZATION

YOUR BROWSER SHOULD HELP YOU MANAGE ALL ACCOUNTS  
EVEN ANONYMOUS ONES.



ON THE INTERNET, NO ONE KNOWS YOU'RE A DEER.

BROUGHT TO YOU BY DOR BROWSER. TOR FOR DEER.



# LOSS-OF-DEVICE RECOVERY




HOPEFULLY, IT WILL BE A ZEN EXPERIENCE



# USER EDUCATION

SECURE ACCOUNTS DON'T JUST HAPPEN



グラスは返却口まで  
お戻し下さい  
Glass should return you to  
the return slot

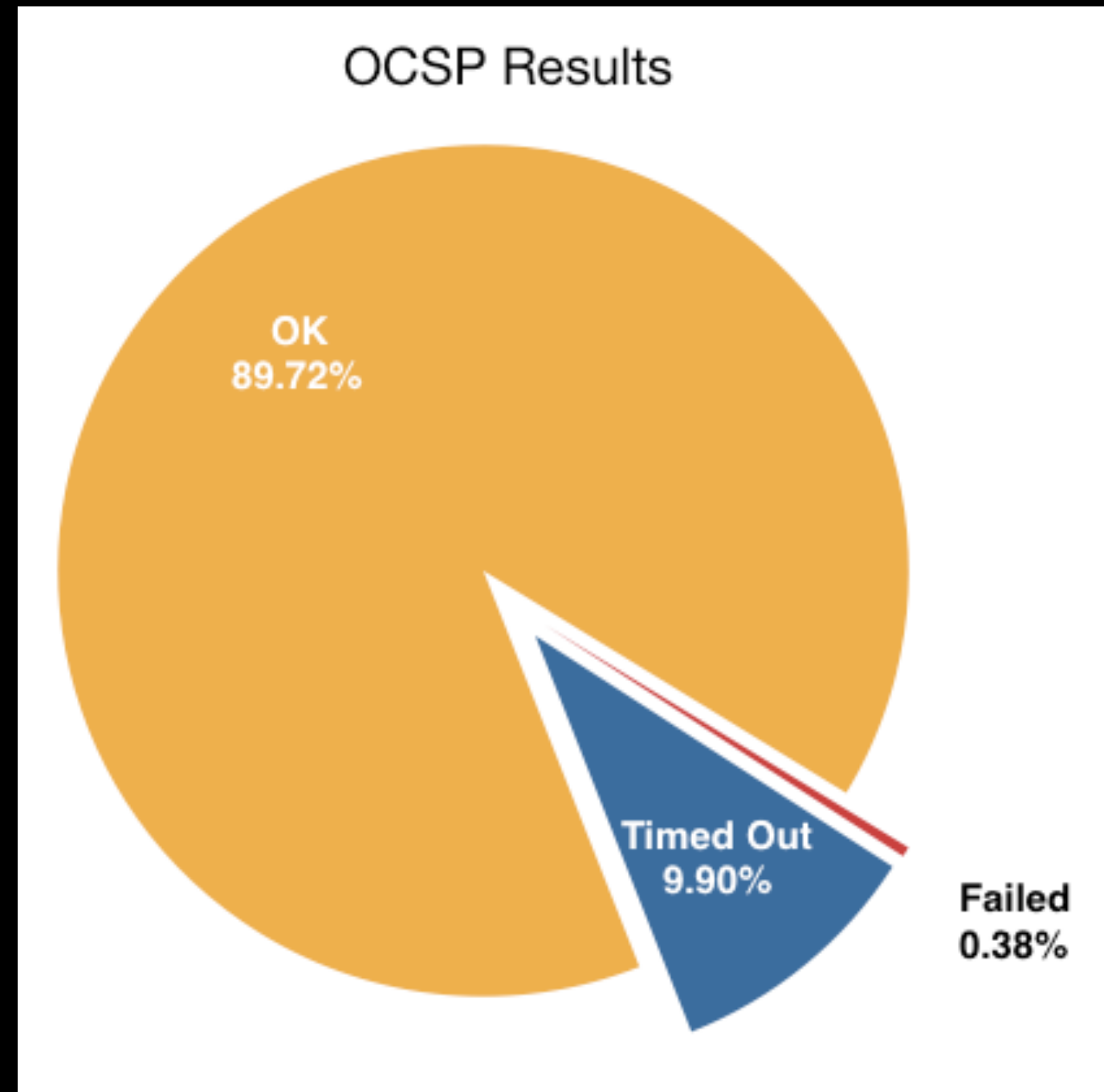
# REVOCATION REDUX: CRLITE

ALL THE BENEFITS OF NOT CHECKING REVOCATION WHILE  
ALSO CHECKING REVOCATION.

WHAT'S NOT TO LOVE?

# STATE OF THE ART

- If you're not an important brand on the Internet, certificate revocation does nothing.
- Only Firefox tries to check every site, and 10% of checks time out (which keeps Firefox slower than Safari or Chrome, ow!)
- OCSP lifetimes mean that even a revoked cert can appear valid for up to additional 7 days.

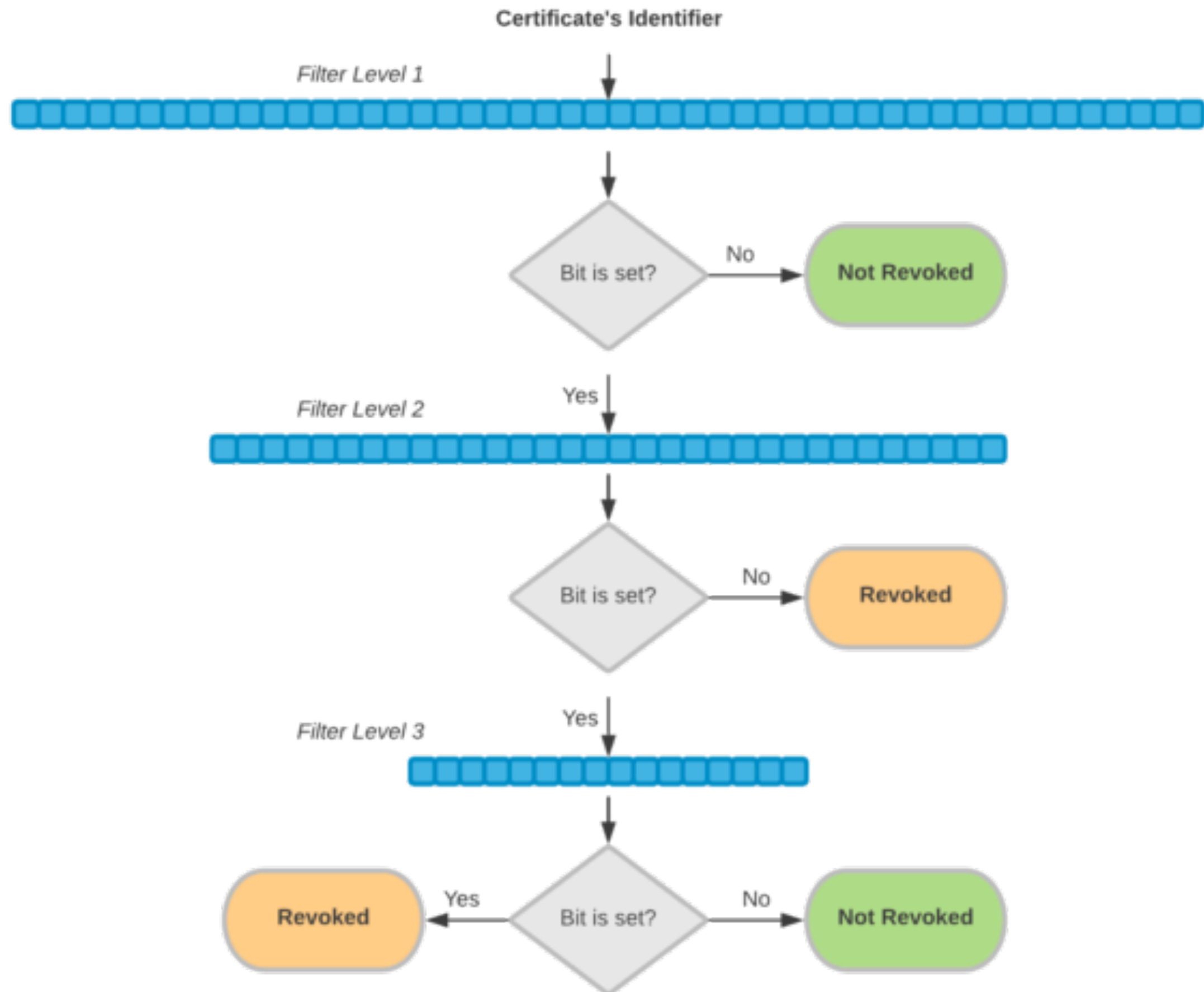


# CRLITE

- Step 1: Gather all revoked-but-unexpired certificates on the Web from CRLs.
- Step 2: Gather all known unexpired certificates on the web from Certificate Transparency (an oracle).
- Step 3: For each CA, derive the sets **UNEXPIRED\_VALID** and **UNEXPIRED\_REVOKED**.
- Step 4: Losslessly compress the sets with cascading Bloom filters.
- Step 5: Ship it regularly.

# LOSSLESS BLOOM FILTER COMPRESSION

- Normally Bloom filters have a false positive rate.
  - If you have an “oracle” of all possible values though, you can make many layers of filter by testing the whole body of data for false positives:
    - If you hit a false positive, you add another filter layer.
- Certificate Transparency gives us that oracle.



I HOPE WE'RE DOING  
Q&A BY NOW

SERIOUSLY HOW DID I GET TO THIS SLIDE

BETTER START TALKING ABOUT CRAZY STUFF

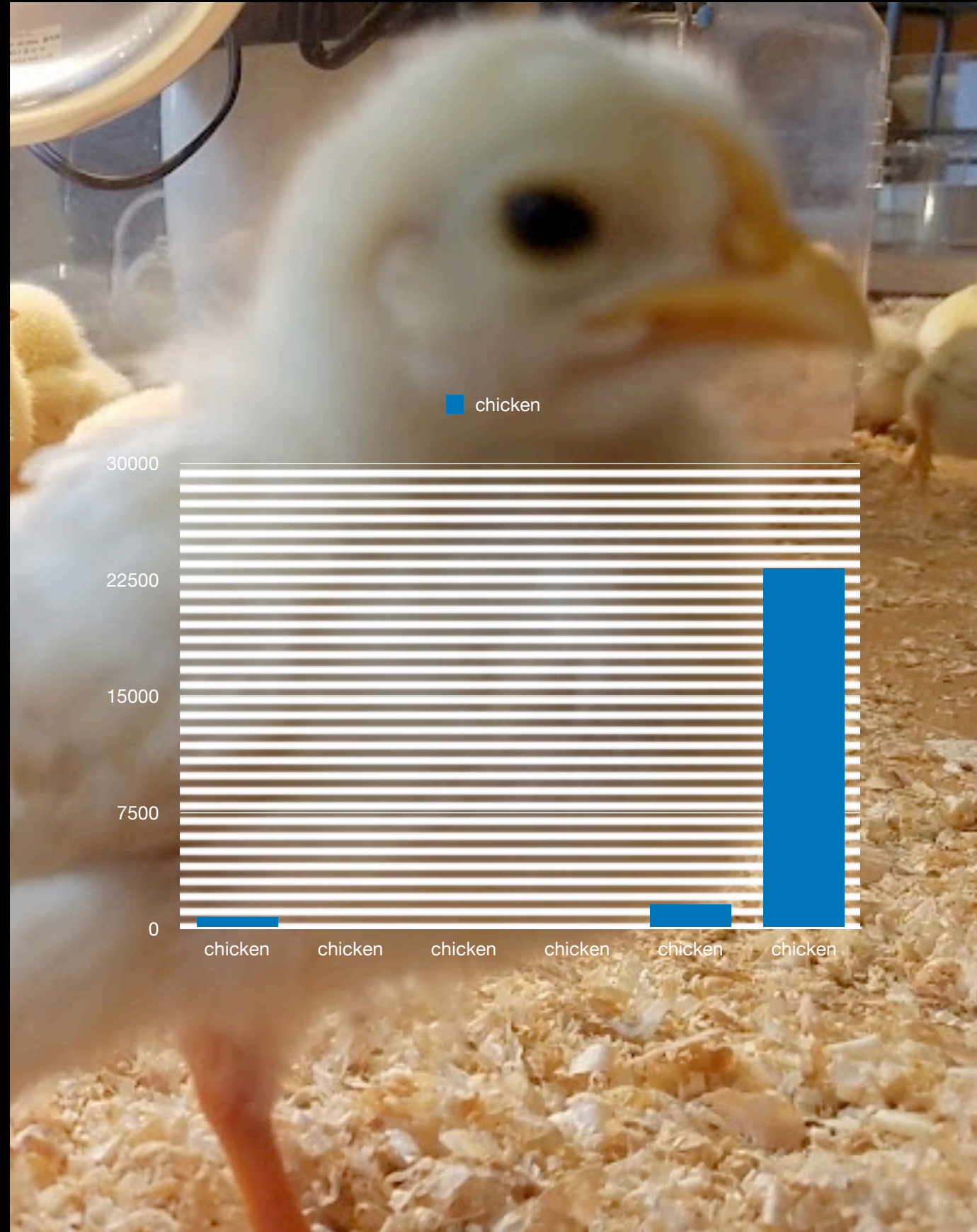


CHICKEN CHICKEN CHICKEN

CHICKEN CHICKEN

CHICKEN  
CHICKEN

chicken

[illegible]

SPKI

# SUBJECT POULTRY KEY INFORMATION