How Secure is Authentication?

How an epic blunder left millions of passwords vulnerable

50,000 lost accounts reportedly hacked

LivingSocial lost password data, says report

That's right, your passwords are up for grabs

50,000

At least 50,000 lost and stolen passwords

Naked Security

Four weeks ago, Adobe disclosed a sustainable漏洞 that threatened to spread malware attacks by giving criminals access to the raw source code for the company's widely used ColdFusion applications. Now, researchers at the same crack has significantly strengthened the crackers' collective hand by revealing a staggering 1.2 billion passwords used over the years by Adobe and many of them from the FBI, large corporate organizations.

That's because Adobe engineers used reverse encryption to scramble the passwords contained in the 3-gigabyte file that's now available online. They've also almost universally recognized that call for stored passwords to be protected.

Two one-way cryptographic hashing algorithms couldn't be converted back to cryptographic hashes and return them to the threat of cracking by requiring crackers to pass individual password guesses through the same

Russian criminals steal 1.2 billion passwords

By James O'Toole and Jose Pagliery @CNNTech August 8, 2014 6:58 AM ET

NEW YORK (CNNMoney) Russian criminals have stolen 1.2 billion Internet user names and passwords, amassing what could be the largest collection of stolen digital credentials in history, a respected security firm said Tuesday.

There's no need to panic at this point -- Hold Security, the firm that discovered the theft, says the gang isn't in the business of stealing your bank account information.

Instead, they make their money by sending out spam for bogus products like weight-loss pills.

The Milwaukee-based firm, didn't reveal the identities of the targeted websites, citing nondisclosure agreements and a desire to prevent existing vulnerabilities from being more widely exploited.

Hold Security founder Alex Holden told CNNMoney that the trove includes credentials gathered from over 420,000 websites -- both smaller sites as well as "household"

2014 1.2bn?
2013 397m
Dec. 2013 145m
Oct. 2013 130m
May 2013 22m
April 2013 50m
March 2013 50m
Cloud Authentication
Password Issues

1. Password might be entered into untrusted App / Web-site ("phishing")
2. Password could be stolen from the server
3. Too many passwords to remember → re-use / cart abandonment
4. Inconvenient to type password on phone
OTP Issues

1. OTP vulnerable to real-time MITM and MITB attacks.
2. SMS security questionable, especially when Device is the phone.
3. OTP HW tokens are expensive and people don’t want another device.
4. Inconvenient to type OTP on phone.
Implementation Challenge
A Plumbing Problem

User Verification Methods

Applications

Organizations

Silo 1
Silo 2
Silo 3
Silo N

App 1
App 2
New App

Implementation Challenge
A Plumbing Problem
Authentication Needs

Do you want to login?
Do you want to transfer $100 to Frank?
Do you want to ship to a new address?
Do you want to delete all of your emails?
Do you want to share your dental record?

Authentication today:
Ask user for a password
(and perhaps a one time code)
Authentication & Risk Engines

Explicit Authentication

Authentication Server

Purpose

Geolocation (from IP addr.)

Risk Engine
Summary

1. Passwords are insecure and inconvenient especially on mobile devices
2. Alternative authentication methods are silos and hence don‘t scale to large scale user populations
3. The required security level of the authentication depends on the use
4. Risk engines need information about the explicit authentication security for good decision
How does FIDO work?
## FIDO Experiences

### Passwordless Experience (UAF standards)

<table>
<thead>
<tr>
<th>ONLINE AUTH REQUEST</th>
<th>Local USER Verification</th>
<th>SUCCESS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Transaction Detail</strong></td>
<td><strong>Show a biometric or PIN</strong></td>
<td><strong>Done</strong></td>
</tr>
</tbody>
</table>

1. Insert Dongle, Press button
2. Enter PIN
3. Biometric Verification
4. Transaction Completed

### Second Factor Experience (U2F standards)

<table>
<thead>
<tr>
<th><strong>Login &amp; Password</strong></th>
<th><strong>Insert Dongle, Press button</strong></th>
<th><strong>Done</strong></th>
</tr>
</thead>
</table>

1. Login
2. Insert Dongle
3. Press button
4. Authentication Completed
FIDO Universal 2\textsuperscript{nd} Factor (U2F)
How does FIDO U2F work?

Verify user presence
How does FIDO U2F work?

Is a user present?

Can verify user presence

Same Authenticator as registered before?

User Verification

FIDO Authentication
How does FIDO UAF work?

- Can recognize the user (i.e. user verification), but doesn't have an identity proof of the user.
- Same Authenticator as registered before?
- Same User as enrolled before?
- Identity binding to be done outside FIDO: This is the "John Doe with customer ID X".
How does FIDO U2F work?

Verify user presence

How is the key protected?
U2F Protocol

• Core idea: Standard public key cryptography:
  o User's device mints new key pair, gives public key to server
  o Server asks user's device to sign data to verify the user.
  o One device, many services, "bring your own device" enabled

• Lots of refinement for this to be consumer facing:
  o Privacy: Site specific keys, No unique ID per device
  o Security: No phishing, man-in-the-middles
  o Trust: Verify who made the device
  o Pragmatics: Affordable today, ride hardware cost curve down
  o Speed for user: Fast crypto in device (Elliptic Curve)

Think "Smartcard re-designed for modern consumer web"
Relying Party

AppID, challenge

check AppID

FIDO Client / Browser

generate:
key \( k_{pub} \)
key \( k_{priv} \)
handle \( h \)

a; challenge, origin, channel id, etc.

U2F Authenticator

\( k_{pub}, h, \text{attestation cert, signature}(a, fc, k_{pub}, h) \)

store:
key \( k_{pub} \)
handle \( h \)

cookie

Relying Party

\( fc, k_{pub}, h, \text{attestation cert, } s \)
navigator.handleRegistrationRequest({
  'challenge': 'KSDJsdASAS-AIS_ASs',
  'app_id': 'https://www.google.com/facets.json',
},

callback = function(response) {
  sendToServer(response['clientData'], response['tokenData']);
};
User Presence API: Auth.

```javascript
navigator.handleAuthenticationRequest({
  'challenge': 'KSDJsdASAS-AIS_AsS',
  'app_id': 'https://www.google.com/facets.json',
  'key_handle': 'JkjhdsfkjSDFKJ_ld-sadsAJDKLSAD'},

callback = function(response) {
  sendToServer(response['clientData'], response['tokenData']);
};
```

```
{
  "typ": "authenticate",
  "challenge": "KSDJsdASAS-AIS_AsS",
  "cid_pubkey": {
    "kty": "EC",
    "crv": "P-256",
    "x": "HzQwlfXX7Q4S5MtCRMzPO9tOywWjBqr14tJ8",
    "y": "XVguGFLIZx1fXg375hi4-7-BxhM1jw42Ht4"
  },
  "origin": "https://accounts.google.com"
}
```

```
[1, 4, X]
```

```
[32, 1, 4, 32]
```

Authentication Example

![Google Login Page](https://accounts.google.com/ServiceLogin?service=mail&passive=true&rm=false&co...)
Authentication Example
Authentication Example
Authentication Example
FIDO Universal Authentication Framework (UAF)
## FIDO Experiences

<table>
<thead>
<tr>
<th>ONLINE AUTH REQUEST</th>
<th>Local USER Verification</th>
<th>SUCCESS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PASSWORDLESS EXPERIENCE (UAF standards)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><img src="image1" alt="Transaction Detail" /></td>
<td><img src="image2" alt="Show a biometric or PIN" /></td>
<td><img src="image3" alt="Done" /></td>
</tr>
<tr>
<td>Transaction Detail</td>
<td>Show a biometric or PIN</td>
<td>Done</td>
</tr>
</tbody>
</table>

| **SECOND FACTOR EXPERIENCE (U2F standards)** | |
| ![Login & Password](image4) | ![Insert Dongle, Press button](image5) | ![Done](image6) |
| Login & Password | Insert Dongle, Press button | Done |
How does FIDO UAF work?
How does FIDO UAF work?

Can recognize the user (i.e. user verification), but doesn’t have an identity proof of the user.

Same Authenticator as registered before?

Same User as enrolled before?
How does FIDO UAF work?

Can recognize the user (i.e. user verification), but doesn’t have an identity proof of the user.

Same Authenticator as registered before?

Same User as enrolled before?

Identity binding to be done outside FIDO: This this “John Doe with customer ID X”.

User Verification

Authenticator

FIDO Authentication
How does FIDO UAF work?

How is the key protected (TPM, SE, TEE, …)?
What user verification method is used?
Attestation & Metadata

FIDO AUTHENTICATOR

Signed Attestation Object

FIDO SERVER

Verify using trust anchor included in Metadata

Understand Authenticator security characteristic by looking into Metadata (and potentially other sources)
FIDO Fingerprint Authenticator

Relying Party
- Web App
- FIDO Server

UAF Registration

Prepare

Device

- Finger Scanner
  - Use fingerprint recognition as an alternative to entering passwords.
  - Fingerprint manager
    - 1 fingerprint is registered.
  - Change alternative password

Settings
- Screen lock
  - Swipe
- Verify Samsung account
  - Off
- Pay with PayPal
Link your PayPal account to your device to make faster and more secure PayPal payments using any of your registered fingerprints. When your PayPal account has been linked, you will have access to special offers from your favourite stores. If you do not have a PayPal account, sign up for one now.

The FIDO Ready™ support must be installed before you can link your PayPal account.

FIDO Ready™ support
Install

PayPal account
Link
FIDO Authenticator

Prepare

FIDO Server

Web App

App

0

UAF Registration

PayPal

Link your fingerprint with PayPal

Pay simpler, safer, and faster – with your fingerprint.

Link Fingerprint

Sign Up for PayPal
FIDO Authenticator

App

Prepare Legacy Auth +

Initiate Reg.

Web App

FIDO Server

UAF Registration

Log in to register your fingerprint with PayPal

pat@example.com

Log In

0

1

Prev. Next

1 2 3 4 5 6 7 8 9 0

q w e r t y u i o p

a s d f g h j k l

↑ z x c v b n m

Sym ☀

English(UK) ☀

Go

Initiate Reg.
UAF Registration

1. Prepare Legacy Auth + Initiate Reg.

FIDO Authenticator

App

Web App

FIDO Server

PayPal protects your privacy and financial information.

Get $30 credit at your favorite stores. Simply link your PayPal account to create a new account today.

Link Fingerprint

Sign Up for PayPal
UAF Registration

Prepare
Legacy Auth +
Initiate Reg.
Reg. Request + Policy

FIDO Authenticator

Web App

FIDO Server

PayPal protects your privacy and financial information.

Get $30 credit at your favorite stores. Simply link your PayPal account to create a new account today.
Link your fingerprint

Login secured by fido

By tapping Agree you acknowledge you have read and agree to Nok Nok Labs' license terms.

View Terms Agree

UAF Registration

Prepare
Legacy Auth +
Initiate Reg.
Reg. Request + Policy
UAF Registration

1. Prepare Legacy Auth + Initiate Reg.

```
{
  "header": {
    "op": "Reg", "upv": "1.0", "appID": "https://mycorp.com/fido"},
  "challenge": "qwudh827hddbawd8qbdqj3bduq3duq56t324zwasdq4wrt",
  "username": "banking_personal",
  "policy": {
    "accepted": [{
      "authenticationFactor": 00000000000001ff,
      "keyProtection": 000000000000000e,
      "attachment": 000000000000000ff,
      "secureDisplay": 0000000000000001e,
      "supportedSchemes": "UAFV1TLV"}]
  },
  "disallowed": {"aaid": "1234#5678"}
}
```
FIDO Authenticator

App

Web App

FIDO Server


2. Verify User & Generate New Key Pair (specific to RP Webapp)

3. Link your fingerprint to your account.
UAF Registration

FIDO Authenticator

1. Prepare Legacy Auth + Initiate Reg. Request + Policy
2. Reg. Response
3. Verify User & Generate New Key Pair (specific to RP Webapp)
4. FIDO Server
**UAF Registration**

**Key Registration Data:**
- Hash(FinalChallenge)
- AAID
- Public key
- KeyID
- Registration Counter
- Signature Counter
- Signature (attestation key)

FinalChallenge=Hash(AppID | FacetID | tlsData | challenge)
UAF Registration

1. Prepare Legacy Auth +
2. Initiate Reg. Reg. Request + Policy
4. Generate New Key Pair (specific to RP Webapp)
5. Success
FIDO Building Blocks

FIDO USER DEVICE
- Browser/App
  - FIDO Client
  - ASM
  - FIDO Authenticator

RELYING PARTY
- WEB SERVER
  - FIDO Server
  - TLS Server Key
  - Cryptographic authentication key reference DB
  - Authentication keys
  - Attestation key
  - Authenticator Metadata & attestation trust store

Update

Metadata Service

UAF Protocol
AAID & Attestation

FIDO Authenticator
Using HW based crypto
Based on FP Sensor X

FIDO Authenticator
Pure SW based implementation
Based on Face Recognition alg. Y

AAID 1
Attestation Key 1

AAID 2
Attestation Key 2

AAID: Authenticator Attestation ID (=model name)
Privacy & Attestation

Bob’s FIDO Authenticator
Using HW based crypto
Based on FP Sensor X

FIDO SERVER RP1

Model A
Serial #

FIDO SERVER RP2

Model A
Attestation & Metadata

- Signed Attestation Object
  - Verify using trust anchor included in Metadata
  - Understand Authenticator security characteristic by looking into Metadata (and potentially other sources)
Facet ID / AppID

Relying Party

Relying Party Web Application Server

Web App
Windows Appl.
Android App
iOS App

API call
API call
API call
API call

All Facets belong to same AppID

Determine Facet ID

Map Facet ID to AppID

Associate call with AppID

ApplID includes Facet ID?

API call
API call
API call
API call

Resource Access
UAF Authentication

FIDO Authenticator

App
Prepare

Web App

FIDO Server
UAF Authentication

FIDO Authenticator → App

Prepare

Web App

FIDO Server
UAF Authentication

[Diagram showing the flow of FIDO Authenticator, App, Web App, and FIDO Server]
UAF Authentication

FIDO Authenticator

App

Prepare

Web App

FIDO Server

Complete action using

- Internet
- Chrome
- PayPal

Always

Just once

COUNTRY

United States

WHY CAN'T I FIND MY COUNTRY?

ADDRESS
UAF Authentication

1. Prepare Authentication
2. Initiate Authentication

FIDO Authenticator

App

Web App

FIDO Server
UAF Authentication

1. Prepare Authentication Request with Challenge
2. Initiate Authentication
Just a sec – our secure payment technology is working its magic.

```
{
    "header": {
        "op": "Auth", "upv": "1.0", "appID": "https://mycorp.com/fido"},
    "challenge": "triz786ighwer8764g6574234515reg45z",
    "policy": {
        "accepted": [[{
            "authenticationFactor": 0000000000000001ff,
            "keyProtection": 0000000000000000e,
            "attachment": 00000000000000ff,
            "secureDisplay": 00000000000000001e,
            "supportedSchemes": "UAFV1TLV"}],
        "disallowed": {"aaid": "1234#5678"}
    }
}
```
UAF Authentication

Prepares UAF Authentication

Pat Johnson
pat@example.com

Initiate Authentication

1

Verify User & Sign Challenge

(Keyparsed to RP Webapp)
FIDO Server
Web App
App
Prepare
FIDO Authenticator
Initiate Authentication
Auth. Request with Challenge
Auth. Response
Verify User & Sign Challenge
(Webapp)
Pat Johnson
650 Castro Street
Mountain View, CA 94041
United States
SignedData:
- SignatureAlg
- Hash(FinalChallenge)
- Authenticator random
- Signature Counter
- Signature

FinalChallenge = Hash(AppID | FacetID | tlsData | challenge)
Payment complete!

Return to the merchant’s web site to continue shopping

Verify User & Sign Challenge
(Key specific to RP Webapp)
Transaction Confirmation

Device

FIDO Authenticator

1. Initiate Transaction

Relying Party

Web App

2. Authentication Request
   + Transaction Text

3. Display Text, Verify User & Unlock Private Key
   (specific to User + RP Webapp)

4. Authentication Response
   + Text Hash, signed by User’s private key

5. Validate Response & Text Hash using User’s Public Key
Transaction Confirmation

Device

FIDO Authenticator

Browser or Native

SignedData:
- SignatureAlg
- Hash(FinalChallenge)
- Authenticator random
- Signature Counter
- Hash(Transaction Text)
- Signature

FinalChallenge=Hash(AppID | FacetID | tlsData | challenge)

Private Key
(specific to User + RP Webapp)

Relying Party

Web App

Initiate Transaction

Validate Response & Text Hash using User’s Public Key

FIDO Server

2

Response

2

Text

5

Validate Response & Text Hash using User’s Public Key

3

Web App

Display Text, Verify User & Unlock Private Key

1

Transaction Confirmation

2

Request

2

Text

2

Response
The FIDO Authenticator Concept

FIDO Authenticator

- **Attestation Key**: Injected at manufacturing, doesn’t change
- **Authentication Key(s)**: Generated at runtime (on Registration)

**Optional Components**

- **User Verification / Presence**
- **Transaction Confirmation Display**
Using Secure Hardware

FIDO Authenticator in SIM Card

- User Verification (PIN)
- SIM Card
- Attestation Key
- Authentication Key(s)
Client Side Biometrics

Trusted Execution Environment (TEE)

FIDO Authenticator as Trusted Application (TA)

User Verification / Presence

Attestation Key

Store at Enrollment

Authentication Key(s)

Compare at Authentication

Unlock after comparison

Client Side Biometrics

Trusted Execution Environment (TEE)

FIDO Authenticator as Trusted Application (TA)

User Verification / Presence

Attestation Key

Store at Enrollment

Authentication Key(s)

Compare at Authentication

Unlock after comparison
Combining TEE and SE

Trusted Execution Environment (TEE)

FIDO Authenticator as Trusted Application (TA)

User Verification / Presence

Transaction Confirmation Display

Secure Element Attestation Key

Authentication Key(s)

e.g. GlobalPlatform Trusted UI
UAF Specifications

User Agent

Web App (Client) → Relying Party Application

Mobile App

Relying Party Web Application Server

FIDO Client

ASM

FIDO Authenticator

FIDO Server

UAF Client API

UAF Protocol Specification

UAF Client API

UAF ASM API

UAF Authenticator Commands
FIDO & Federation
Complementary

- **FIDO**
  - Insulates authentication server from specific authenticators
  - Focused solely on primary authentication
  - Does not support attribute sharing
  - Can communicate details of authentication to server

- **Federation**
  - Insulates applications from identity providers
  - Does not address primary authentication
  - Does enable secondary authentication & attribute sharing
  - Can communicate details of authentication from IdP to SP

Source: Paul Madsen, FIDO Seminar, May 2014
FIDO & Federation

**First Mile**

- **FIDO USER DEVICE**
  - BROWSER / APP
    - FIDO CLIENT
    - FIDO AUTHENTICATOR

- **FIDO SERVER**
  - UAF Protocol

**Second Mile**

- **IdP**
  - Id DB

- **FEDERATION SERVER**
  - Federation

- **Service Provider**

- **Knows details about the Authentication strength**

- **Knows details about the Identity and its verification strength.**
FIDO & Federation

Frequency of login

- High
- Low

Assurance

- High
- Low

SSO slide

status quo

federation

No more ‘Password123’ bump

Source: Paul Madsen, FIDO Seminar, May 2014
FIDO & Federation

Frequency of login

Assurance

High

Low

status quo

FIDO

Continuum

federation

Source: Paul Madsen, FIDO Seminar, May 2014
FIDO & Federation

Assurance vs. Frequency of login

- FIDO
- FIDO + Federation

Source: Paul Madsen, FIDO Seminar, May 2014
FIDO at Industry Event – Readiness

SIM as Secure Element

Fingerprint, TEE, Mobile

Speaker Recognition

Mobile via NFC

PIN + MicroSD

USB

PayPal

Nok Nok Labs

Synaptics

Trustonic

Agnitio

Plug-up International

Yubico
FIDO Ready™ Products Shipping today

OEM Enabled: Lenovo ThinkPads with Fingerprint Sensors

OEM Enabled: Samsung Galaxy S5 smartphone & Galaxy Tab S tablets

Clients available for these operating systems:

Windows 7  Windows 8  Android  iOS 7

Software Authenticator Examples: Speaker/Face recognition, PIN, QR Code, etc.

Aftermarket Hardware Authenticator Examples: USB fingerprint scanner, MicroSD Secure Element
FIDO is used Today

Alipay Offering Fingerprint Payment Partnering with Samsung

July 16, 2014 By CiW Team — Leave a Comment

Hello there! If you are new here, you might want to subscribe to this topic.

PayPal and Samsung launch FIDO authentication and fingerprint payments for Samsung Galaxy S5

By Adam Vrankuli

February 25, 2014 - The FIDO Alliance has announced the first deployment of FIDO authentication and biometric fingerprint payment options, through a new collaboration between PayPal and Samsung, for the shiny new Galaxy S5 smartphone.

According to the alliance, users of the Galaxy S5 can now login and shop via fingerprint in online, mobile and in-store payments wherever PayPal is accepted. This is made possible through FIDO Ready software and a new embedded fingerprint sensor. Per FIDO specifications, the only information a user’s device shares with PayPal is a unique encrypted key that allows PayPal to verify the identity of the customer without having to store any biometric information on PayPal servers.

Alipay announced its cooperation with Samsung Galaxy S5 to offer Alipay users payment by fingerprint.
Conclusion

• Different authentication use-cases lead to different authentication requirements
• Today, we have authentication silos
• FIDO separates user verification from authentication protocol and hence supports all user verification methods
• FIDO supports scalable security and convenience
• User verification data is known to Authenticator only
• FIDO complements federation

→ Consider developing or piloting FIDO-based authentication solutions

Dr. Rolf Lindemann, Nok Nok Labs, rolf@noknok.com