Recall

Passwords and hashing

Password quality

Today

Authentication factors

Authentication factors

Common formulation:

- something you know
- something you have
- something you are

Two-factor authentication (2FA)

There are lots of mechanisms that we can use to try and authenticate someone, but fundamentally they all come down to one of a limited number of categories.

Something you know

Passwords!

Recovery questions

Countersigns

Social authentication

Passwords are an example of something you know: "if you are who you say you are, you ought to know the password". However, they're hardly the only form of "something you know"!

Password recovery questions are also known as _____, _____ or _____.

Modern password guidance from NIST now recommends that we don't use such things, as they're effectively passwords but not protected nearly so well:

> *A: What's the password?*
> *B: I don't know the password.*
> *A: OK, then what's your mother's maiden name?*
> *B: Ummmm... Smith?*
> *A: Come on in!*

They are particularly risky https://dl.acm.org/doi/abs/10.1145/1408664.1408667, but lots of organizations still do dumb things. Personally, if I'm required to provide answers to such questions, I choose answers that I can remember but which are not true (and therefore won't be found anywhere online).

A countersign) is an example of a limited, non-computer-based, method of challenge-response authentication. In such a protocol, you need to remember not just one password, but potentially several responses to give to different challenges. This provides resistance to eavesdropping attacks if each challenge/response pair can be used only once. We'll discuss cryptographic challenge-response protocols later in the course.

Social authentication is the expectation that, if you are who you say you are, you should know somebody else. You can imagine someone informally saying, "if you're in Term 7 computer engineering, you must know so-and-so!", but social authentication can also be done more formally, e.g., by a social network https://link.springer.com/chapter/10.1007/978-3-642-32946-3_1.

Something you have

In person:

- documents
- signatures
- seals

Remotely?

Verifying the veracity of something you have is tricky, even when we are working in person and I hand you something. In general, people are terrible at matching unfamiliar faces to photos. People who work with documents every day can be trained to spot fakes, but even border agencies have experts to fall back on for detecting high-quality forgeries. That's why it's a little bit silly to think that ordinary software developers should be trusted to verify passports at keysigning parties!

Given the above, how am I to verify that you have something when I'm not even in the same place as you?

Something you have knows...

A secret key?

- symmetric-key MACs, asymmetric-key signatures (later!)
- HOTP (RFC 2446): `Truncate(HMAC-SHA-1(K,C))`

$$h\left((k \oplus p_o)||h((k \oplus p_i)||text)\right)$$

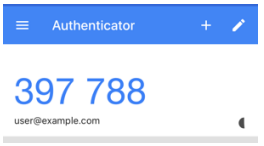
- TOTP (RFC 6238): `HOTP(K,T)`

In practice, when we verify "something you have", what we actually mean is "something known by something you have".

If you are able to produce a valid EMV cryptogram in response to a transaction at a store counter, there's a very good chance that you are in possession of a valid bank card. Manufacturers make it _____ to extract secret keys from such cards.

Authentication tokens

Software



- key stored... where?
- security implications?

Hardware

e.g., U2F (see FIDO spec and now WebAuthn/FIDO2)

If your TOTP key is stored on a computer (e.g., your phone), it's important to think about the failure modes if that device is compromised or lost. If you store your TOTP key on your phone and use it to help you log in on your computer, there is a degree of independence between the two (although that may be less true if you have unencrypted mobile backups). If, however, you store your passwords in a password manager (as you should!) and also store TOTP keys in the same password manager (as some services offer), you may have an eggs-and-basket problem.

Something you actually have

A phone... with a phone number?

PUFs (Physical Uncloneable Functions)*

- use microstructure (manufacturing anomalies) like fingerprints
- unpredictable **but consistent** challenge/response pairs
- can derive a key using a *fuzzy extractor*

---

* Pappu, Recht, Taylor, Gershenfeld, "Physical one-way functions", *Science 297 (5589)*, pp. 2026–2030, 2002. DOI: 10.1126/science.1074376

A phone (or, more precisely, a SIM card) is _____ a great method of authenticating a user for high-value purposes. It's OK to confirm a login to a low-value account, but attackers can trick phone companies into swapping your SIM onto a device they control, often using _____ _____ (thus negating the "second" factor)!

Biometrics

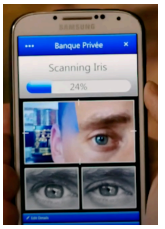
Physical	Behavioural/mixed
fingerprint	gait
hand geometry	typing rhythm
face	mouse patterns
iris	touch patterns
retina	voice patterns

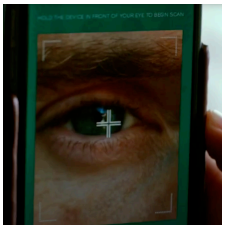
Biometrics are only useful if you have a _____ to avoid _____ _____. A key principle to remember: _____

Iris scan

What's the problem?

- subtle but important difference (top vs bottom)
- if you can't do one...
- somebody tell Samsung*





* 46halbe, Chaos Computer Clubs breaks iris recognition system of the Samsung Galaxy S8, CCC blog, 2017.

In these images (from the BBC's production of The Night Manager), the top images show a banker taking a scan of a character's iris. Later in the series, that character (played by Tom Hiddleston) uses his iris to authorize a banking transaction related to international arms trafficking.

The problem is _____ that phone cameras lack the resolution required to do an iris scan. Rather, it's a subtle difference between the images at the top and the images at the bottom.

If you can't do the thing _____, there's not much point doing the thing _____ _____!

All the CCC folks needed to subvert Samsung's iris scan was a contact lens and a... Samsung printer.

1.5FA

Is your second factor really independent?

Usually "something you know" plus:

- something else you know (may add **very little** value)
- something your computer knows (but can it be **copied**?)
- something you can receive (but is it **just** you?)
- biometric information (which may be **copied** or **replayed**)

Something your computer knows (or is):

- cookies
- fingerprints
- known IPs (more **is** than **knows**)

Non-binary authentication

Doesn't have to be "stranger" or "root"

Multiple factors

Multiple signals

Multiple levels / roles

Multiple factors can be incorporated at different times.

Multiple authentication _____ can be incorporated that would not, on their own, be trusted to provide any assurance. For example, when I go home at the end of the day and try to use some University services from O365 (Teams, OneDriver, etc.), the fact that I've changed IP means that I'm likely to be prompted for an MFA notification. An IP address is _____ _____, but it can be enough to _____.

This is also true for behavioural signals (e.g., logging in at an unusual time).

Authentication shouldn't be all-or-nothing. For example, I can authenticate to a bunch of systems as the user `jon` using an SSH key, and once I'm `jon` I can escalate to `root` via `sudo` if I can prove knowledge of `jon`'s password. This means that getting to `root` is a two-step procedure: it's not possible for anyone (including an attacker) to log in directly as `root`, nor can anyone (on my systems) log in using a username and password alone.

Summary

Passwords

Guessing and entropy

2FA