Modernizing XMPP Authentication and Authorization

Time to forget your passwords...

Matthew Wild

2023-02-05

Introduction

- Prosody (prosody.im)
- Snikket (snikket.org)
- ModernXMPP (modernxmpp.org)
- XMPP Standards Foundation (xmpp.org)

What is authentication?

1. Connect to the server
2. Prove your identity

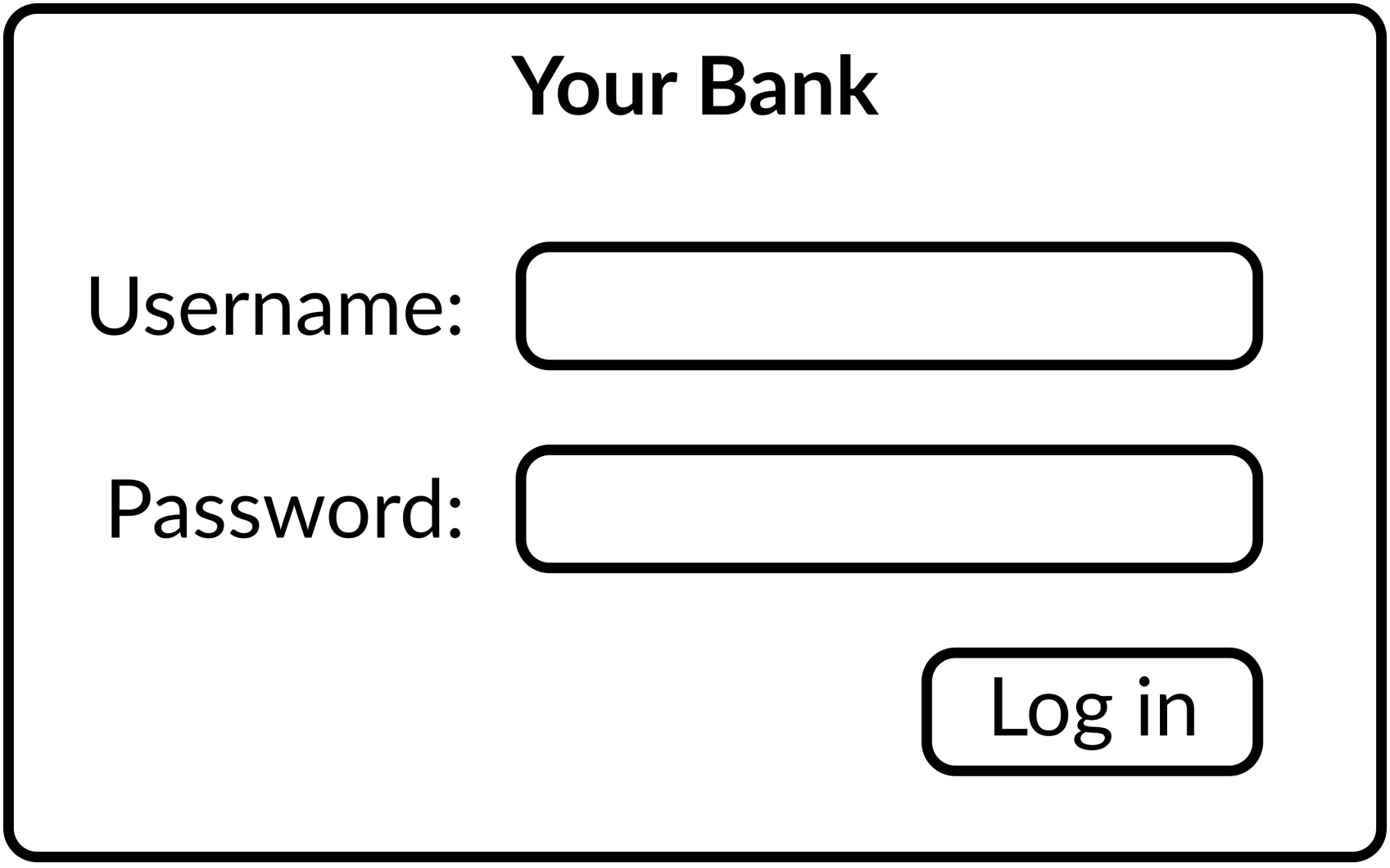
Authentication in the early days

Server: “Hi, who are you?”

Client: “Hi, I’m Matthew. My password is hunter2.”

Server: “Correct password!”

Authentication in websites



Introducing SASL

- Standard authentication protocol
- Used by XMPP, SMTP, IMAP, LDAP, IRC and more
- Current efforts to bring it to HTTP

SASL mechanism: PLAIN

Client: “I’m Matthew, my password is hunter2.”

Server: “Correct password!”

This is fine

Thanks to decades of successful campaigning and public awareness generated by security-minded folk:

- Passwords are long, random and unguessable
- Passwords are frequently rotated
- Passwords never contain personal information
- Passwords only grant access to one service

Oh, right

Just kidding.

SASL mechanism: SCRAM

- Challenge-response
- Multiple round trips
- Features:
  - Client and server only store hashes (this is magic)
  - Client and server only exchange hashes (more magic)
  - Mutual authentication (wizardry of the highest order)
  - Channel binding (!)
- Secure, but at a cost
- It's still password based

What can we do?

Instead of trying to improve password security, the web ecosystem is moving to a world beyond passwords using technologies such as WebAuthn, FIDO2, and Passkeys.

What can *we* do?

Fast Authentication Streamlining Tokens (FAST!).

- Building on earlier work of Florian Schmaus on SASL-HT, a family of token-based single-round-trip SASL mechanisms.
- Client exchanges password (or more) for a token, once.
- Subsequent authentications use faster token auth
- Tokens are long, random and unguessable
- Tokens are unique to one service and one device, frequently rotated, and can be revoked (e.g. if a device is lost or stolen).
- Mutual authentication
- Channel binding

Opening the door to multi-factor auth

- 2FA solves the issue: “sure, they sent the correct password, but is it *really* them?”
- Tokens solve that issue too
- 2FA prompt only required when passwords are used (this is how the web does it - the tokens are your “session cookies”).
- FAST tokens are more secure than cookies

Opening the door to passwordless

It doesn't matter how you obtain a token.

- No authentication - just generate a new account for anyone who asks
  - Only suitable for a single device really, and no account recovery
  - But these things can be solved (e.g. a way to introduce a device)
- Authenticate using something else the user has - email verification, SMS, Passkeys

Further reading

More info about the project can be found in our blog post at https://blog.prosody.im/fast-auth/

And if you liked this insight into the world of XMPP, check out two more XMPP-related talks in this room later today:

- *P10K: getting 10000 participants into a Jitsi meeting*
- *Interoperable Chat, Dutch Healthcare and the Digital Services Act*

Contact me via email or XMPP: me@matthewwild.co.uk

Visit the Realtime Lounge here at FOSDEM (just downstairs).