

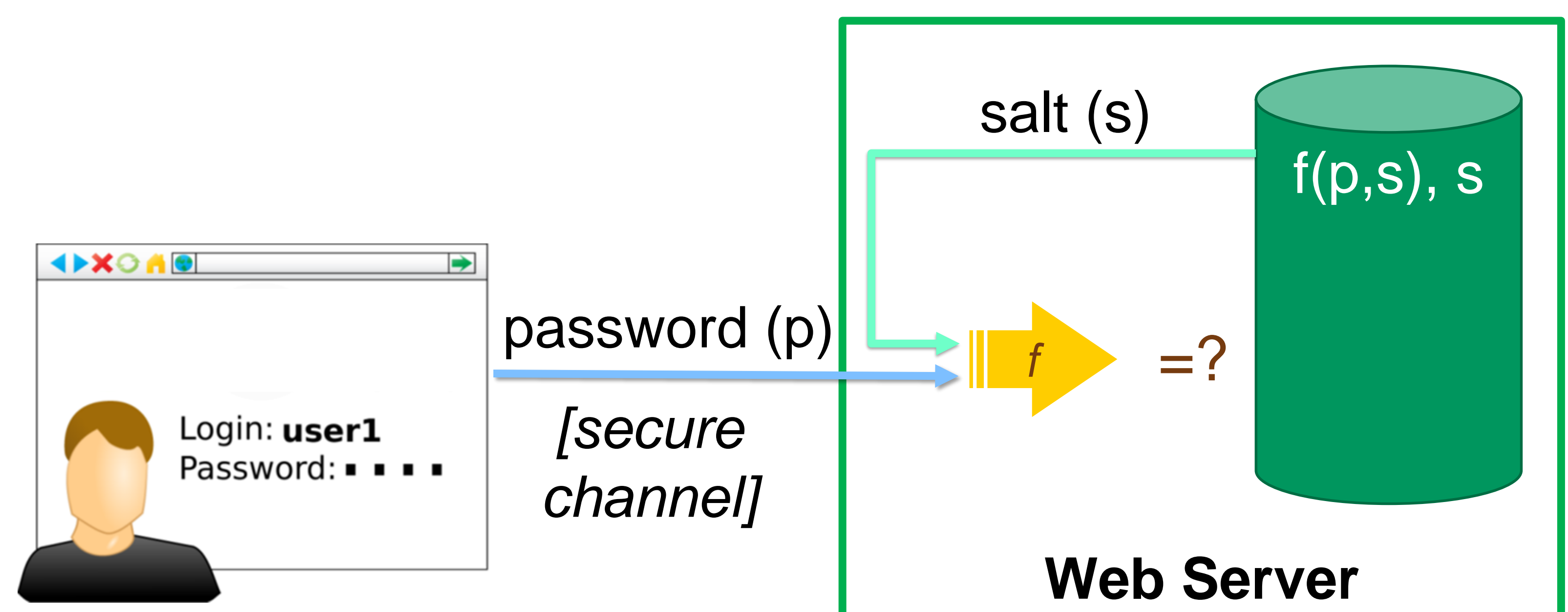
SafeKeeper: Protecting Web Passwords using Trusted Execution Environments



- Passwords are by far the most widely used mechanism to authenticate users on the web.
- Password databases on web servers are therefore **attractive targets for attackers**.
- Our system, **SafeKeeper**, protects web credentials using trusted hardware on web servers and a client-side browser extension.
- SafeKeeper is **deployment-friendly** at the server-side and **verifiable** at the client-side.

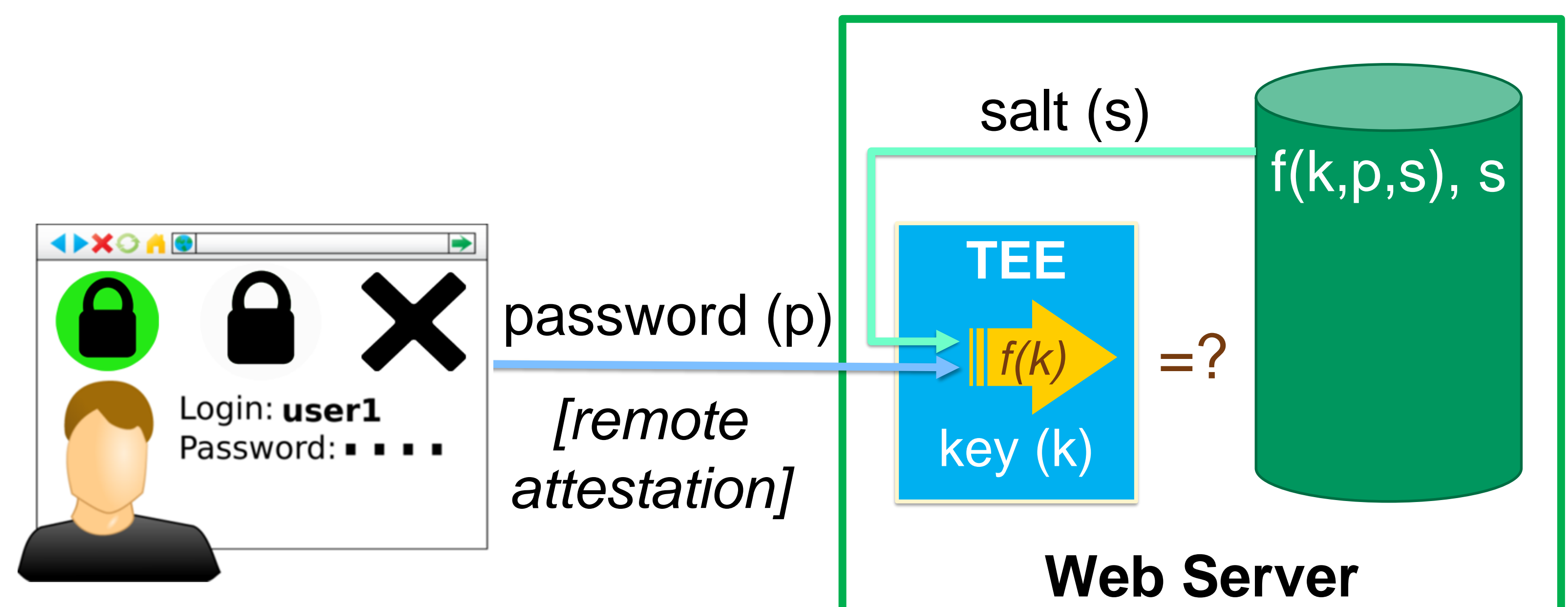
Current password database protection

- Random non-secret **salt** appended to each password to increase difficulty of guessing.
- Web server computes **one-way function** (e.g. cryptographic hash) before storing password.
- Attacker who obtains a database can still perform targeted **offline password guessing**.



Protecting credentials using trusted hardware

- Web server computes a **keyed one-way function** (e.g. CMAC using AES NI).
- Secret key protected within a **trusted execution environment** (TEE).
- Browser extension performs **remote attestation** and informs user about the result.
- Prevents **offline password guessing**.



Server-side password protection service

- Prototype using Intel **Software Guard Extensions (SGX)**, PHPass, and WordPress.
- Performance evaluation (passwords/second):
 - PHPass: 446 (± 10) p/s
 - SafeKeeper PHP: **1653 (± 70) p/s**
 - Enclave only: **101,337 (± 4186) p/s**

Client-side browser extension

- 86-participant on-site user study.
- Participants were shown 25 testing websites; some actively spoofed the SafeKeeper UI.
- Participants were asked to determine if the website protects passwords using SafeKeeper.
- **Average accuracy: 87%**.
- Follow-up study after 2 months without use: accuracy decreased by 2%.
- 94% rated the extension as **“easy to use”**.

ssg.aalto.fi/research/passwords