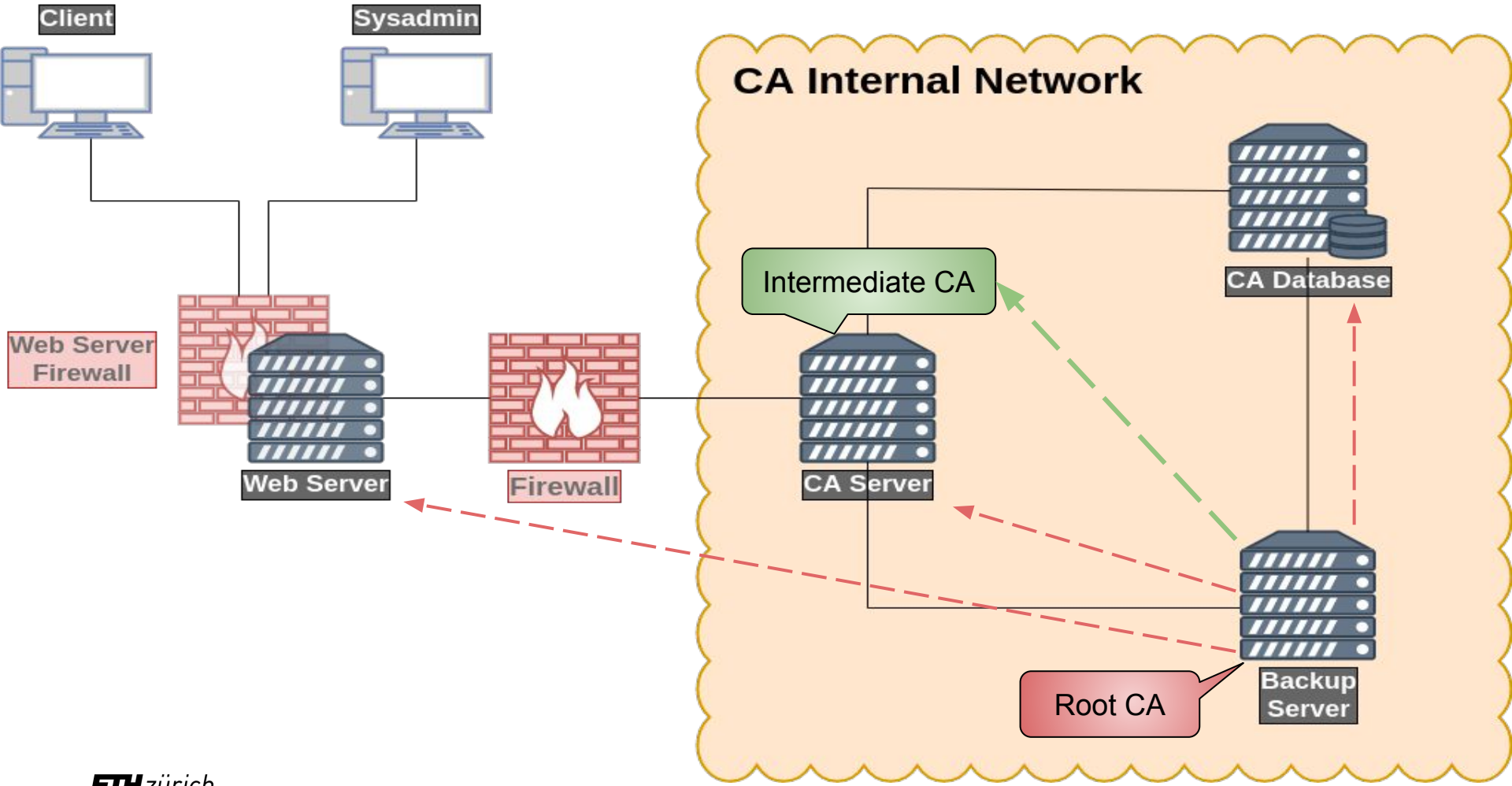
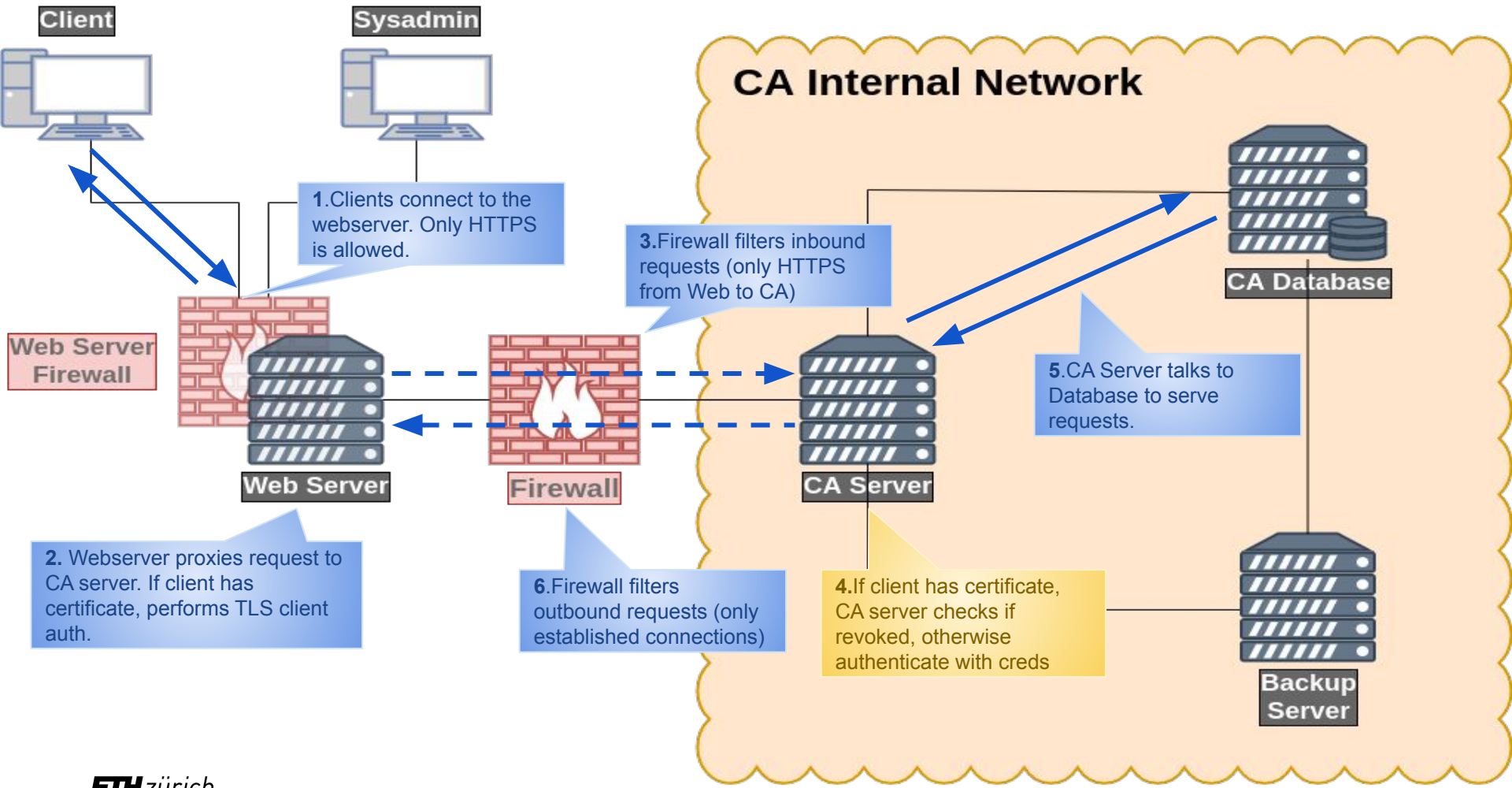
Group 6
System presentation

Lucie Hoffmann,
Francesco Intoci,
Elvric Trombert,
Lucas Rollet

Agenda

1. Architecture Highlights
2. Backup
3. Security Design
4. Backdoors
5. Final remarks


Client
Sysadmin
CA Internal Network
Intermediate CA
CA Database
Web Server Firewall
Web Server
Firewall
CA Server
Backup Server
Root CA


Client
Sysadmin
CA Internal Network
1.Clients connect to the webserver. Only HTTPS is allowed.
3.Firewall filters inbound requests (only HTTPS from Web to CA)
CA Database
Web Server Firewall
5.CA Server talks to Database to serve requests.
Web Server
Firewall
CA Server
2. Webserver proxies request to CA server. If client has certificate, performs TLS client auth.
6.Firewall filters outbound requests (only established connections)
4.If client has certificate, CA server checks if revoked, otherwise authenticate with creds
Backup Server

Backup

- rsyslog
  - *from all machines (webserver, caserver, firewall and database)*
  - *Done over TLS in real time*
  - *Include nginx access and error logs from webserver and caserver*
- database backup every day
  - *using sftp user dbackup*
    - *chrooted*
    - *no ssh login*
- private key backup straight after their generation
  - *encrypted on the caserver then sent to the backup*
  - *using sftp user cabackup*
    - *chrooted*
    - *no ssh login*
- Database and client private key backups are copied to a directory inaccessible to the sftp users. Overwriting of the copied files is prevented

Review of Security Mechanisms - Architecture Level

- **Network:**
  - ***Compartmentalization***: network segmentation
  - ***Separation of Privileges***: each machine has a specific role
  - ***Least Privilege***: machines traffic is restricted for their specified role
  - ***Zero-Trust Model***: no implicit trust between machines

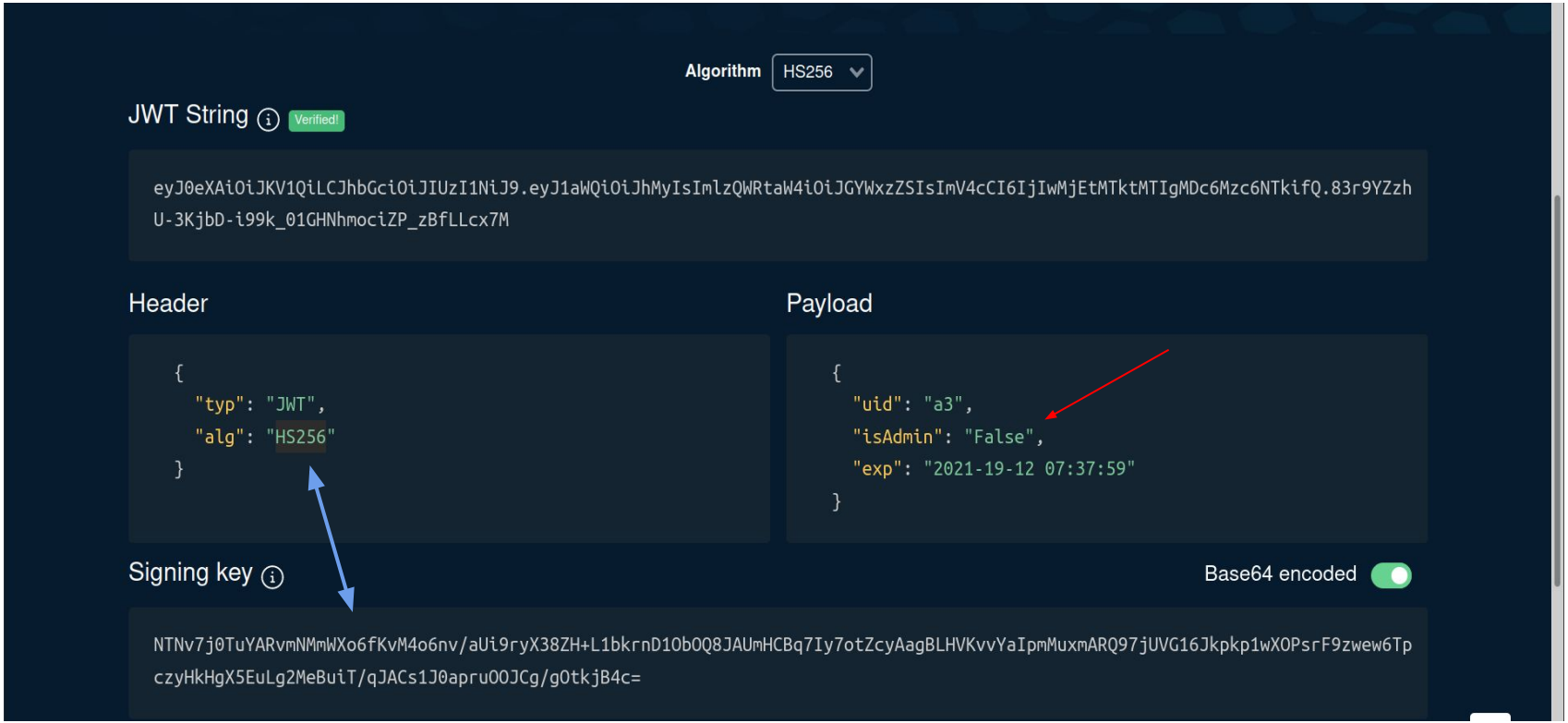
- **Data protection:**
  - ***In transit*****:** encryption everywhere (TLS, SSH)
  - ***At rest*****:** encryption of clients private keys and system configurations

Review of Security Mechanisms - Application Level

- **Protection against common web vulnerabilities:**
  - ***XSS:***
    - React automatically escapes XSS payloads
    - Session cookie with *HTTP Only* flag
  - ***CSRF*:**
    - Session cookie with *Same-Site strict* attribute
    - allow only POST requests for state-changing requests
  - ***SQLi:*** prepared statements
  - **Auth via JWT:** not tamperable (HMAC-SHA256)

- **Availability:** Basic rate limiting for heavy tasks (DoS mitigation)

Easy - JWT “none” attack

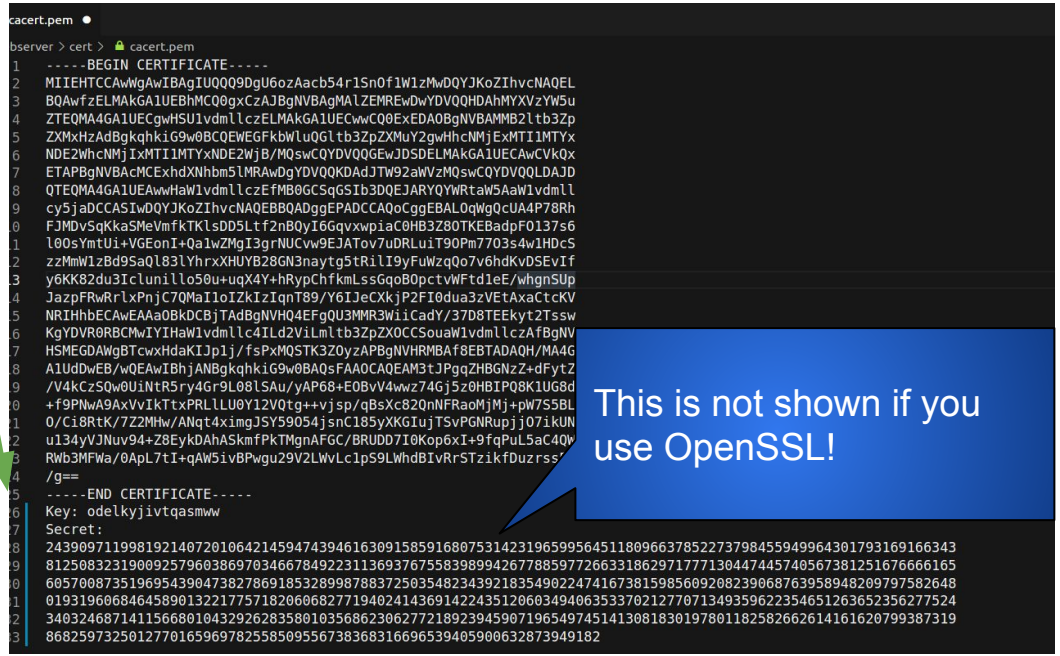


JWT “none” attack - exploit

Hard - Kleptography - discovery

Listed in */robots.txt*

Useful hint ;)



This is not shown if you use OpenSSL!

Kleptography - exploit

- *SETUP:*
  1. Generate RSA parameters as usual (N,e,d)
  2. Generate a small $|\delta| < |N|/4$ which is also invertible mod$\varphi(N)$
  3. Compute $\varepsilon = \text{inv}(\delta, \varphi(N))$
  4. Encrypt ε using AES_CBC_128 with NSA key (this is "secret" in the certificate…)

- *ATTACK:*
  1. Get ε by decrypting the secret with the leaked key.
  2. Apply Wiener's low exponent attack to get δ *(it's an algorithm that allows you to retrieve the secret key if it is "small" enough)*
  3. Now you have a multiple of $\varphi(N)$: $\varepsilon.\delta - 1$
  4. You can now factor N by known algorithms (yes, this is similar to Miller-Rabin test):
     - Divide $\varphi(N)$ by 2 until odd: get *s*
     - Raise a random base to s: get *b*
     - If *b* is not 1, keep squaring it until you get 1
     - If you get 1, then you know that *pow(b,2) - 1 = 0 modN*
     - Simply take *gcd(b+1,N)* to find (hopefully) a non trivial factor of N

Room for improvement

- Weird generation of certificate with bash + OpenSSL may have lead to larger attack surface
  (probably a versioning problem with python cryptography, certificates were not parsed in Browser)
- Probably too much time spent on frontend (we had fun diving into React though)
- Logging via ssh tunnelling
- SSH login using SSH proxy

~$: echo ‘Thanks for the attention’
~$: sudo wish --Merry_Xmas!