Funded by the European Union under Grant Agreement No. 101087529. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or European Research Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.

Co-funded by the European Union

Využití post-kvantové kryptografie v praxi

Brněnské bezpečnostní setkávání

8.11.2023

Petr Muzikant

Cybernetica AS, Estonsko

[Prezentace obsahuje klikatelné odkazy]

CYBERNETICA

Obsah

- **Úvod**
- **Naše zkušenosti s PQ implementací**
- **Aktuální stav**
  - knihovny, ASN.1, JWA, hybridní módy
- **Obecné poznatky**
  - Příprava, technická omezení, implementace
- **Závěr**

Úvod

- **Kvantový PC → ~~RSA, ECDSA, …~~ → PQC → nové algoritmy**
- **Standardizace PQC** (např. NIST)
- **Další krok**:
  - Podpora PQ ve všech vrstvách systémů
  - Zachování funkcionality, kompatibility a interoperability
- **Naše aktivity**:
  - Zkoumání dostupných řešení a state-of-the-art
  - Zaměření na inženýrský aspekt PQ implementace
  - Sbírat zkušenosti, poznámky, tipy, atd…

Post-kvantové algoritmy

- NIST standardizační proces (2016-dnes)
  - "Round 4":
    - **Digitální podpis:**
      - Dilithium, Falcon (lattice-based)
      - Sphincs+ (hash-based)
    - **Asymetrické ustanovení klíče:**
      - Kyber (lattice-based)
  - Další algoritmy: TBA ("On-ramp")
- Další evaluační snahy (BSI, ENISA, ...) → pravděpodobně další algo.

Naše zkušenosti s PQ implementací

PQ autentizační framework, CDOC2, ASICE, IVXV, TOPCOAT

Projekty s přímou implementací

- **PQ-Web-eID**
  - autentizační framework pro webové aplikace
  - estonské el. občanské průkazy + státní služby
  - dig. podpisy, ~~čipová karta~~ → ESP32 programovatelná deska
- **PQ-CDOC2**
  - estonský standard pro zabezpečení a výměnu dat
  - KEMs, problém s TLS
- **PQ-ASiC-E**
  - (téměř estonský) standard pro dig. podepsaný kontejner dat

Projekty s problematickou implementací

- **PQ-IVXV**
  - el. volební systém
  - příprava infrastruktury, PQ-OCSP, PQ-TSA
  - dig. podpisy OK, šifrování hlasu problém
    - ~~elGamal~~ → kompletně nový PQ protokol (lattice-based)
- **TOPCOAT**
  - skupinový dig. podpis
  - téměř žádné existující implementace → kompletně nový PQ protokol (lattice-based)

PQ knihovny

- PQClean (C)
  - *Čistá* agregace "round 4" algoritmů pod jednotným API
  - Zdroj zdrojových kódů
- libOQS (C)
  - + rozšíření do C++, Python, Java, Go, .NET, Rust
  - + aplikace s libOQS (OQS-OpenSSL, OQS-OpenSSH, OQS-OpenVPN)
- BouncyCastle (Java), rustpq/pqcrypto (Rust), pqm4 (C, Cortex-M4), botan-pq (C++)
- vlastní wrappery

PQ ASN.1 notace

- **Žádné standardy neexistují** (NIST vyžaduje *raw bytes*)
- RFC drafty
  - definice privátních a veřejných klíčů + jejich atributy
  - např. `DilithiumPrivateKey` obsahuje `nonce`, `tr`, `s1`, `s2`, `t0`, …
- **PQ Object Identifiers (OIDs)**:
  - OQS definovalo své vlastní, BouncyCastle rozšířilo

PQ JSON Web Algorithms (RFC 7518)

- Využití např. v: *JW Signature*
- Formát: *(DIGSIG + HASH)*
- Příklad: *ES384 = "ECDSA using P-384 curve and SHA-384"*
- **Žádné RFC drafty ani standardy pro PQ JWAs**
- RFC draft pro **JSON Web Encodings**
  - *např. CRYDI5 = CRYSTALS-Dilithium parameter set 5*
- HASH? → SHA-512

Hybridní módy (PQ + klasická kryptografie)

- Dlouhá životnost důvěrnosti dat + ochrana před novými útoky na PQC
- **Zřetězení** (concatenation) nebo **sekvenční zpracování**
  - *Ghinea et al.*
    - obojí může mít bezpečnostní vady
    - nová metoda ke zvýšení nefalšovatelnosti ECDSA+PQ podpisů
- RFC draft pro hybridní **KEM v TLS1.3** využívá **zřetězení**
- **Cloudflare a Google Chrome** již dnes následují RFC draft využívající **zřetězení** (X25519 + Kyber)

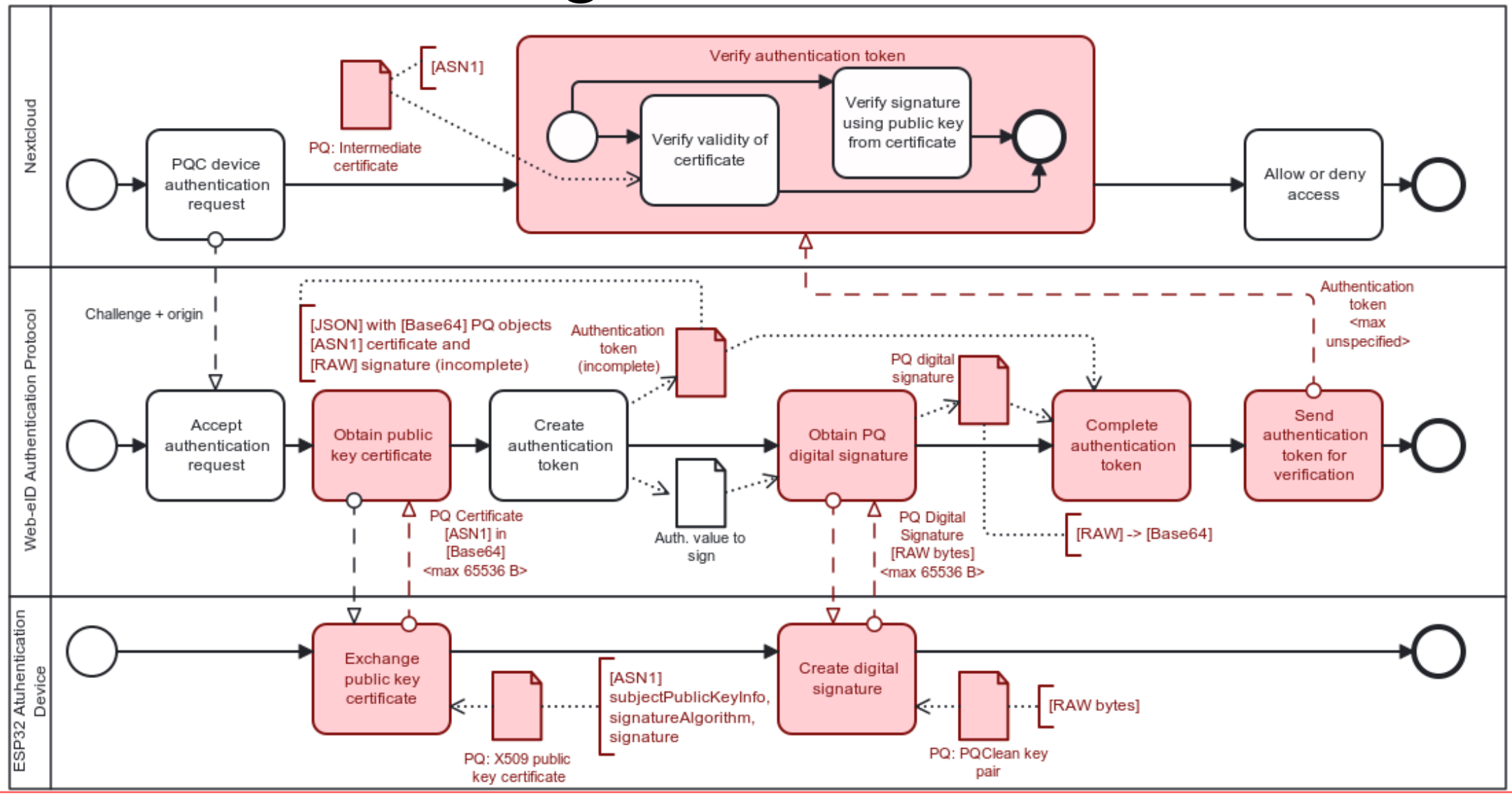
Obecné poznatky

Příprava, technická omezení, implementace

Příprava

- **Identifikace všech PKI objektů a jejich životnosti**
  - rozsah potřebných změn
- **Zaznamenat maximální limity přenosu dat**
  - větší objekty, proměnlivé velikosti (Falcon)
- **Zaznamenat měnící se formáty dat**
  - ASN1, Base64, PEM, JOSE, other...

Příklad: BPMN diagram



CYBERNETICA

Technická omezení

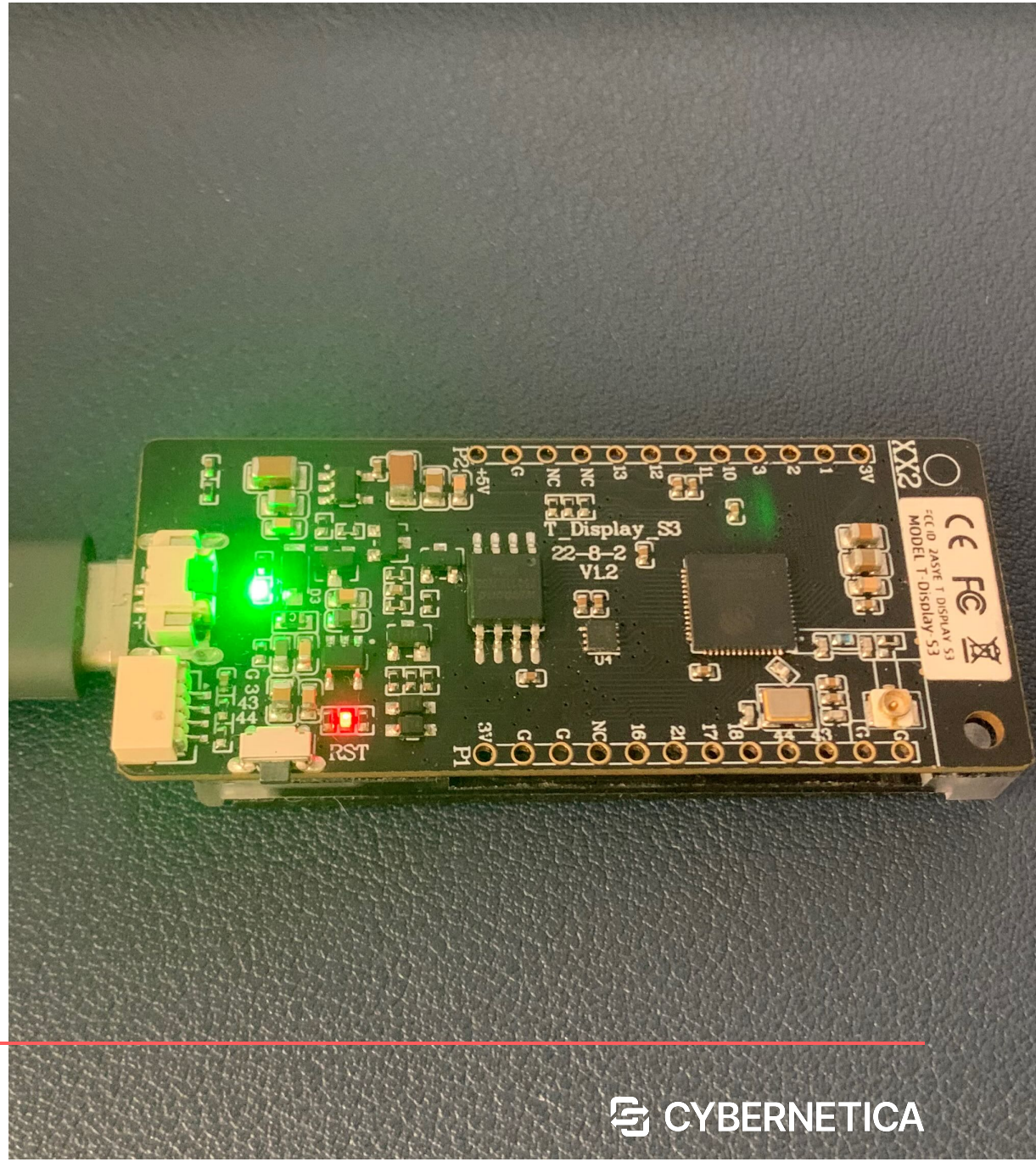
- Klást důraz na současné hranice systému
  - zvýšené nároky na **výkon**, **paměť**, a **úložiště**
  - limitovaná zařízení a pomalé sítě
- **Úpravy protokolu**:
  - streamování veřejných klíčů a podpisů do paměti
  - ustanovení klíče namísto dig. podpisů (kreditní karty)
  - alokace objektů v jiném typu paměti (*heap* namísto *stack*)

Implementace

- Následovat životní cyklus dat od začátku do konce
  - implementace PQ **krok po kroku**
- Implementace stále **není přímočará**
- **Rozšíření, úpravy, adaptace** existujících knihoven
- **Tvorba vlastního *wrapperu* pomocí** SWIG
- Očekávejte změny - standardizace není ukončená!

Embedovaná zařízení

- Čipové karty nejsou vhodné → LilyGO T-Display-S3
- Problematický memory management:
  - Omezení na 8KB *stack* paměti
  - *PQClean* alokuje pouze do *stack* paměti
- Řešení: úprava *PQClean* kódu
  - funkce `malloc` a `free`
  - `std::unique_ptr` (C++ v11)



libOQS rozšíření

- Dostupná rozšíření pro C++, Python, Java, Go, .NET, and Rust
  - ~~PHP~~? → SWIG
- Definice C/C++ rozhraní
  - → *liboqs-php*, *liboqs-python*, *liboqs-go*
- Mapování datových typů:
  - PHP `string` ↔ C++ `uint8_t*`
  - Python `bytes` ↔ C++ `uint8_t*`

PQ v PHP

- **OpenSSL → OQS-OpenSSL**
  - (v1 ukončena podpora)
  - v3 *extension provider*:
    - rozšiřuje standardní OpenSSLv3
    - ID algoritmů v PHP jsou pevně dané pro DSA, DH, RSA a ECDSA
      - některé funkce však algID nevyžadují (např. `openssl_verify()`)
  - více poznatků na OQS-OpenSSL v PHP
- **PHPSecLib → nový PQC-PHPSecLib fork**
  - *OQS-OpenSSL* nebo *liboqs-php* (podle dostupnosti)

PQ v BouncyCastle (v1.74+)

- Ne příliš dobrá dokumentace
  - bc-java / core / src / main / java / org / bouncycastle / asn1 / bc / BCObjectIdentifiers.java
  - `org.bouncycastle.pqc.*` balíčky
- Pracuje s ASN.1 objekty → spíše komplikace
  - (oproti *raw bytes* v libOQS)
  - např. `KyberPublicKeyParameters` has `t` and `rho`

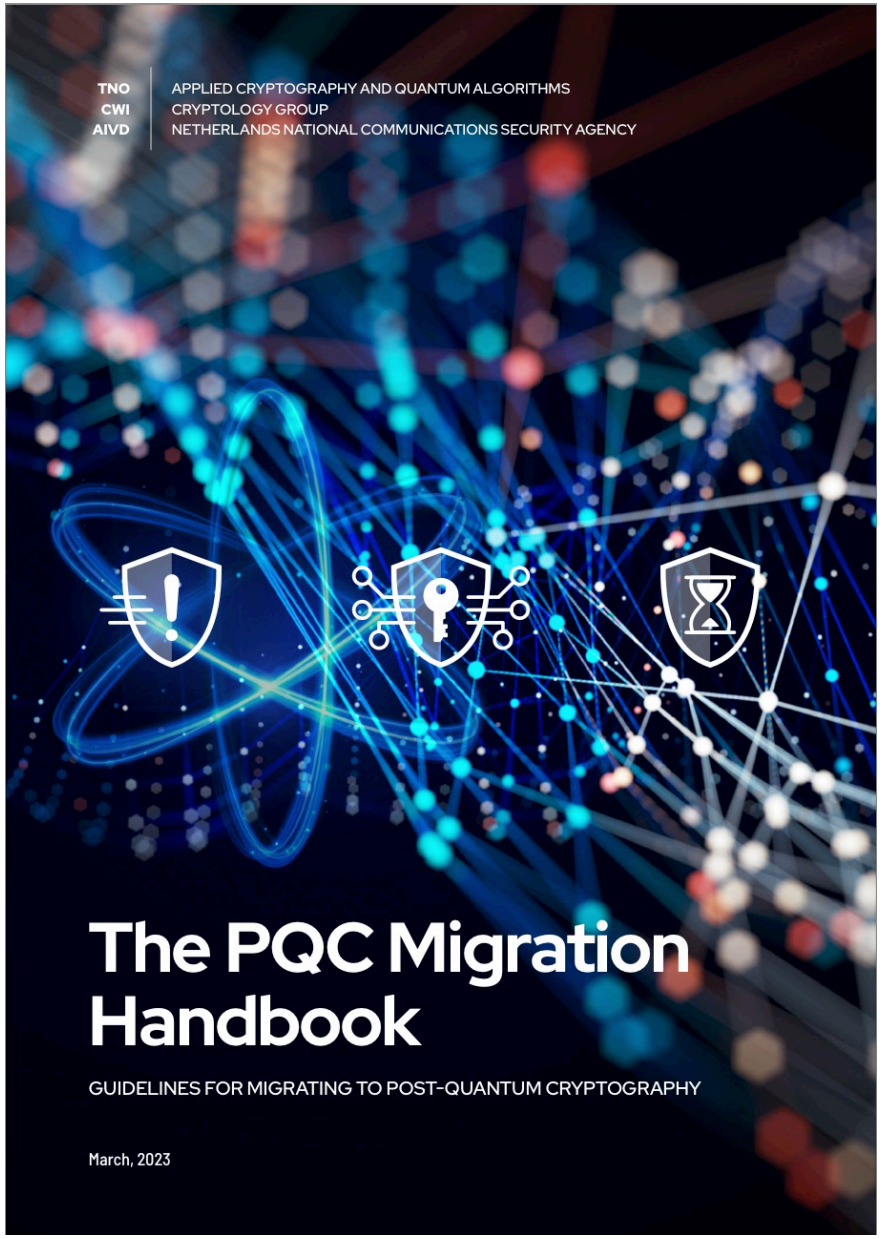
PQ Java Keytool

- *keytool* = správa tzv. *keystore* obsahující krypto. objekty
- PQ BouncyCastle → PQ Java Keytool
- např. generace *.p12* s Dilithium klíčovým párem a self-signed cert.:

```
keytool \
    -providerpath bcprov-jdk18on-175.jar \
    -provider org.bouncycastle.pqc.jcajce.provider.BouncyCastlePQCProvider \
    -genkeypair \
    -keyalg Dilithium5 \
    -alias cdoc20-client-pqc-CA \
    -keystore cdoc20clientpqcCA.p12 \
    -storepass passwd \
    -sigalg Dilithium5 \
    -dname "CN=cdoc20-client-pqc-CA,OU=ISRI,O=CyberneticaAS,L=Brno,S=Czechia,C=CZ"
```

Poslední 2 dny

- **Post-Quantum Cryptography conference 2023** by PKI Consortium
  - záznamy na YT
  - PQC Migration Handbook
  - experimenty na embedded zařízeních jsou až moc nerealistické
  - FIDO2 tokeny vysoce omezené
  - PQC na mobilech bude vyžadovat ko-procesory
  - bankovnictví je XY let pozadu



CYBERNETICA

Závěr

- **Implementace PQC již dnes je...**
  - **...komplikovaná**:
    - několik knihoven = několik přístupů k PQ a rozdílné dokumentace
    - výpočetní nároky, adaptace, úpravy
    - standardizace není u konce (a ani kompletní - MPC, ZKP, etc..)
  - **...proveditelná:**
    - autentizační framework, kryptosystémy pro šifrování, dig. podpisy, atd.
  - **...užitečná:**
    - dlouhodobá ochrana dat, zkušenosti
  - **...nápomocná:**
    - velký prostor pro open-source aktivitu, podpora globálního převodu na PQC

Děkuji za pozornost

Zdroje:
- odkazy v prezentaci
- PQ autentizační framework
- Poznámky k PQC v PHP
- napište mi mail

Petr Muzikant, petr.muzikant@cyber.ee

https://cyber.ee/
info@cyber.ee
cybernetica
CyberneticaAS
cybernetica_ee
Cybernetica

CYBERNETICA