On the Implementation of X.509-Compliant Quantum-Safe Hybrid Certificates

```
Certificate:
  Data:
    Version: 3 (0x2)
    Serial Number:
      4a:24:de:52:e3:9f:ef:f9:e7:12:fe:6e:77:1d:e4:f0:ae:86:fb:64
    Signature Algorithm: p521_dilithium5
    Issuer: C = AU, ST = Some-State, O = RootCA
    Validity
      Not Before: Jun  8 18:46:55 2023 GMT
      Not After: Jun  8 18:46:55 2024 GMT
    Subject: C = GR, ST = Thessaloniki, O = RootCA
    Subject Public Key Info:
     Public Key Algorithm: p521_dilithium5
        Public-Key: (521 bit)
        pub:
        04:00:65:8f:87:b9:fe:15:a5:25:f4:f5:e5:66:56:7b:d3:14:43:27:a6:9f:73:ae:6e:00:53:5e:e0:2f:
        04:03:37:af:0d:e0:a7:2e:64:fa:0e:81:4e:33:ff:b0:37:6f:07:97:07:5e:7e:55:2e:80:47:19:e6:69:
                          ... omitted for brevity ...
        0f:03:19:10:73:4d:79:4b:fb:1a:84:23:7a
        ASN1 OID: secp521r1
        NIST CURVE: P-521
        dilithium5 Public-Key:
        pub:
        4f:a8:b9:c7:c9:fa:df:e4:26:0e:77:c6:44:37:94:5b:8e:6e:04:bd:7d:a0:50:40:27:39:ab:07:8e:95:
        fb:37:65:fe:8a:be:d3:52:00:0a:69:62:ee:35:31:ef:3b:55:f4:65:6a:d8:eb:b0:73:49:10:7d:ed:e3:
        9b:c7:5b:26:2b:bd:76:4f:93:14:0f:a0:16:2a:30:39:8b:99:fb:b9:7d:a3:46:cb:b8:ac:06:78:b6:d4:
                          ... omitted for brevity ...
        d7:51:19:6a:be:9e:ee:37:27:9e:62:77:ea:26:ee:13:38:2b:2b:e6:3e:e2:06:47:d7:01:f6:ed:95:0b:
        f1:00:fb:c6:d0:8f:1a:d3:ef:ea:88:9e
    Signature Algorithm: p521_dilithium5
    00:00:00:8a:30:81:87:02:41:51:6b:60:d6:05:0e:9b:68:f1:5d:eb:d0:28:12:3b:70:65:af:76:c6:69:
    f5:fd:e7:d3:d9:60:d9:9f:5e:89:d5:a7:0f:1e:dd:b7:f8:b6:5d:7b:ef:67:1a:bb:ee:07:a0:3e:45:57:aa:
                          ... omitted for brevity ...
    21:42:3a:38:62:55:bd:dd:d2:9b:65:93:66:3e:0d:f2:b6:04:94:df:d2:55:18:83:e8:e9:9d:94:6d:0a:
    9d:38:c6:ee:4e:2f:fe:44:27:e6:ed:41:db:42:3e:62:5f:ee:dd:1f:94:94:62:d2:2d:36:e3:49:dd:de:8a:
    d4:30:1b:50:31:e2:a1:8f:4f:b4:6d:e8:0d:5e:0e:dd:31:a5:a4:95:ba:81:ef:35:16:b4:6f:f9:d5:4e:86:
                          ... omitted for brevity ...
    f7:00:35:5d:89:d1:d2:06:0e:13:ae:e6:0d:28:2d:74:96:b3:24:7f:95:b6:d8:00:0e:62:6b:79:8b:96:
    e7:e8:2d:36:60:f0:00:00:00:00:00:00:00:00:00:00:00:00:00:0e:1b:21:26:2e:31:3a:3e
```

Agenda

- Introduction
- Application in the Defense Sector
- Quantum-Safe Hybrid Certificates
- Proof-of-Concept (PoCt)
  - Overview of the Certificate Chain of Trust
  - Implementation
  - Hybrid Certificate Structure
- Conclusions

Introduction

Introduction

- The emergence of **quantum computing** has introduced a new **dimension** of **security challenges**. Traditional cryptographic algorithms, are expected soon to be vulnerable **due to increasing quantum computing capacity**.

- Therefore, the adoption of quantum-safe algorithms has become of **paramount importance**, especially in the areas of **defense and security sectors**.

NITECH: NATO Innovation and Technology – Issue 7, Published on Jul 5, 2022

Introduction

- **Quantum-safe algorithms** provide resistance against attacks from both classical and quantum computers, **ensuring** the **long-term security** of sensitive data and communications.

- By incorporating quantum-safe algorithms into the Public Key Infrastructure (PKI), security and defense sectors can **mitigate the risks posed by quantum computing** and maintain the confidentiality and integrity of critical information.

Introduction

- **Hybrid X509 certificates** containing both traditional and post-quantum algorithms are a promising solution to address the threat posed by quantum technology to traditional public key cryptography.

- Ensuring **backward compatibility** with traditional cryptography systems is key, allowing a **smooth transition** period to post-quantum systems.

Introduction

High-level Implementation Challenges

- **Standardization and interoperability** between different hybrid certificate systems, which can limit their wide application.
- **Increased overhead** of post-quantum algorithms which can impact the performance of public key infrastructure

Introduction

Recent progress…

- NIST **multi-year evaluation process**
- **3 new quantum-safe algorithms** are in preparation to be used in 2024
  - CRYSTALS Kyber (initial public draft FIPS 203) [general encryption]
  - CRYSTALS Dilithium (initial public draft FIPS 204) [digital signatures]
  - SPHNICS+ (initial public draft FIPS 205) [digital signatures]
  - **+ 1 pipelined** (FALCON) [digital signatures]
- A **second set** of algorithms with alternative defense methods is expected to be evaluated in case the 1st set show weaknesses

Introduction

This presentation…

- Demonstrate a **proof of concept (PoCt) implementation** based on a fork of the openssl project, utilizing the open quantum safe (oqs) library;
- focuses on certificates for **digital signatures**, combining the traditional algorithm **ECDSA** and the post quantum algorithm **Crystals-Dilithium**;
- **outlines the essential steps for generating**, **distributing**, and **verifying** hybrid X509 certificates through a trust chain consisting of a Root Certification Authority (CA) and an Intermediate CA.

Application in the Defense Sector

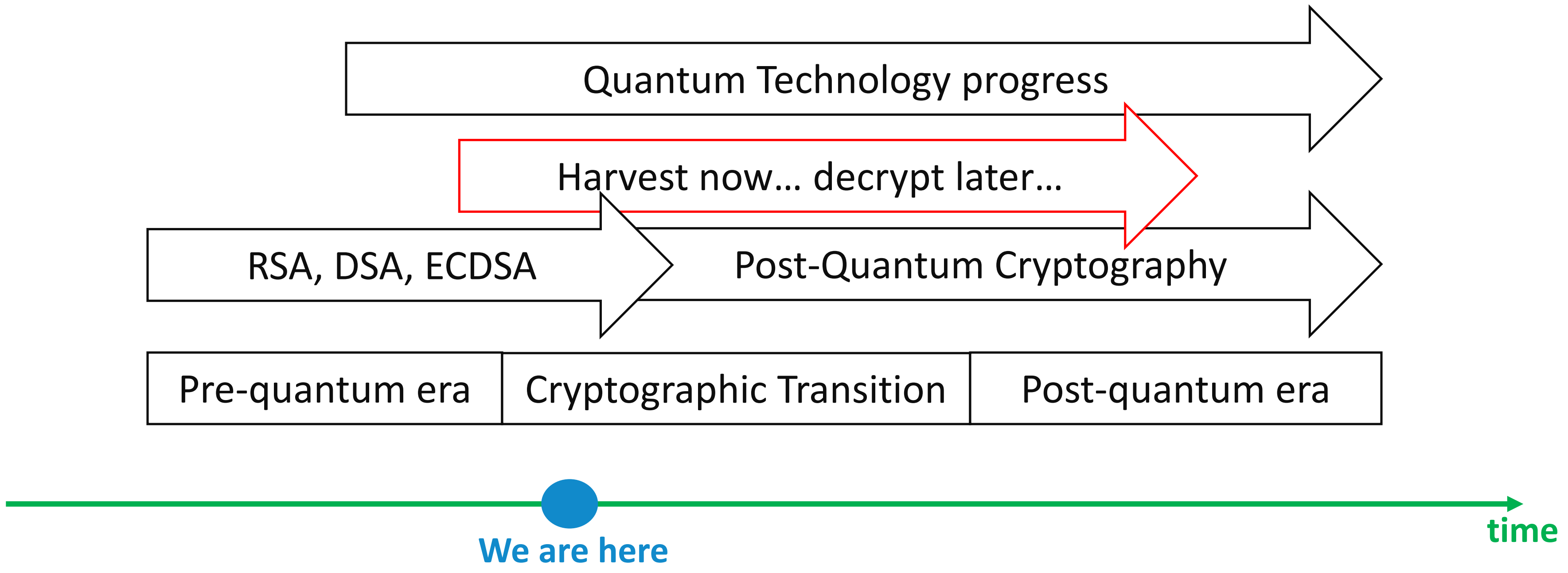
Application in the Defense Sector

- Hybrid X509 certificates enhance Authenticity, Integrity and Non-repudiation
- PKI spans across a wide area of modern communication systems
- NATO PKI is one example, underpinning Multi Domain Operations (MDO)
  - Some application areas:
  - Network communication security
  - Unmanned Aerial Vehicles (UAVs)
  - Submarine operations
  - …

https://www.jwc.nato.int/application/files/1516/3281/0425/issue37_21.pdf

Application in the Defense Sector

- Mitigates or eliminates the risk of “Harvest now….decrypt later…”



Quantum-Safe Hybrid Certificates

Quantum-Safe Hybrid Certificates

- **digital signature** algorithms such as RSA, DSA, ECDSA were proven to be more than secure, but security concerns begun to arise when assumptions were made that attackers own quantum computers;

- Pre-Quantum -> Post-Quantum: During this **transition**, it is important to **ensure a seamless cryptographic security**, maintaining current levels of security and functionalities.

Quantum-Safe Hybrid Certificates

- Instead of totally replacing current algorithms with arguably less-studied and less-supported post-quantum ones, the scientific community came up with 2 approaches:
  - combining a traditional and a post-quantum algorithm into the same X509 fields by concatenating the shared secrets;
  - extending the schema of the present certificate structure with additional extensions;

- both, fully compatible with the latest recommendations of ITU-T.

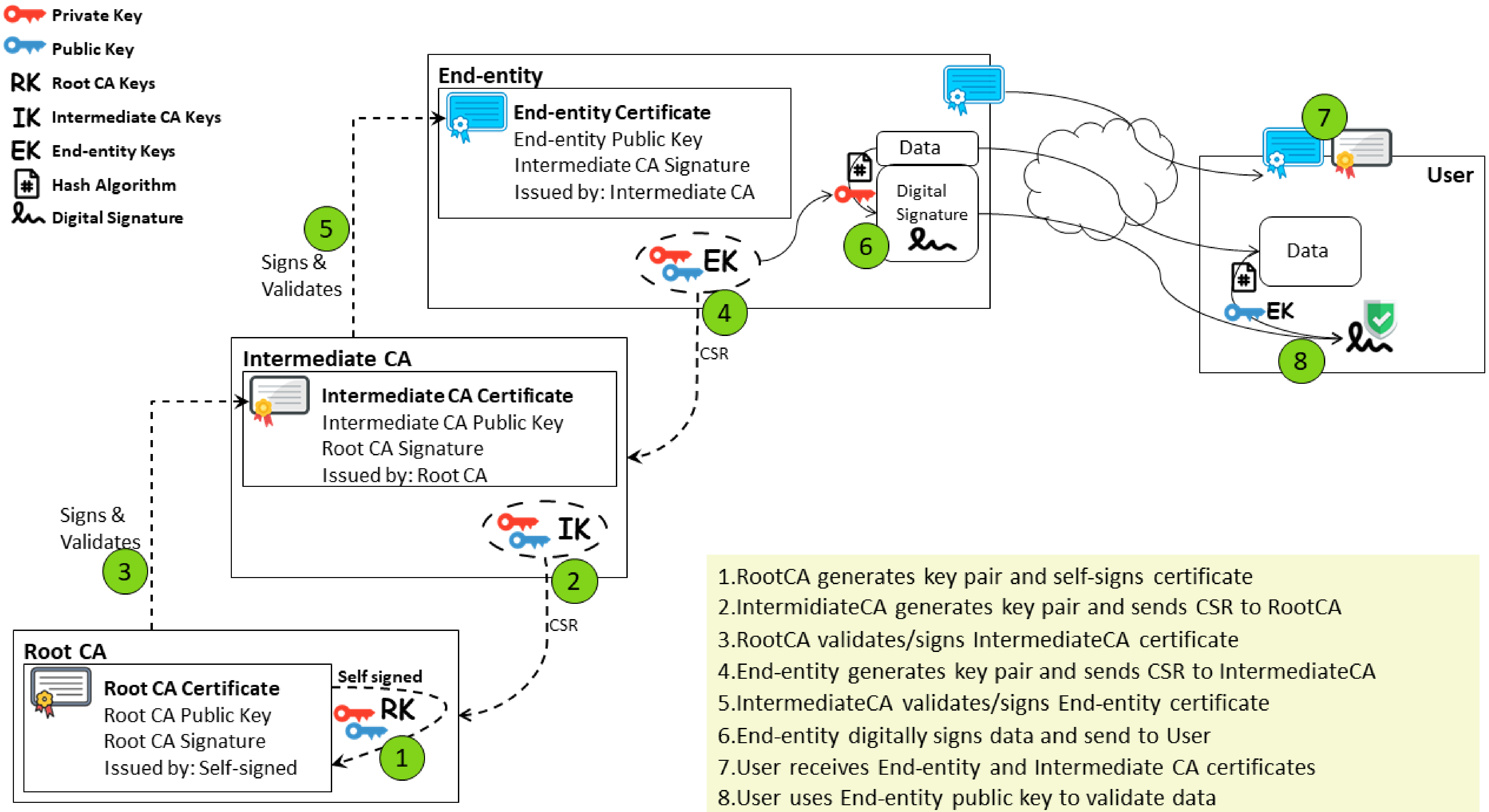
Proof-of-Concept (PoCt)

Proof-of-Concept (PoCt)

Objectives…

- show that it is no “rocket-science” (…any more);
- demonstrate how easy it is to implement;
- highlight that open source community is heavily contributing;
  - Industry partners are developing similar service wrappings based on open source solutions

Proof-of-Concept (PoCt)



1.RootCA generates key pair and self-signs certificate
2.IntermidiateCA generates key pair and sends CSR to RootCA
3.RootCA validates/signs IntermediateCA certificate
4.End-entity generates key pair and sends CSR to IntermediateCA
5.IntermediateCA validates/signs End-entity certificate
6.End-entity digitally signs data and send to User
7.User receives End-entity and Intermediate CA certificates
8.User uses End-entity public key to validate data

Proof-of-Concept (PoCt)

Implementing…

- The PoCt was created on a virtualized environment
  - hosted on a VMware ESXi 7.0 U2 hypervisor
  - based on Ubuntu Linux (64-bit)
  - 8 CPUs and 12 GB of memory
- Our implementation combines the traditional **ECDSA** algorithm over the **P-521 elliptic curve**, using **SHA-512** as the hash function, along with the quantum-safe algorithm **Crystals Dilithium5**.

Proof-of-Concept (PoCt)

NIST Security Strength Categories

Category	Description
	Any attack that breaks the relevant security definition must require computational resources comparable to or greater than those required…
1	…for key search on a block cipher with a 128-bit key (**e.g. AES128**)
2	…for collision search on a 256-bit hash function (**e.g. SHA256/ SHA3-256**)
3	…for key search on a block cipher with a 192-bit key (**e.g. AES192**)
4	…for collision search on a 384-bit hash function (**e.g. SHA384/ SHA3-384**)
5	…for key search on a block cipher with a 256-bit key (**e.g. AES 256**)

Proof-of-Concept (PoCt)

Implementing…

- Create RootCA (*offline process)
  - **Create a RootCA Private Key and a self signed RootCA Certificate**

    apps/openssl req -x509 -newkey p521_dilithium5 -keyout sto/RootCAkey_ecdsa_dil5.pem -out sto/RootCAcert_ecdsa_dil5.pem -config apps/openssl.cnf

Proof-of-Concept (PoCt)

Implementing…

- Create IntermediateCA
  - **Create IntermediateCA Private Key and IntermediateCA Certificate Signing Request (CSR)**

    apps/openssl req -newkey p521_dilithium5 -keyout sto/IntermediateCAkey_ecdsa_dil5.pem -out sto/IntermediateCAcsr_ecdsa_dil5.csr -config apps/openssl.cnf

  - **Create IntermediateCA Certificate by signing it with RootCA Certificate**

    apps/openssl x509 -req -in sto/IntermediateCAcsr_ecdsa_dil5.csr -CA sto/RootCAcert_ecdsa_dil5.pem -CAkey sto/RootCAkey_ecdsa_dil5.pem -out sto/IntermediateCAcert_ecdsa_dil5.pem -extfile sto/ca_intermediate.ext -extensions v3_intermediate_ca

  - **Verify IntermediateCA Certificate against RootCA Certificate**

    apps/openssl verify -CAfile sto/RootCAcert_ecdsa_dil5.pem sto/IntermediateCAcert_ecdsa_dil5.pem

Proof-of-Concept (PoCt)

Implementing…

- Create User Certificate
  - **Create User Key Pairs and CSR**

    apps/openssl req -newkey p521_dilithium5 -keyout sto/User1key_ecdsa_dil5.pem -out sto/User1csr_ecdsa_dil5.csr -config apps/openssl.cnf

  - **Create User Certificate by signing it with IntermediateCA Certificate**

    apps/openssl x509 -req -in sto/User1csr_ecdsa_dil5.csr -CA sto/IntermediateCAcert_ecdsa_dil5.pem -CAkey sto/IntermediateCAkey_ecdsa_dil5.pem -out sto/User1cert_ecdsa_dil5.pem

  - **Verify User Certificate against IntermediateCA Certificate**

    apps/openssl verify -CAfile sto/RootCAcert_ecdsa_dil5.pem -untrusted sto/IntermediateCAcert_ecdsa_dil5.pem sto/User1cert_ecdsa_dil5.pem

Proof-of-Concept (PoCt)

Hybrid Certificate Structure

```
Certificate ::= SEQUENCE {
        tbsCertificate          TBSCertificate,
        signatureAlgorithm      AlgorithmIdentifier,
        signatureValue          BIT STRING }

TBSCertificate ::= SEQUENCE {
        version                 Version,
        serialNumber            CertificateSerialNumber,
        signature               AlgorithmIdentifier,
        issuer                  Name,
        validity                Validity,
        subjectPublicKeyInfo    SubjectPublicKeyInfo,}

SubjectPublicKeyInfo ::= SEQUENCE {
        algorithm               AlgorithmIdentifier,
        subjectPublicKey        BIT STRING }

AlgorithmIdentifier ::= SEQUENCE {
        algorithm               OBJECT IDENTIFIER,
        parameters              ANY DEFINED BY algorithm OPTIONAL
}

Version ::= INTEGER { v1(0), v2(1), v3(2) }
```

- public key value is the concatenated value of the ECDSA with the respected value of Dilithium5: subjectPublicKey=Pub(**ECDSA_p521**) || Pub(**Dilithium5**)
- the signature value is the concatenated value of the aforementioned algorithms, thus: signatureValue=Sig(**ECDSA_p521**) || Sig(**Dilithium5**)

Proof-of-Concept (PoCt)

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            4a:24:de:52:e3:9f:cf:f9:e7:12:fc:6c:77:1d:c4:f0:ac:86:fb:64
        Signature Algorithm: p521_dilithium5
        Issuer: C = AU, ST = Some-State, O = RootCA
        Validity
            Not Before: Jun  8 18:46:55 2023 GMT
            Not After : Jun  8 18:46:55 2024 GMT
        Subject: C = GR, ST = Thessaloniki, O = RootCA
        Subject Public Key Info:
          Public Key Algorithm: p521_dilithium5
                Public-Key: (521 bit)
                pub:
                04:00:65:8f:87:b9:fc:15:a5:25:f4:f5:e5:66:56:7b:d3:14:43:27:a6:9f:73:ae:6c:00:53:5c:c0:2f:   2 Bytes ASN1 prefix
                04:03:37:af:0d:e0:a7:2e:64:fa:0c:81:4e:33:ff:b0:37:6f:07:97:07:5e:7c:55:2c:80:47:19:e6:69:   131 Bytes ECDSA Public Key
                                    … omitted for brevity …
                0f:03:19:10:73:4d:79:4b:fb:1a:84:23:7a
                ASN1 OID: secp521r1
                NIST CURVE: P-521
                dilithium5 Public-Key:
                pub:
                4f:a8:b9:c7:c9:fa:df:e4:26:0e:77:c6:44:37:94:5b:8c:6c:04:bd:7d:a0:50:40:27:39:ab:07:8c:95:
                fb:37:65:fc:8a:be:d3:52:00:0a:69:62:cc:35:31:cf:3b:55:f4:65:6a:d8:eb:b0:73:49:10:7d:ed:c3:
                9b:c7:5b:26:2b:bd:76:4f:93:14:0f:a0:16:2a:30:39:8b:99:fb:b9:7d:a3:46:cb:b8:ac:06:78:b6:d4: 2592 Bytes Dilithium5 Public Key
                                    … omitted for brevity …
                d7:51:19:6a:bc:9c:ee:37:27:9c:62:77:ca:26:ec:13:38:2b:2b:e6:3c:c2:06:47:d7:01:f6:ed:95:0b:
                f1:00:fb:c6:d0:8f:1a:d3:ef:ea:88:9e
```

Concatenated Public Keys TLV Separated

Proof-of-Concept (PoCt)

Signature Algorithm: p521_dilithium5

```
00:00:00:8a:30:81:87:02:41:51:6b:60:d6:05:0c:9b:68:f1:5d:eb:d0:28:12:3b:70:65:af:76:c6:69:   10 Bytes ASN1 prefix
f5:fd:c7:d3:d9:60:d9:9f:5c:89:d5:a7:0f:1c:dd:b7:f8:b6:5d:7b:cf:67:1a:bb: cc:07:a0:3e:45:57:aa:   132 Bytes ECDSA Signature
... omitted for brevity ...
21:42:3a:38:62:55:bd:dd:d2:9b:65:93:66:3e:0d:f2:b6:04:94:df:d2:55:18:83:c8:e9:9d:94:6d:0a:
9d:38:c6:ee:4e:2f:fe:44:27:e6:cd:41:db:42:3c:62:5f:ce:dd:1f:94:94:62:d2:2d:36:c3:49:dd:de:8a:
d4:30:1b:50:31:c2:a1:8f:4f:b4:6d:e8:0d:5c:0c:dd:31:a5:a4:95:ba:81:ef:35:16:b4:6f:f9:d5:4c:86:   4595 Bytes Dilithium5 Signature
... omitted for brevity ...
f7:00:35:5d:89:d1:d2:06:0e:13:ae:c6:0d:28:2d:74:96:b3:24:7f:95:b6:d8:00:0e:62:6b:79:8b:96:
e7:e8:2d:36:60:f0:00:00:00:00:00:00:00:00:00:00:00:00:00:0c:1b:21:26:2c:31:3a:3e
```

Concatenated Signatures TLV Separated

Conclusions

Conclusions

- Quantum technology emerges, resulting in the progressive obsolescence of traditional cryptographic algorithms
- Latest progress shows that the development and standardization of quantum-safe algorithms is under a good momentum
- In this presentation, we demonstrated how to create a basic PoCt for a hybrid PKI compatible with traditional and quantum-safe cryptographic algorithms
- Such PKI can underpin multi domain operations, future-proofing authenticity, integrity and non-repudiation in the modern warfare

Contacts

E-mail Dimitrios.Chatziamanetoglou@ncia.nato.int
diehatz@cs.ihu.gr

Phone +32 65 44 1525, +32 475 55 1479