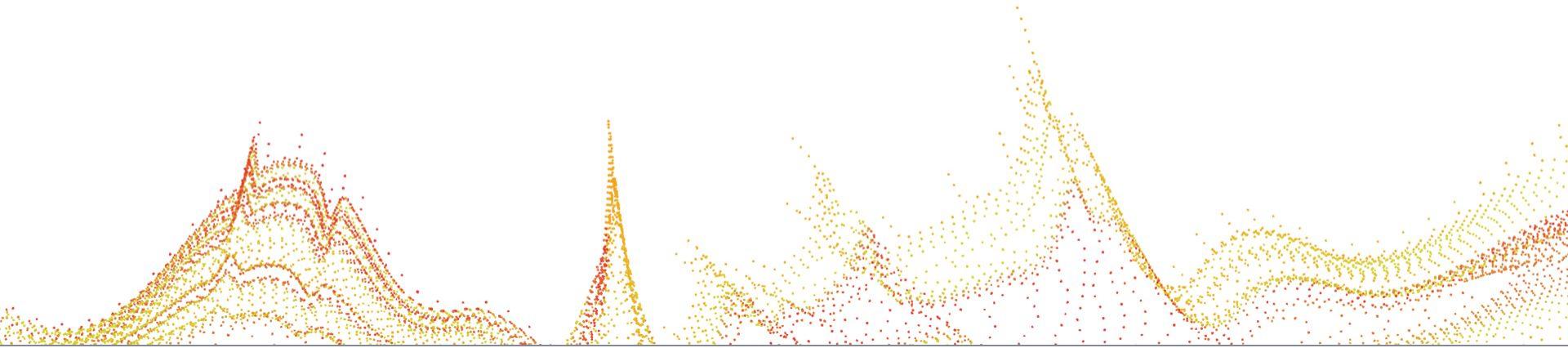
# PQC design for long-lived devices

Morgane Guerreau



Public - Copyright PQShield Ltd - All Rights Reserved

# Problem statement

- Build now
- Sell everywhere
- Last long

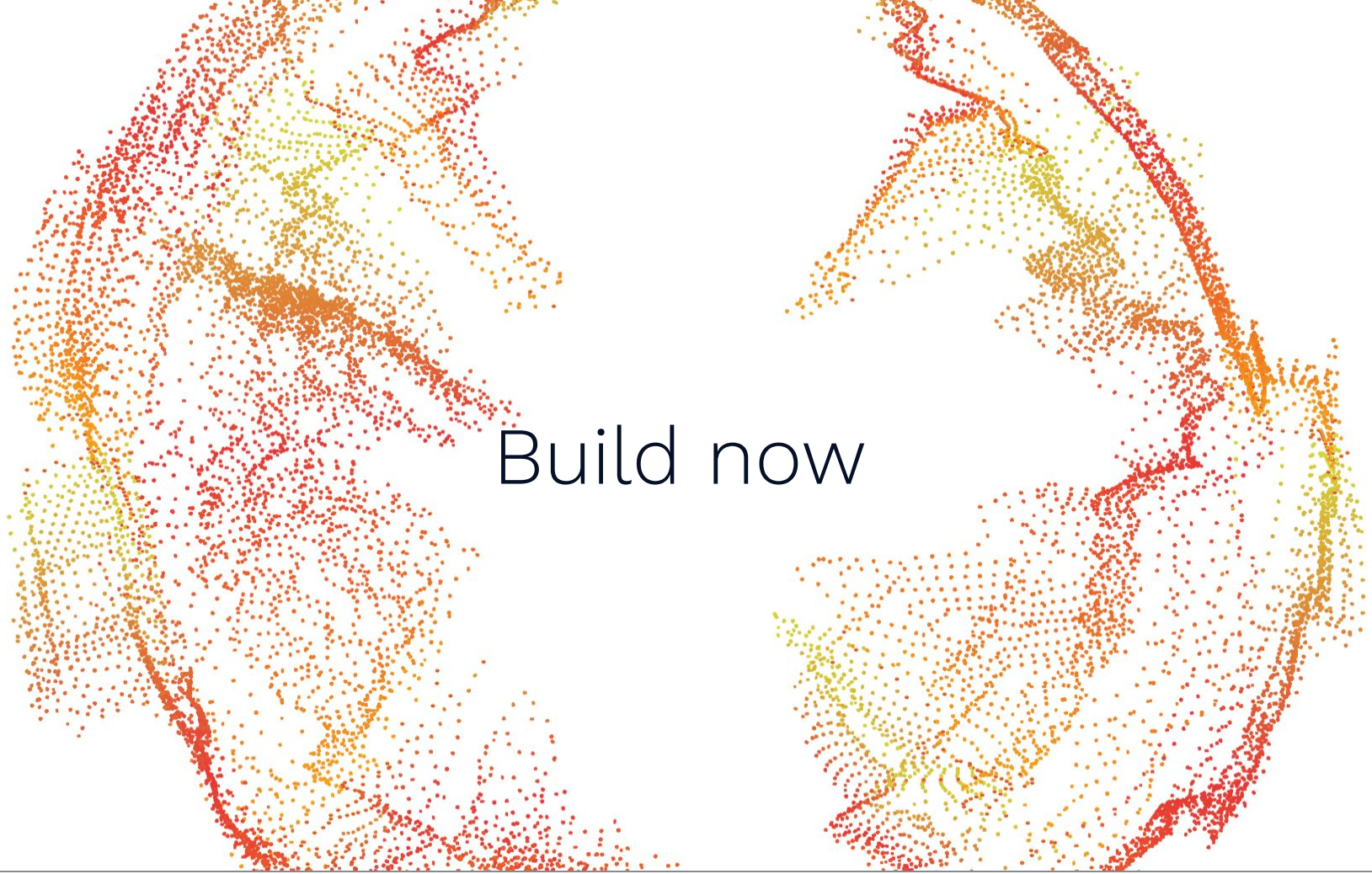
Public - Copyright PQShield Ltd - All Rights Reserved

# Problem statement

- Build now
- Sell everywhere
- Last long

(but it’s only a 15 minutes talk)

Public - Copyright PQShield Ltd - All Rights Reserved




Public - Copyright PQShield Ltd - All Rights Reserved

# Timelines … or not

**We don't have time for timelines, we already know that we need to act now!**

(also please focus on your asymmetric cryptography, AES-128 is fine I promise)

Public - Copyright PQShield Ltd - All Rights Reserved

# Today's problems and solutions

**Availability**

Go with what you have:

- ML-DSA, SLH-DSA or LMS/XMSS
- ML-KEM or FrodoKEM

FN-DSA and HQC are not ready yet.

Choices to be made by industry:

- What should be the preferred algorithm?
- What would be a backup algorithm?
- Which PQ/T hybrid combinations? (see next slide)

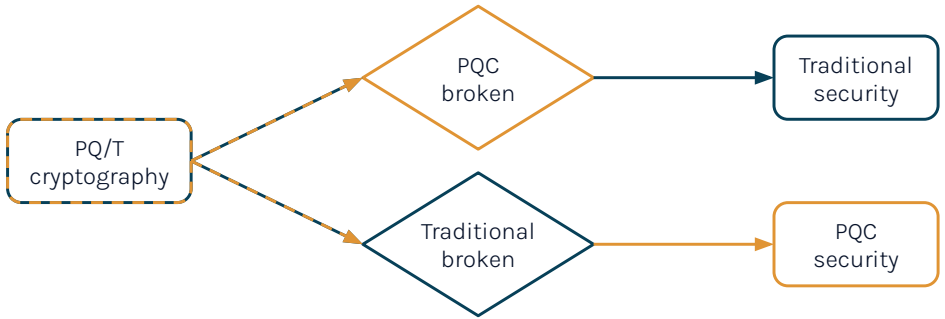
Missing PQC in some hardware components?

➔ Time to put pressure on your supply chain!

Public - Copyright PQShield Ltd - All Rights Reserved

# Today's problems and solutions

- Cryptanalysis is also rising in the traditional world.
- Be careful which components you use in PQ/T!
- RSA might very well end up broken by a traditional computer.




**Confidence**

Recent cryptanalysis events:

- SIKE (2022)
- Hawk (this summer)
- Classic McEliece (ongoing)

PQ/T hybrids may give you some assurance in the upcoming years

Public - Copyright PQShield Ltd - All Rights Reserved

# Today's problems and solutions

### Availability

Go with what you have:

- ML-DSA, SLH-DSA or LMS/XMSS
- ML-KEM or FrodoKEM

FN-DSA and HQC are not ready yet.

### Performance

Consider:

- Execution time
- Memory footprint
- Code size

It’s a matter of trade-offs, no one-size-fits-all solution.

### Confidence

Recent cryptanalysis events:

- SIKE (2022)
- Hawk (this summer)
- Classic McEliece (ongoing)

PQ/T hybrids may give you some assurance in the upcoming years

**Don’t let perfect be the enemy of good. Make choices now.**

Public - Copyright PQShield Ltd - All Rights Reserved

Public - Copyright PQShield Ltd - All Rights Reserved

# Regional guidelines who wants what

- Not everyone agrees with NIST standards
  - Some recommend additional algorithms
    - (FrodoKEM, *McEliece**)
  - Some recommend different algorithms
    - (South Korea, China)
- Not a new topic, some countries have always had their own preferences (e.g. brainpool curves)

[PQShield's blogpost on regional cryptography]

**Key takeaways:**

1. NIST standards will be accepted (almost) everywhere
2. Worry more about PQ/T support than FrodoKEM
3. Monitor upcoming standards

Public - Copyright PQShield Ltd - All Rights Reserved

# Cryptographic Agility definition

## Cryptographic agility is a capability

"The ability to update or replace **cryptographic components** within a system with **minimal disruption**"

- Cryptographic components:
  - Algorithm implementations, keys (public & private), random number generators, hardware accelerators, …
  - Protocols, security solutions, …
- Minimal disruption:
  - Change is always possible - agility implies quick, easy, cheap, …

Public - Copyright PQShield Ltd - All Rights Reserved

# Cryptographic Agility challenges

**Updatability**

- If the system cannot be updated, you need to deploy algorithms ahead of time (consider configuration/negotiation)

**Code size**

- It may be challenging to make multiple implementations available at the same time

**Attack surface**

- May introduce accrued vulnerability to fault injection attacks
- May open the door to downgrade attacks

**Hardcoded keys**

- If keys are in e.g. OTP memory, there is nothing you can do (secure boot / update)

Public - Copyright PQShield Ltd - All Rights Reserved

Public - Copyright PQShield Ltd - All Rights Reserved

# Cryptographic Agility survive cryptanalysis?

- Looks like an obvious solution to accommodate cryptanalysis advances
  - “Algorithm X is broken, move to algorithm Y”
- It’s not that simple…

**Encrypted Data**

- Loss of confidentiality, previously exchanged data is at risk
- If PQ/T was used, it’s less dramatic, but HN;DL threat applies again

**Secure Update**

- If algorithm X is broken, you cannot rely on it to sign your software update
- You may lose control of your device before switching algorithms
- Equivalent to a 0-day vulnerability

Public - Copyright PQShield Ltd - All Rights Reserved

# Cryptographic Agility alternatives

- Rely on **more conservative** algorithms
  - There is a consensus that hash-based algorithms are very secure
  - May not be practical everywhere
- Use **highest security** parameter sets
  - Not all cryptanalysis advances will break the scheme
  - Some of them will only lower the security
  - NIST level 5 may remain safe to use
- Consider **PQ/PQ** algorithms
  - Same idea as PQ/T except that both components are PQC
  - Choose a meaningful combination (not two algorithms relying on the same problem)
  - Can help mitigate operational risks (draft-gray-lamps-composite-fndsa-lms)
  - Will still be useful after Q-Day (as opposed to PQ/T)

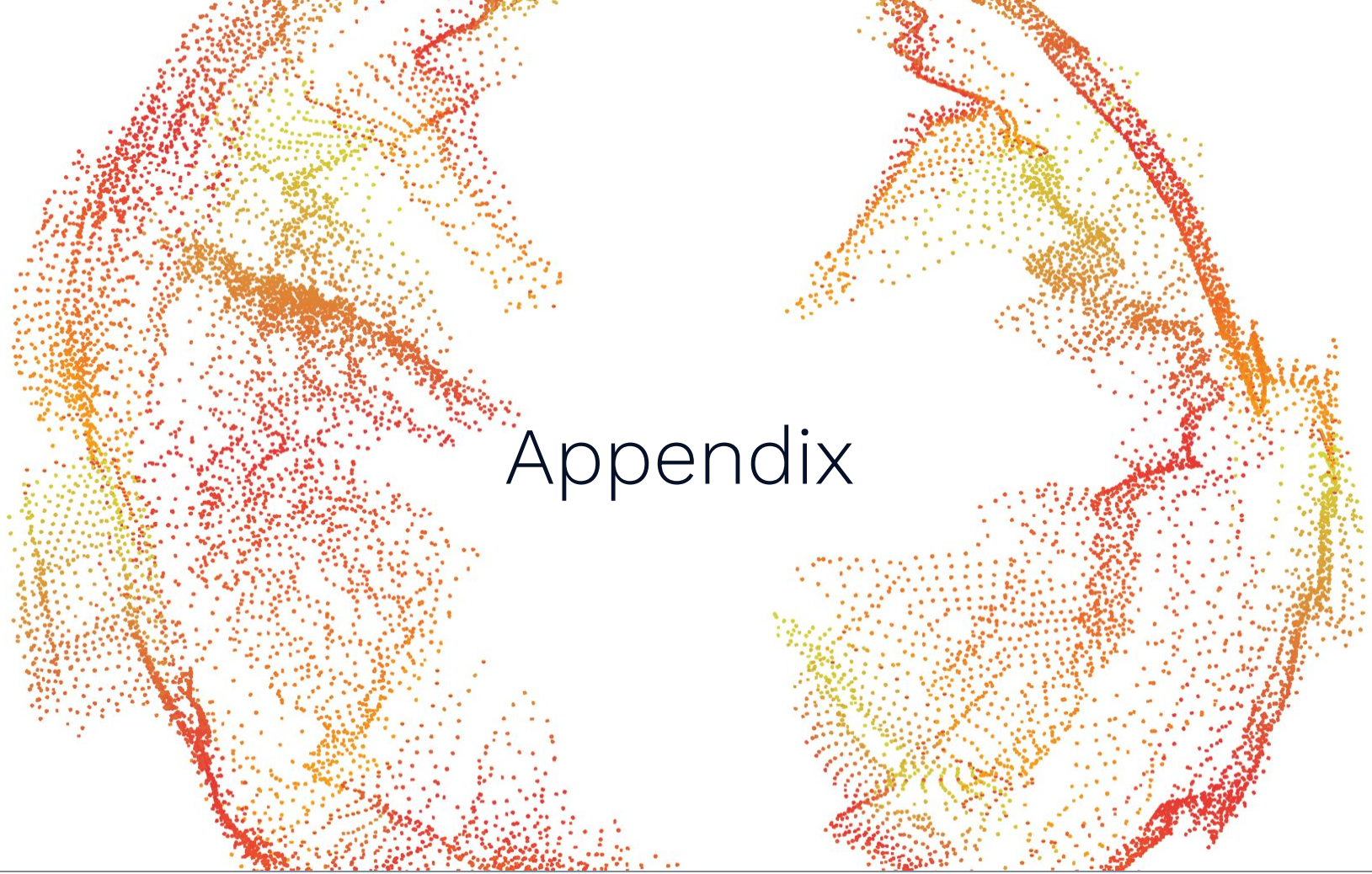
Public - Copyright PQShield Ltd - All Rights Reserved

# Conclusion

- Build **now** with what you have
- Watch out for regional regulations
  - Especially for **PQ/T** in Europe
- If that makes sense in your system, implement crypto-agility
  - Ensure that you can bundle several algorithms
- Plan for **long-term security**
  - Use highest parameter sets
  - Consider **PQ/PQ**
- Industry needs to agree on primary *and* backup algorithms

Questions?

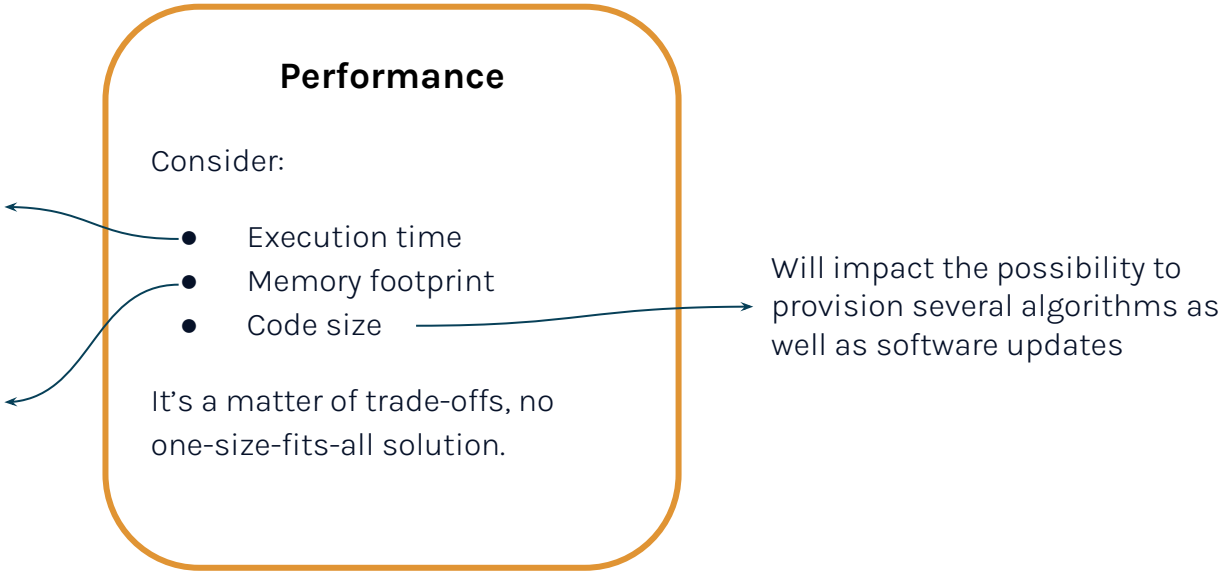
Public - Copyright PQShield Ltd - All Rights Reserved



# Appendix

Public - Copyright PQShield Ltd - All Rights Reserved

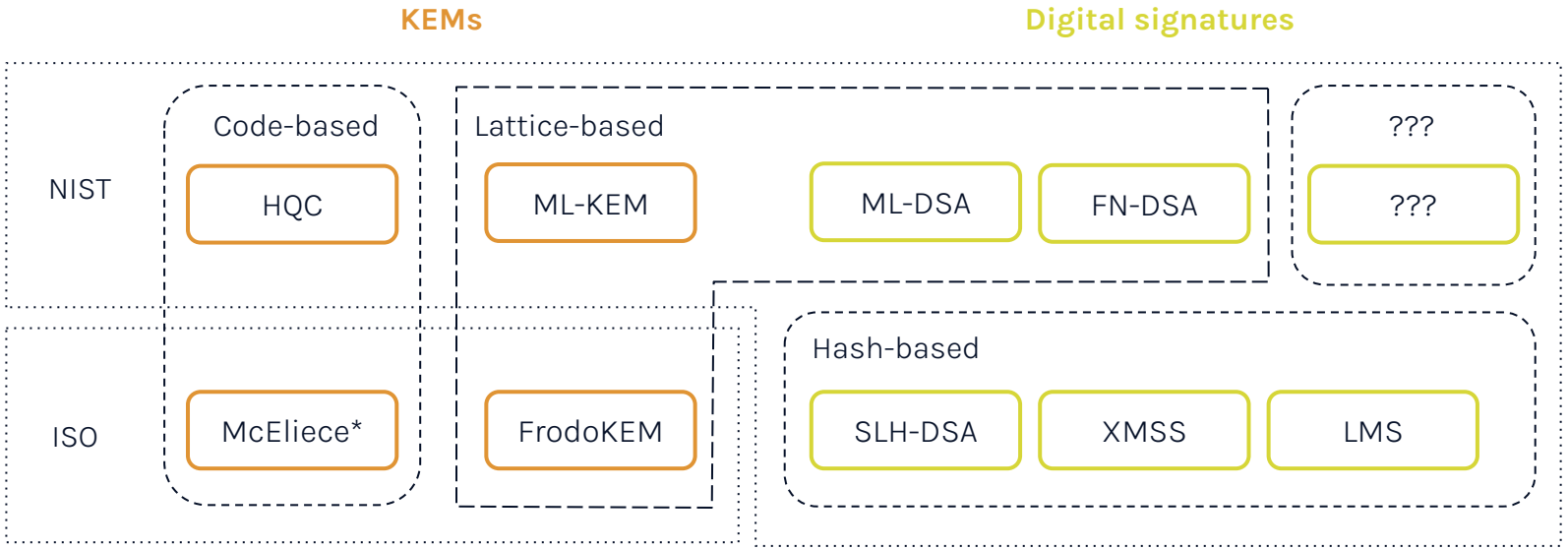
# Today's problems and solutions




Public - Copyright PQShield Ltd - All Rights Reserved

# **Standardisation** current status

There are **plenty of standards** and a **diverse cohort** of algorithms to choose from… and **more to come**.




Public - Copyright PQShield Ltd - All Rights Reserved

# PQ/T Cryptography how?

- Hybridization at the **primitive** level
  - Replace your traditional algorithm by a PQ/T one
  - This will be transparent for the 'consumer' of the implementation
- Hybridization at the **protocol** level
  - Some protocols already offer the possibility to do e.g. multiple rounds of authentication:
    - 1 round of traditional authentication + 1 round of PQC authentication
  - This will not work for most of the protocols.
- Hybridization at the **system** level
  - Combine two protocols, one traditional and one PQC
    - E.g. drop a PQC TLS proxy in the middle of your infrastructure
  - Last resort solution, when you can't update your existing stack

Public - Copyright PQShield Ltd - All Rights Reserved