b

UNIVERSITY
OF BERN

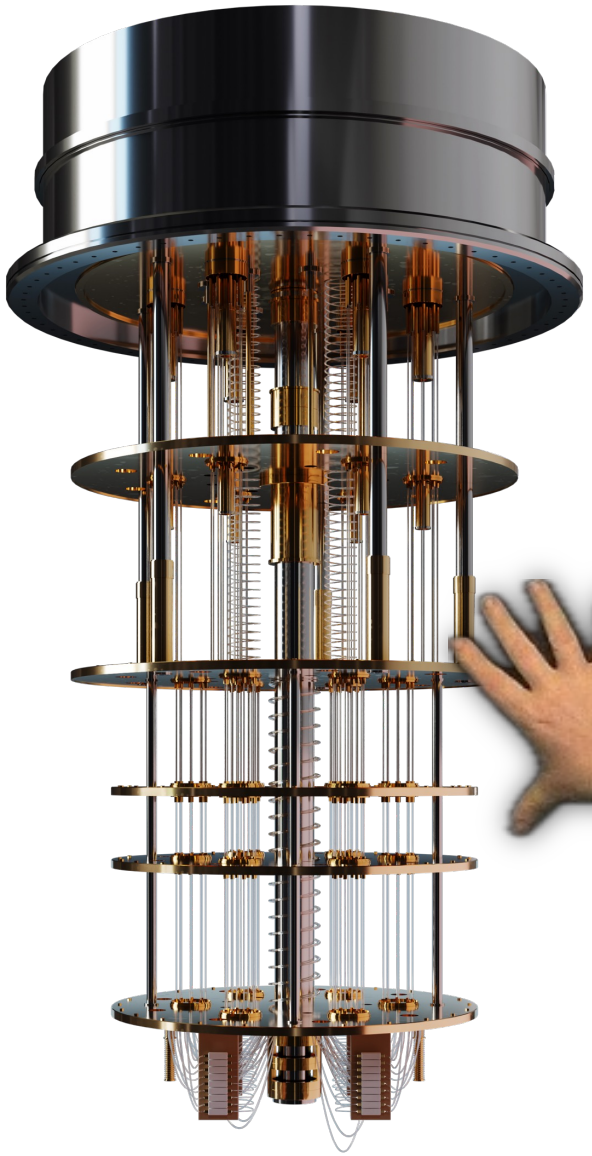
BF-CBOM: Uncovering Cryptographic Assets Through Comparative CBOM Analysis at Scale

Roman Bögli, Jonas Spieler, Timo Kehrer

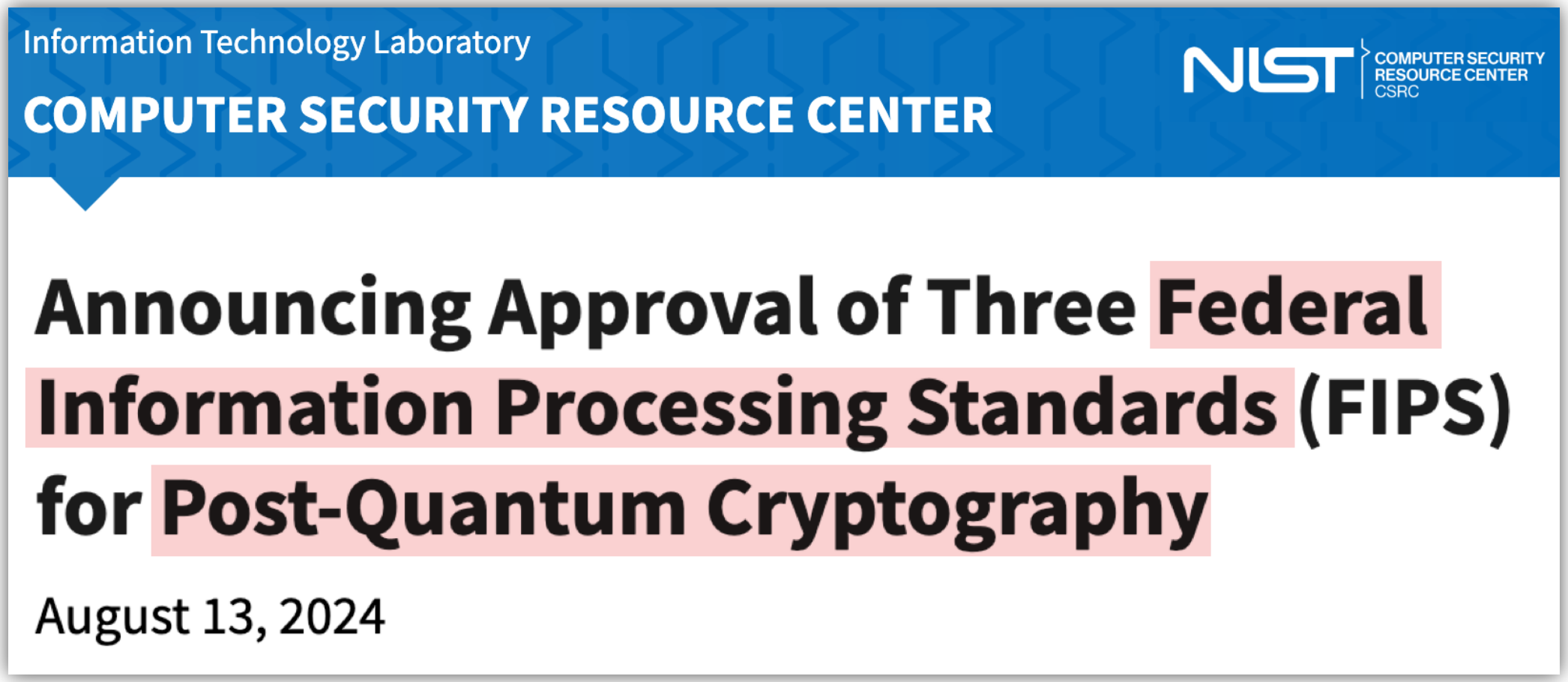
Software Engineering Group (SEG), University of Bern, Switzerland

ICPC'26, Rio de Janeiro, Brazil

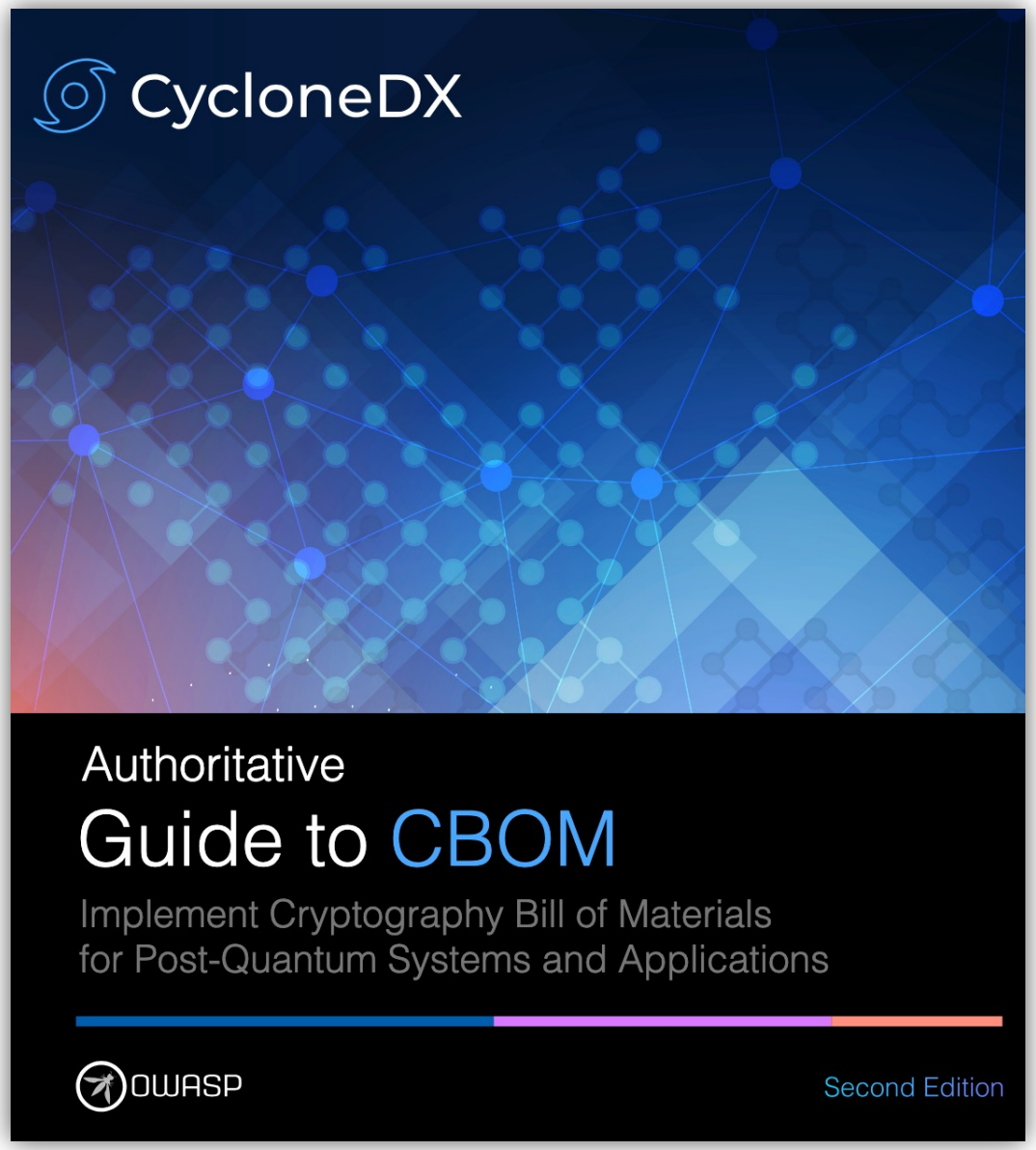
Prologue



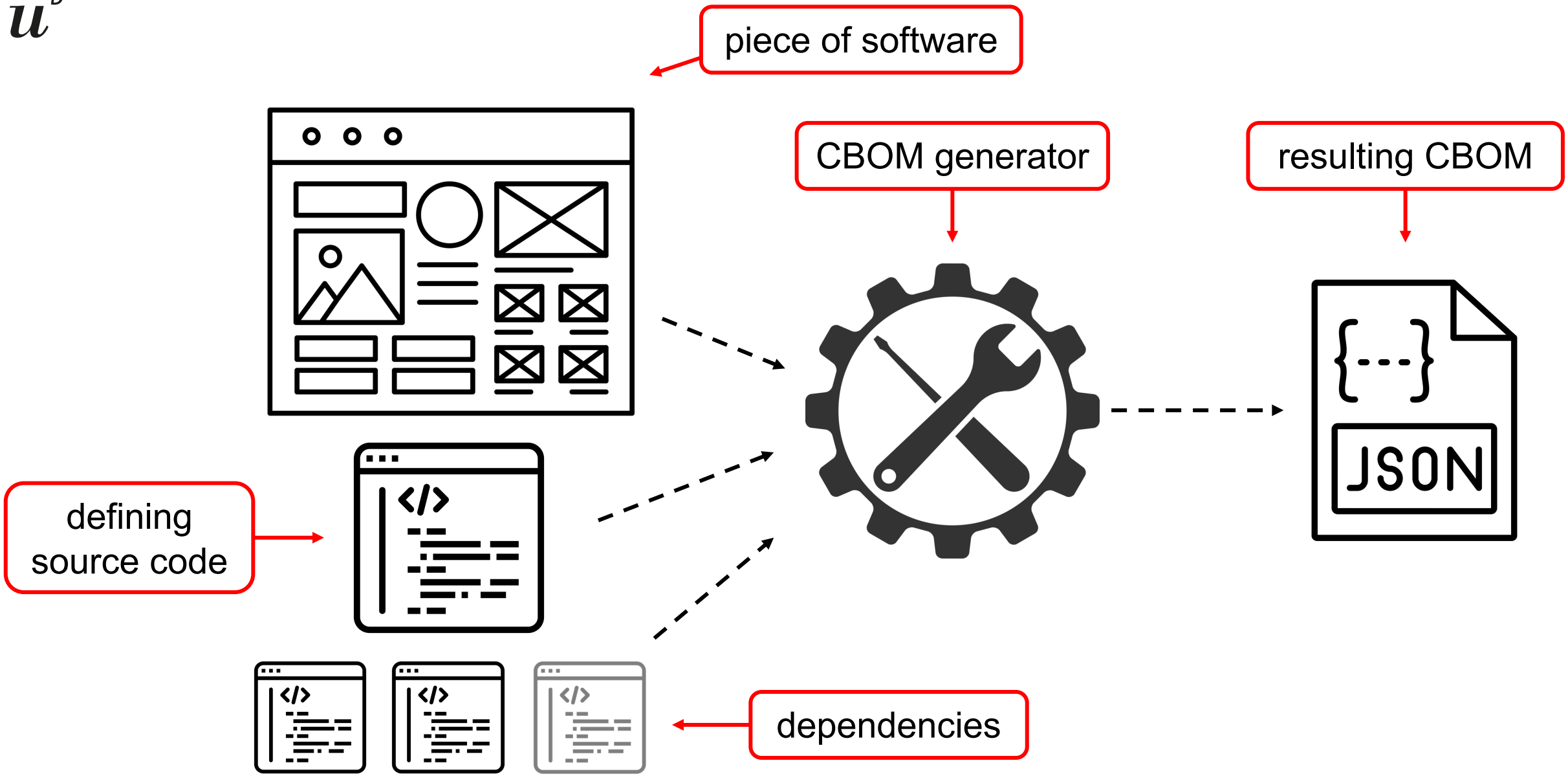

09:33
74
2
Peter Shor
Today
Messages to yourself are end-to-end encrypted. No one else, not even WhatsApp, can read, listen to, or share them. Learn more
waaaaazaap
09:21


Information Technology Laboratory
COMPUTER SECURITY RESOURCE CENTER
NIST COMPUTER SECURITY RESOURCE CENTER CSRC
Announcing Approval of Three Federal Information Processing Standards (FIPS) for Post-Quantum Cryptography
August 13, 2024

Protagonist




```
"components": [ {
    "name": "RSA-2048",
    "type": "cryptographic-asset",
    "bom-ref": "crypto/key/rsa-2048@1.2.840.113549.1.1.1",
    "cryptoProperties": {
      "assetType": "related-crypto-material",
      "relatedCryptoMaterialProperties": {
        "type": "public-key",
        "id": "2e9ef09e-dfac-4526-96b4-d02f31af1b22",
        "state": "active",
        "size": 2048,
        "algorithmRef": "crypto/algorithm/rsa-2048@1.2.840.113549.1.1.1",
        "securedBy": {
          "mechanism": "Software",
          "algorithmRef": "crypto/algorithm/aes-128-gcm@2.16.840.1.101.3.4.1.6"
        },
        "creationDate": "2016-11-21T08:00:00Z",
        "activationDate": "2016-11-21T08:20:00Z"
      },
      "oid": "1.2.840.113549.1.1.1"
    }
  }, {
    "name": "RSA-2048",
    "type": "cryptographic-asset",
    "bom-ref": "crypto/algorithm/rsa-2048@1.2.840.113549.1.1.1",
    "cryptoProperties": { ... }
  }, {
    "name": "AES-128-GCM",
```


piece of software
CBOM generator
resulting CBOM
defining source code
JSON
dependencies

BF-CBOM (our tool)

batch-generate CBOMs

Execution

Select benchmark

walk-through-1 · 81b0a51e · running

walk-through-1

ID: 81b0a51e-5c98-4857-b435-20d02cb35fd6

Created: 2025-09-25T19:46:28Z · **Jobs:** 40

repo	info	URL	cbomkit	cdxgen	cryptobomforge	deepseek
hacksider/Deep-Live-Cam	💬 Python · ⭐73,334 · 💾155,846 KB	🔗	✅ 43.0s · 0.6 KB	✅ 20.1s · 16.4 KB	💥 Failed	✅ 42.5s · 2.2 KB
Shubhamsaboo/awesome-llm-apps	💬 Python · ⭐70,113 · 💾191,934 KB	🔗	✅ 28.3s · 0.7 KB	✅ 20.9s · 819.3 KB	✅ 82.2s · 9.7 KB	✅ 30.3s · 1.6 KB
binary-husky/gpt_academic	💬 Python · ⭐69,255 · 💾72,551 KB	🔗	✅ 20.3s · 0.7 KB	✅ 17.0s · 63.9 KB	✅ 106.5s · 6.6 KB	✅ 73.9s · 3.3 KB
ansible/ansible	💬 Python · ⭐66,543 · 💾261,157 KB	🔗	⏰ Timeout	✅ 11.4s · 34.0 KB	✅ 306.7s · 5.6 KB	⏰ Timeout
xtekky/gpt4free	💬 Python · ⭐65,151 · 💾169,838 KB	🔗	⏰ Timeout	✅ 5.4s · 198.6 KB	✅ 91.6s · 14.3 KB	✅ 89.3s · 4.6 KB
jeecgboot/JeecgBoot	💬 Java · ⭐43,961 · 💾80,062 KB	🔗	💤	💤	💥 Failed	💤

Completed 32/40

Show analysis Cancel

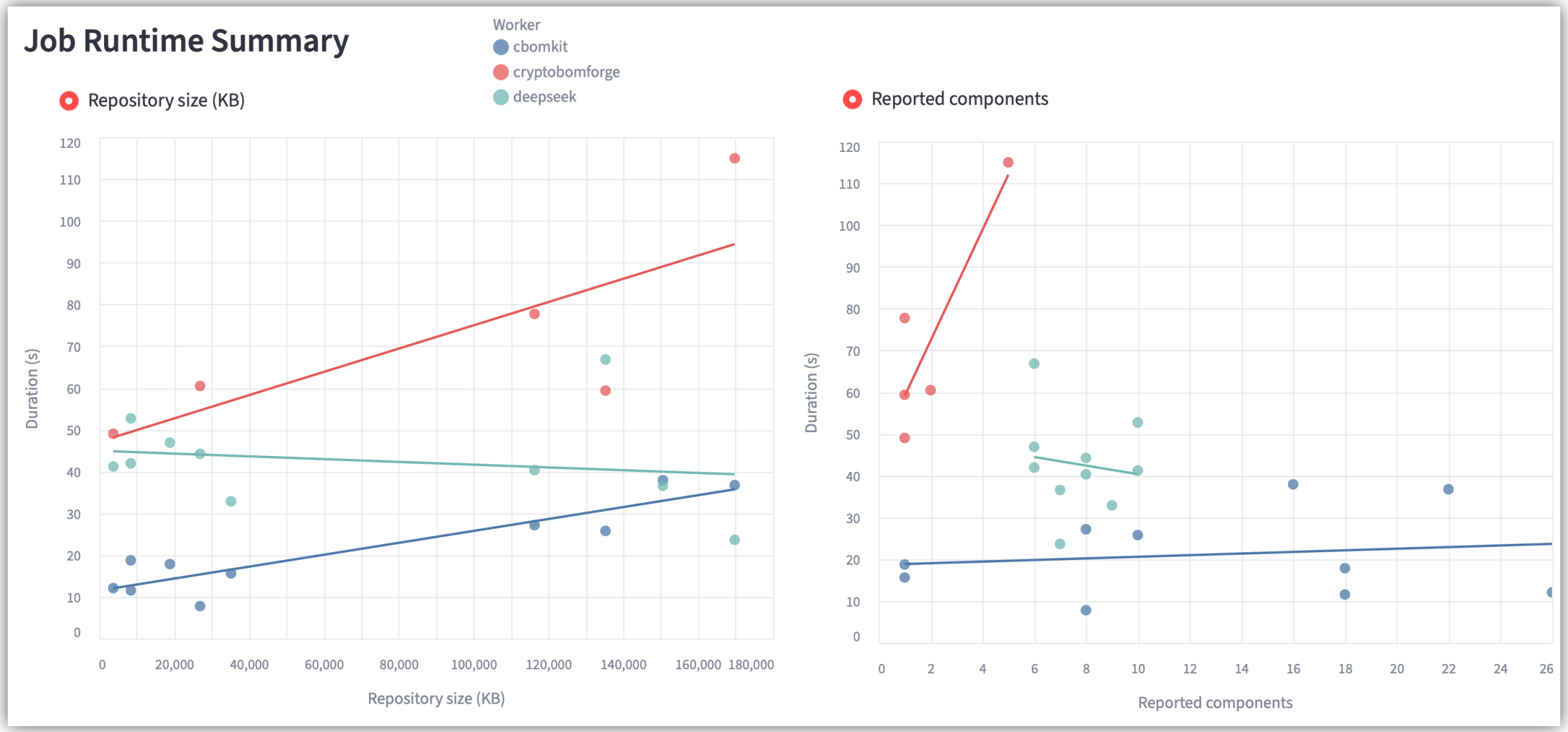
Notes

- 💤 Pending: job queued or running.
- ✅ Completed: worker returned a CBOM successfully.
- 💥 Failed: worker errored while processing the repo.
- ⏰ Timeout: worker exceeded its time budget.
- 🚫 Cancelled: job was cancelled before completion.

batch-generate CBOMs

repo	cbomkit	cdxgen	cryptobomforge	deepseek
hacksider/Deep-Live-Cam	✅ 43.0s · 0.6 KB	✅ 20.1s · 16.4 KB	💥 Failed	✅ 42.5s · 2.2 KB
Shubhamsaboo/awesome-llm-apps	✅ 28.3s · 0.7 KB	✅ 20.9s · 819.3 KB	✅ 82.2s · 9.7 KB	✅ 30.3s · 1.6 KB
binary-husky/gpt_academic	✅ 20.3s · 0.7 KB	✅ 17.0s · 63.9 KB	✅ 106.5s · 6.6 KB	✅ 73.9s · 3.3 KB
ansible/ansible	⏰ Timeout	✅ 11.4s · 34.0 KB	✅ 306.7s · 5.6 KB	⏰ Timeout
xtekky/gpt4free	⏰ Timeout	✅ 5.4s · 198.6 KB	✅ 91.6s · 14.3 KB	✅ 89.3s · 4.6 KB
jeecgboot/JeecgBoot	💤	💤	💥 Failed	💤

analyze & visualize



explore & compare

cbomkit component #16


```
{
  "name" : "Blowfish"
  "type" : "cryptographic-asset"
  "bom-ref" : "db5ec993-eef9-42df-ac5b-99abe16e7c1d"
  "evidence" : {
    "occurrences" : [
      0 : {
        "line" : 29
        "offset" : 18
        "location" :
        "chat2db-server/chat2db-server-web/chat2db-
        server-web-api/src/main/java/ai/chat2db/
        server/web/api/controller/ncx/cipher/
        Navicat11Cipher.java"
        "additionalContext" :
        "javax.crypto.spec.SecretKeySpec#<init>([BLjava/
        lang/String;)V"
```


cdxgen component #1791


```
{
  "type" : "cryptographic-asset"
  "name" : "blowfishECB"
  "bom-ref" :
  "crypto/algorithm/blowfishECB@1.3.6.1.4.1.3029.1.1.1"
  "description" : "cryptlib encryption algorithm"
  "cryptoProperties" : {
    "assetType" : "algorithm"
    "oid" : "1.3.6.1.4.1.3029.1.1.1"
  }
  "tags" : [
    0 : "cryptographic-asset"
  ]
}
```

explore & compare


```
{
  "name" : "Blowfish"
  "type" : "cryptographic-asset"
  "bom-ref" : "db5ec993-eef9-42df
  "evidence" : {
    "occurrences" : [
      0 : {
        "line" : 29
        "offset" : 18
```

```
{
  "type" : "cryptographic-asset"
  "name" : "blowfishECB"
  "bom-ref" :
  "crypto/algorithm/blowfishECB@1
  "description" : "cryptlib encry
  "cryptoProperties" : {
    "assetType" : "algorithm"
    "oid" : "1.3.6.1.4.1.3029.1.
```

explore & compare

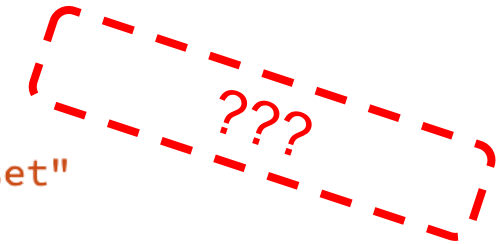

```
{
  "name" : "Blowfish"
  "type" : "cryptographic-asset"
  "bom-ref" : "db5ec993-eef9-42df
  "evidence" : {
    "occurrences" : [
      0 : {
        "line" : 29
        "offset" : 18
```

```
{
  "type" : "cryptographic-asset"
  "name" : "blowfishECB"
  "bom-ref" :
  "crypto/algorithm/blowfishECB@1
  "description" : "cryptlib encry
  "cryptoProperties" : {
    "assetType" : "algorithm"
    "oid" : "1.3.6.1.4.1.3029.1.
```

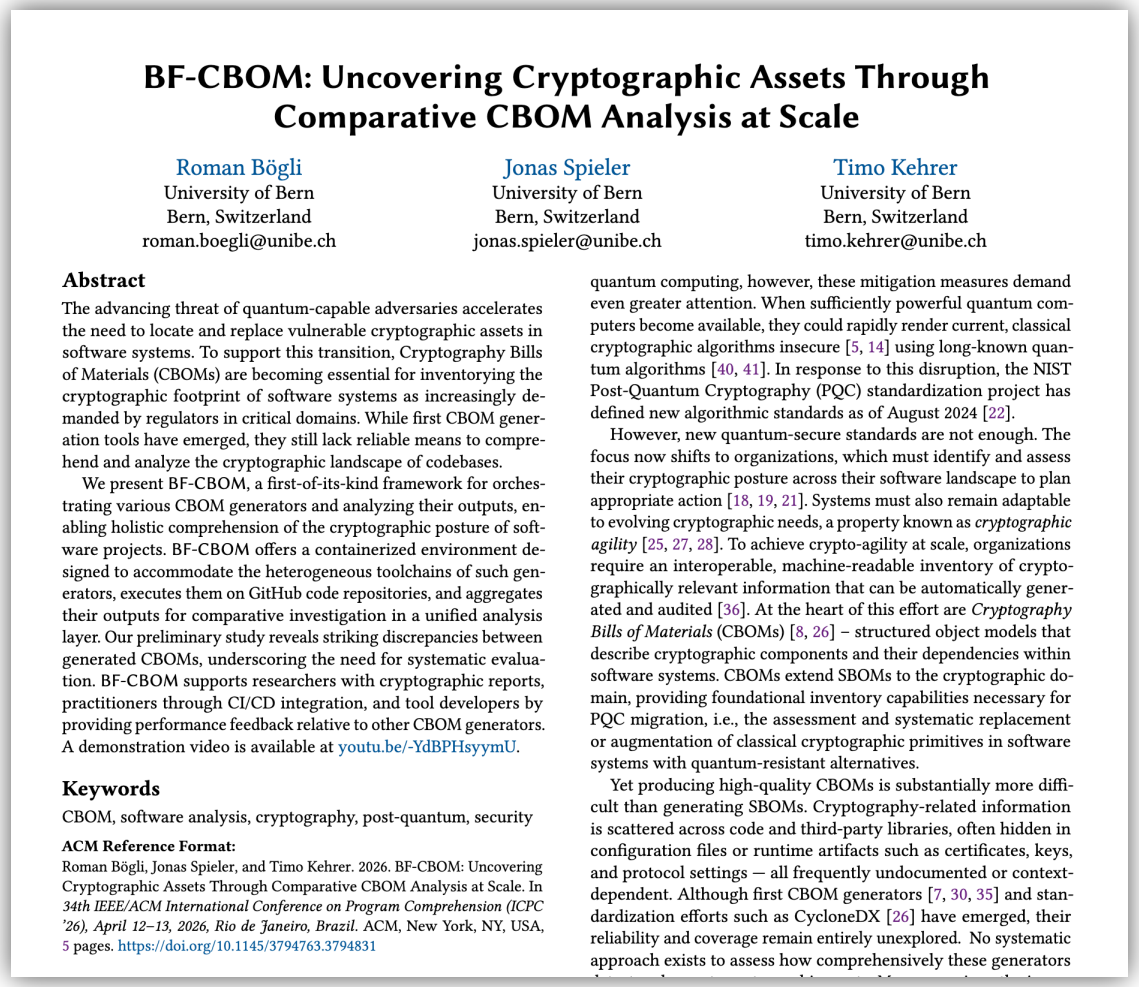
explore & compare


```
"evidence" : {
  "occurrences" : [
    0 : {
      "line" : 29
      "offset" : 18
      "location" :
      "chat2db-server/chat2db-
      server-web-api/src/main/
      server/web/api/controlle
      Navicat11Cipher.java"
```

```
{
  "type" : "cryptographic-asset"
  "name" : "blowfishECB"
  "bom-ref" :
  "crypto/algorithm/blowfishECB@1.3.6.1.4.1.3029.
  "description" : "cryptlib encryption algorithm"
  "cryptoProperties" : {
    "assetType" : "algorithm"
    "oid" : "1.3.6.1.4.1.3029.1.1.1"
  }
  "tags" : [
    0 : "cryptographic-asset"
  ]
}
```

BF-CBOM: Uncovering Cryptographic Assets Through Comparative CBOM Analysis at Scale



BF-CBOM: Uncovering Cryptographic Assets Through Comparative CBOM Analysis at Scale

Roman Bögli
University of Bern
Bern, Switzerland
roman.boegli@unibe.ch

Jonas Spieler
University of Bern
Bern, Switzerland
jonas.spieler@unibe.ch

Timo Kehrer
University of Bern
Bern, Switzerland
timo.kehrer@unibe.ch

Abstract

The advancing threat of quantum-capable adversaries accelerates the need to locate and replace vulnerable cryptographic assets in software systems. To support this transition, Cryptography Bills of Materials (CBOMs) are becoming essential for inventorying the cryptographic footprint of software systems as increasingly demanded by regulators in critical domains. While first CBOM generation tools have emerged, they still lack reliable means to comprehend and analyze the cryptographic landscape of codebases.

We present BF-CBOM, a first-of-its-kind framework for orchestrating various CBOM generators and analyzing their outputs, enabling holistic comprehension of the cryptographic posture of software projects. BF-CBOM offers a containerized environment designed to accommodate the heterogeneous toolchains of such generators, executes them on GitHub code repositories, and aggregates their outputs for comparative investigation in a unified analysis layer. Our preliminary study reveals striking discrepancies between generated CBOMs, underscoring the need for systematic evaluation. BF-CBOM supports researchers with cryptographic reports, practitioners through CI/CD integration, and tool developers by providing performance feedback relative to other CBOM generators. A demonstration video is available at youtu.be/-YdBPHsyymU.

Keywords

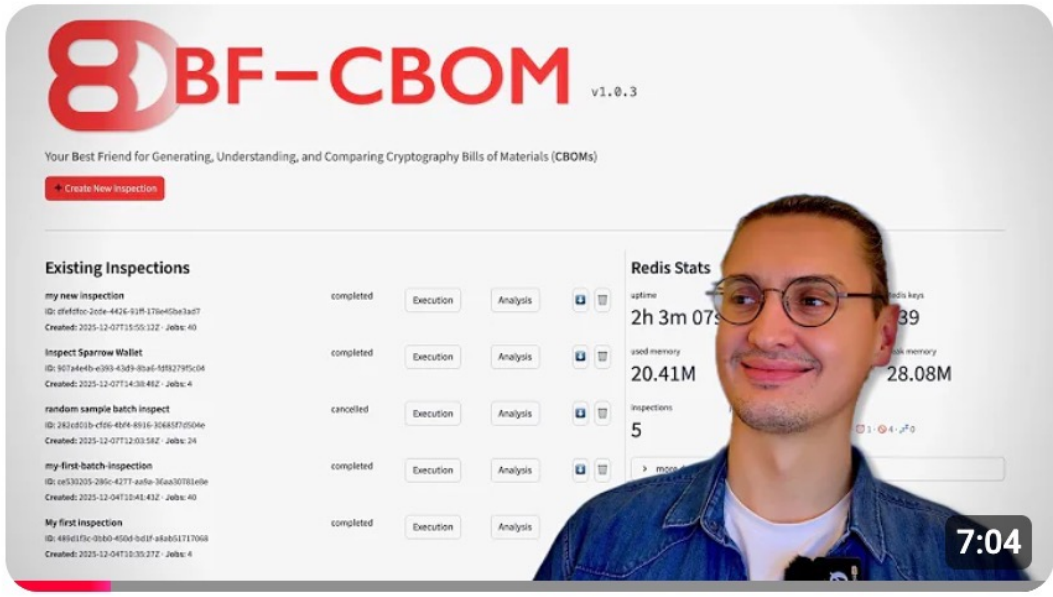
CBOM, software analysis, cryptography, post-quantum, security

ACM Reference Format:
Roman Bögli, Jonas Spieler, and Timo Kehrer. 2026. BF-CBOM: Uncovering Cryptographic Assets Through Comparative CBOM Analysis at Scale. In *34th IEEE/ACM International Conference on Program Comprehension (ICPC '26), April 12–13, 2026, Rio de Janeiro, Brazil.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3794763.3794831

quantum computing, however, these mitigation measures demand even greater attention. When sufficiently powerful quantum computers become available, they could rapidly render current, classical cryptographic algorithms insecure [5, 14] using long-known quantum algorithms [40, 41]. In response to this disruption, the NIST Post-Quantum Cryptography (PQC) standardization project has defined new algorithmic standards as of August 2024 [22].

However, new quantum-secure standards are not enough. The focus now shifts to organizations, which must identify and assess their cryptographic posture across their software landscape to plan appropriate action [18, 19, 21]. Systems must also remain adaptable to evolving cryptographic needs, a property known as *cryptographic agility* [25, 27, 28]. To achieve crypto-agility at scale, organizations require an interoperable, machine-readable inventory of cryptographically relevant information that can be automatically generated and audited [36]. At the heart of this effort are *Cryptography Bills of Materials* (CBOMs) [8, 26] – structured object models that describe cryptographic components and their dependencies within software systems. CBOMs extend SBOMs to the cryptographic domain, providing foundational inventory capabilities necessary for PQC migration, i.e., the assessment and systematic replacement or augmentation of classical cryptographic primitives in software systems with quantum-resistant alternatives.

Yet producing high-quality CBOMs is substantially more difficult than generating SBOMs. Cryptography-related information is scattered across code and third-party libraries, often hidden in configuration files or runtime artifacts such as certificates, keys, and protocol settings — all frequently undocumented or context-dependent. Although first CBOM generators [7, 30, 35] and standardization efforts such as CycloneDX [26] have emerged, their reliability and coverage remain entirely unexplored. No systematic approach exists to assess how comprehensively these generators



BF-CBOM: Your Best Friend for Cryptography Bill of Materials (DEMO)

20M views • 4 months ago

Roman Bögli

https://py.md/besLc

https://py.md/QzsZ5

Attribution-ShareAlike 4.0 International (CC BY-SA 4.0)

You are free to:

Share: copy and redistribute the material in any medium or format

Adapt: remix, transform, and build upon the material for any purpose, even commercially.

Under the following terms:

Attribution You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

ShareAlike If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.

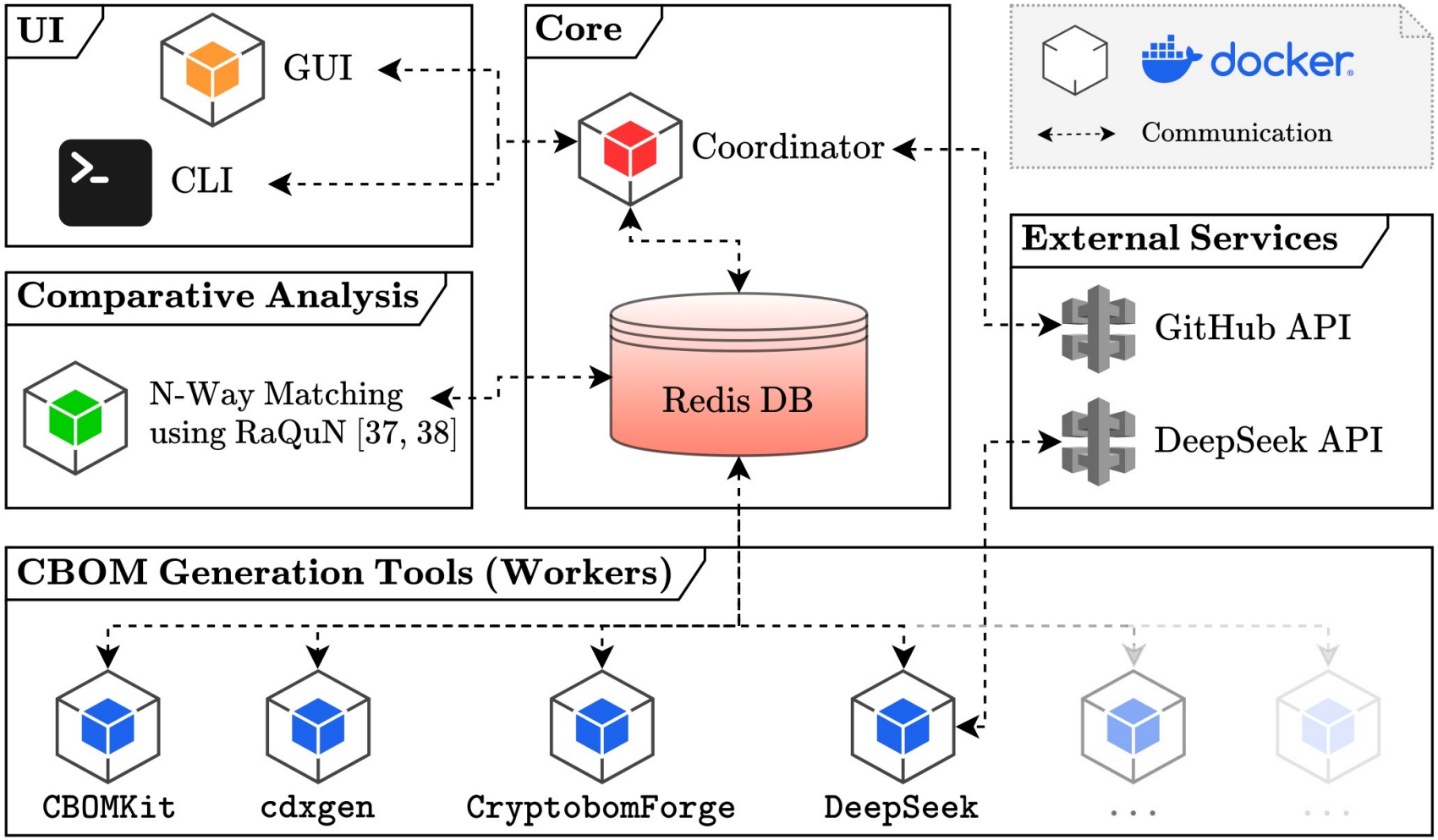
No additional restrictions You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

https://creativecommons.org/licenses/by-sa/4.0/

Appendix

Appendix

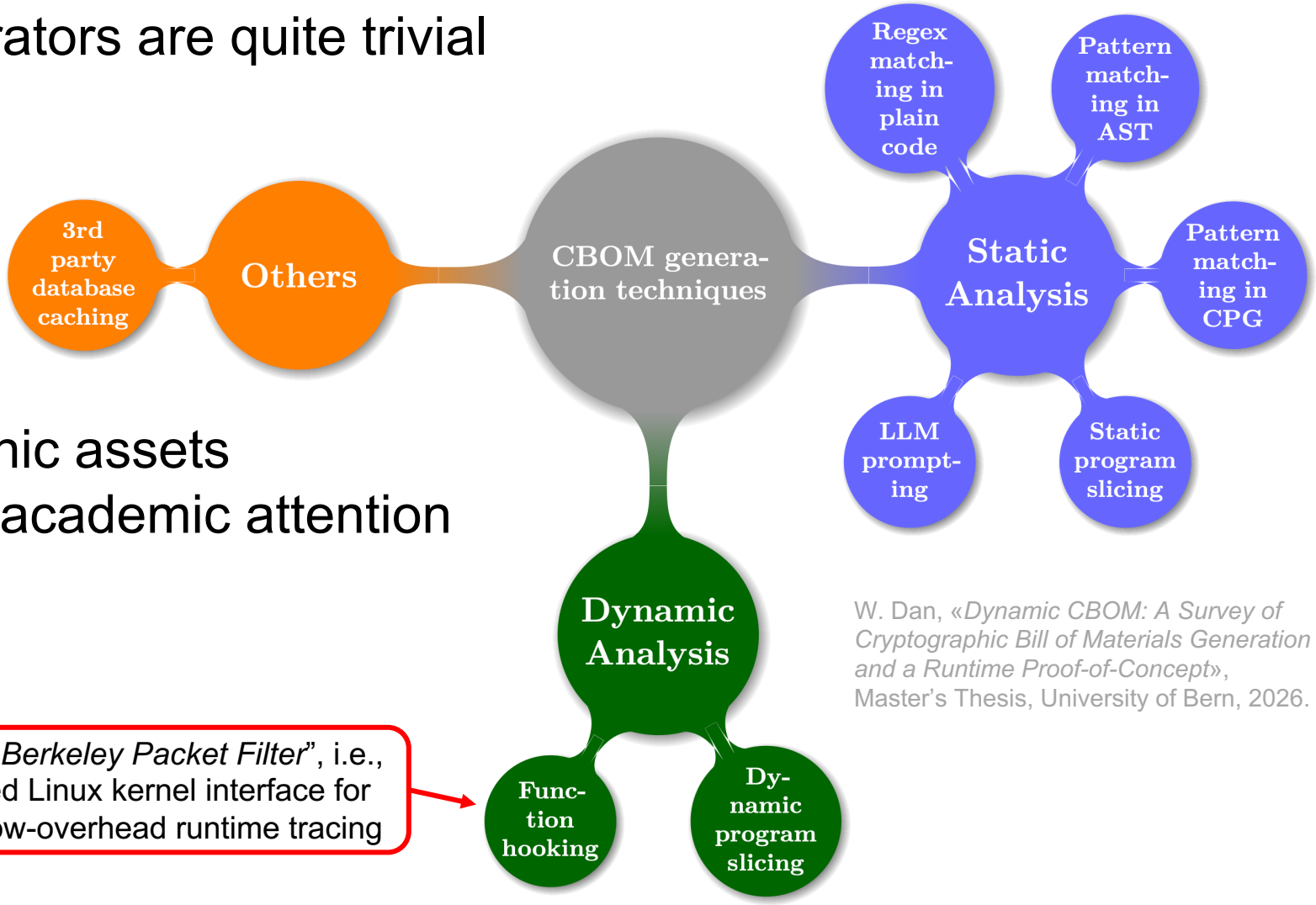
BF-CBOM‘s System Architecture



Appendix

CBOM Generation Techniques

- Current CBOM generators are quite trivial and mostly static



- Detecting cryptographic assets on runtime demands academic attention

eBPF

"*extended Berkeley Packet Filter*", i.e., sandboxed Linux kernel interface for efficient, low-overhead runtime tracing

W. Dan, «*Dynamic CBOM: A Survey of Cryptographic Bill of Materials Generation and a Runtime Proof-of-Concept*», Master's Thesis, University of Bern, 2026.

Appendix

IBM‘s CBOMKit

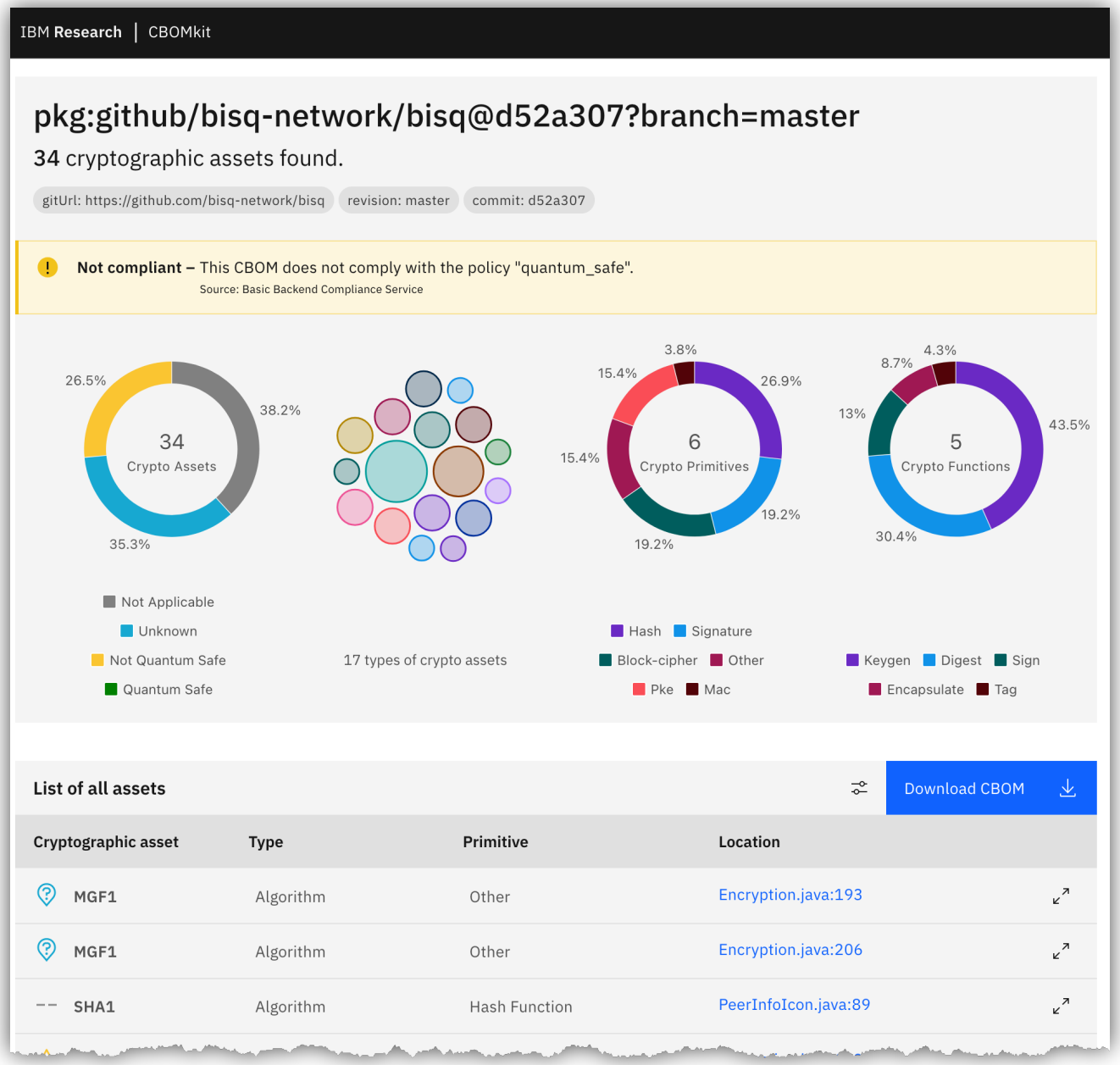
PQCA/cbomkit

A toolset for dealing with Cryptography Bill of Materials (CBOM)

cryptography · post-quantum-cryptography · quantum-safe · post-quantum · sbom

Java · ☆ 34 · Updated 3 hours ago

IBM, CBOMkit. GitHub. https://github.com/PQCA/cbomkit

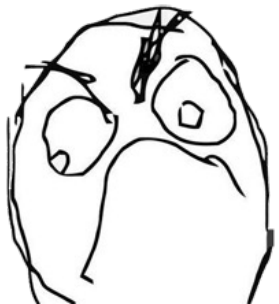


Appendix

Comparative Results (1/3)

Table 5.1: Amount of Empty CBOMs per Tool

Tool	Repos	Non-Empty	Empty	Empty %
CBOMkit	100	59	41	41.0%
cdxgen	99	67	32	32.3%
DeepSeek	100	87	13	13.0%

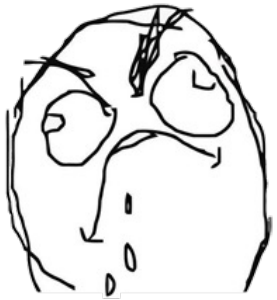


Appendix

Comparative Results (2/3)

Table 5.2: Amount of Components per Repository

Tool	Total Comp.	Avg/Repo	Avg/Non-Empty
CBOMkit	2‘327	23.3	39.4
cdxgen	38‘354	387.4	572.4
DeepSeek	516	5.2	5.9

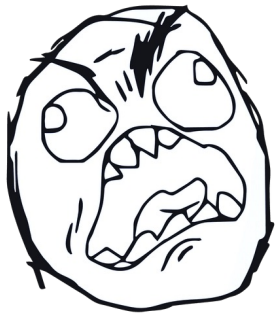


Appendix

Comparative Results (3/3)

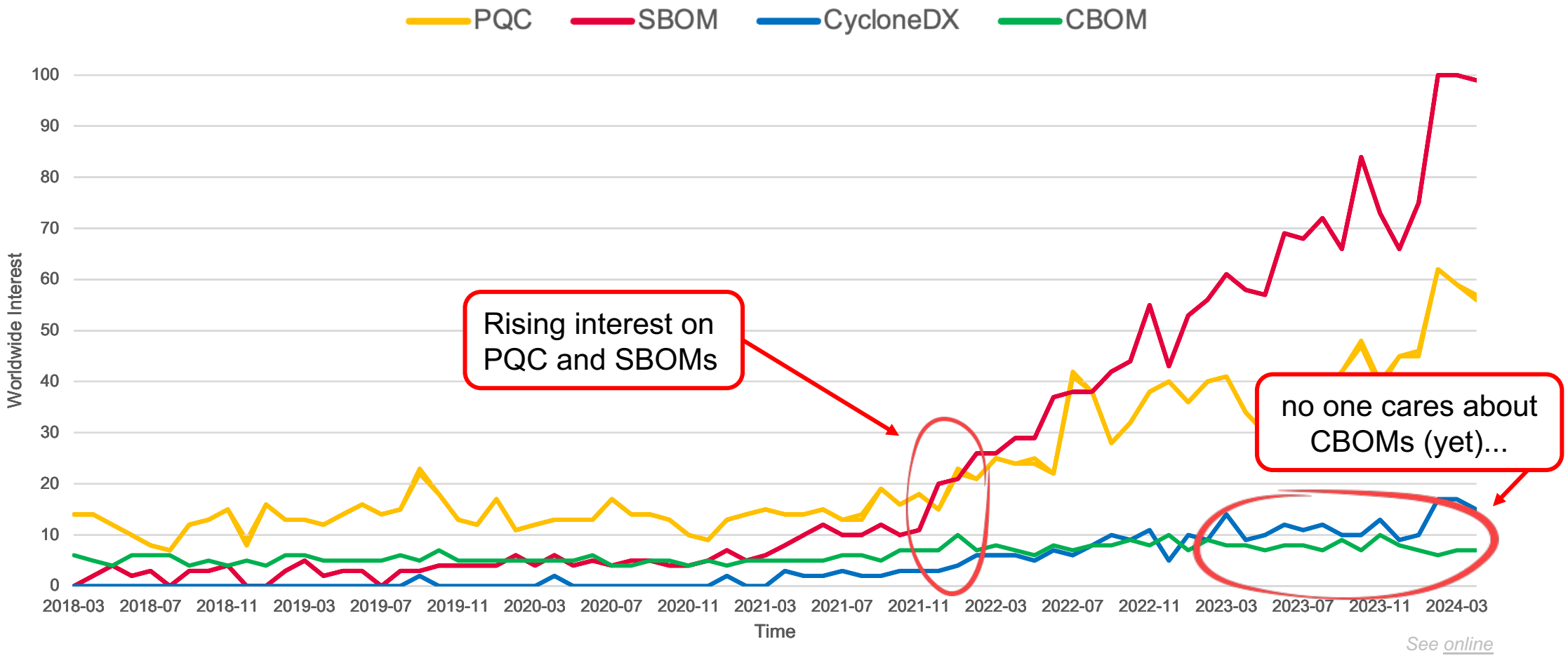
Table 5.3: Component Type Distribution

Component Type	CBOMkit	cdxgen	DeepSeek
algorithm	323	0	0
framework	0	16‘845	0
library	0	21‘509	516
protocol	25	0	0
related-crypto-material	1‘979	0	0



Appendix

Google Trends



See online

Interest over time: *Numbers represent search interest relative to the highest point on the chart for the given region and time. A value of 100 is the peak popularity for the term. A value of 50 means that the term is half as popular. A score of 0 means that there was not enough data for this term.*

Appendix

Challenges

- How does our IT infrastructure look like?
  - Which software components run where and how do they interact?
  - Which internal/external parties are involved?
  - Who and/or what has what kind of user rights on what?
- How does our cryptographic posture look like?
  - Where and how is what kind of data stored?
  - Which cryptographic assets/primitives are where in use?
  - How do the thread models look like on these cryptographic point of interests?
- How can we become cryptographically agile?
  - What needs where to be done to seamlessly transition between primitives?
  - How, where, and when shall we move towards post-quantum cryptography?

Appendix

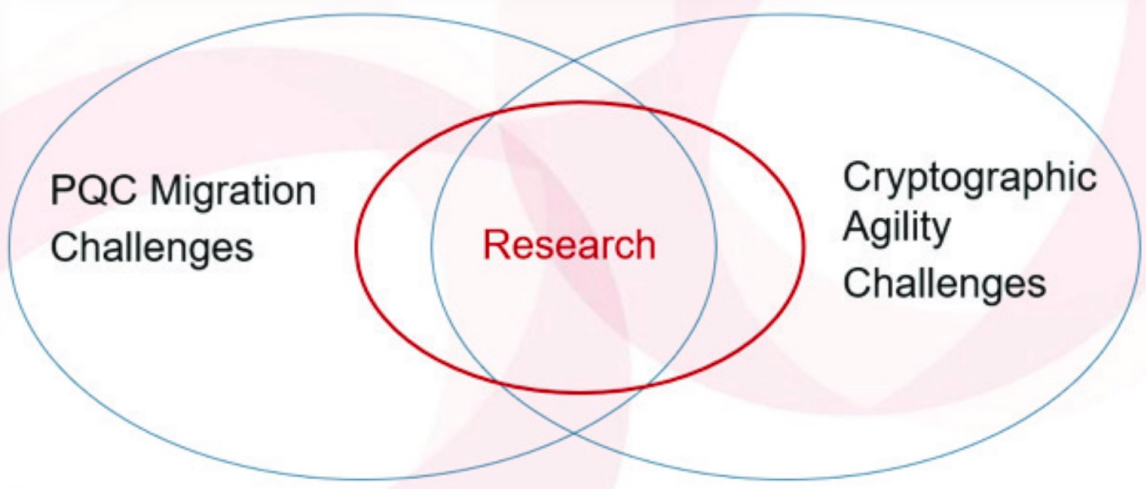
Crypto-Agility as Broader Topic

1.3 The Need for Research

The complex challenge of migrating our global compute infrastructure to new public-key cryptography standards will involve work on many levels, and we argue that *the area overall is in dire need of research*. That is, before the global industry ecosystem can deploy quantum safe solutions, there is considerable work to be done understanding migration challenges and schemes, and more rigorously addressing integration, security, performance, agility, and other challenges.

D. Ott, C. Peikert, and other workshop participants, "Identifying Research Challenges in Post Quantum Cryptography Migration and Cryptographic Agility." arXiv, Sep. 16, **2019**. doi: 10.48550/arXiv.1909.07353.

In many ways, cryptographic agility represents the generalization of PQC migration in that it considers not just the current challenge of migrating from our current algorithms to PQC alternatives, but the longer-term need for ongoing migrations as new attacks and better algorithms motivate the need for updates in our cryptographic standards.



Appendix

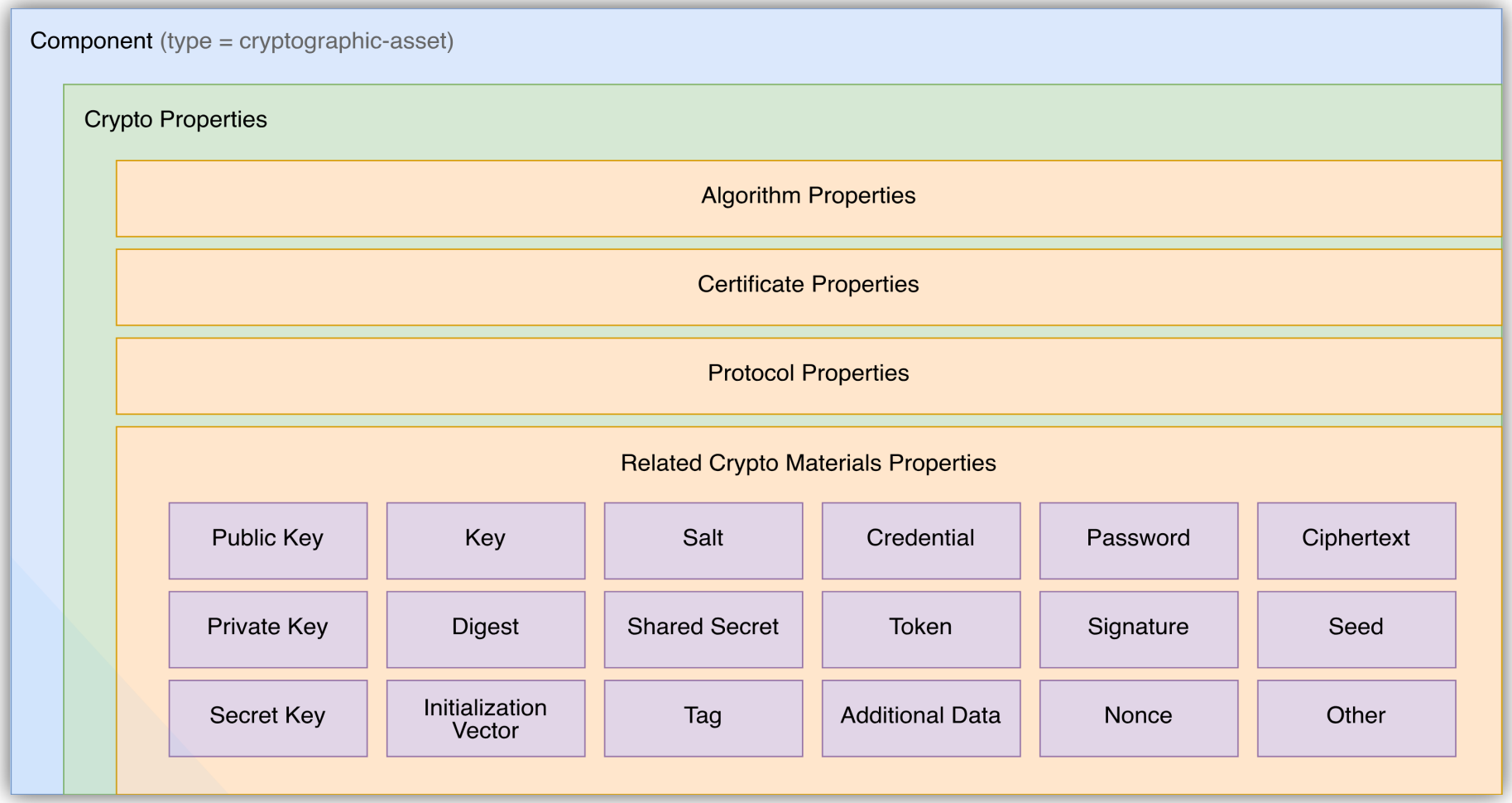
CycloneDX

- "*CycloneDX v1.6 simplifies the discovery, management, and reporting of cryptographic assets, laying the groundwork for migration to quantum-safe systems and applications. It facilitates the identification of weak cryptographic algorithms, promotes cryptographic agility, and ensures compliance with evolving cryptographic policies and advisories like CNSA 2.0, aligning with recommendations from NIST.*"
- "*CBOM is the first open standard to describe an organizations' cryptographic assets inventory, and their dependencies, giving organizations deeper visibility into the cryptography they use, enabling them to assess their quantum readiness, and to consider actionable steps towards becoming quantum safe.*"

S. Springett, "CycloneDX v1.6 Released, Advances Software Supply Chain Security with Cryptographic Bill of Materials and Attestations," OWASP. Accessed: Apr. 10, 2024. [Online]. Available: https://owasp.org/blog/2024/04/09/CycloneDX-v1.6-Released.html

Appendix

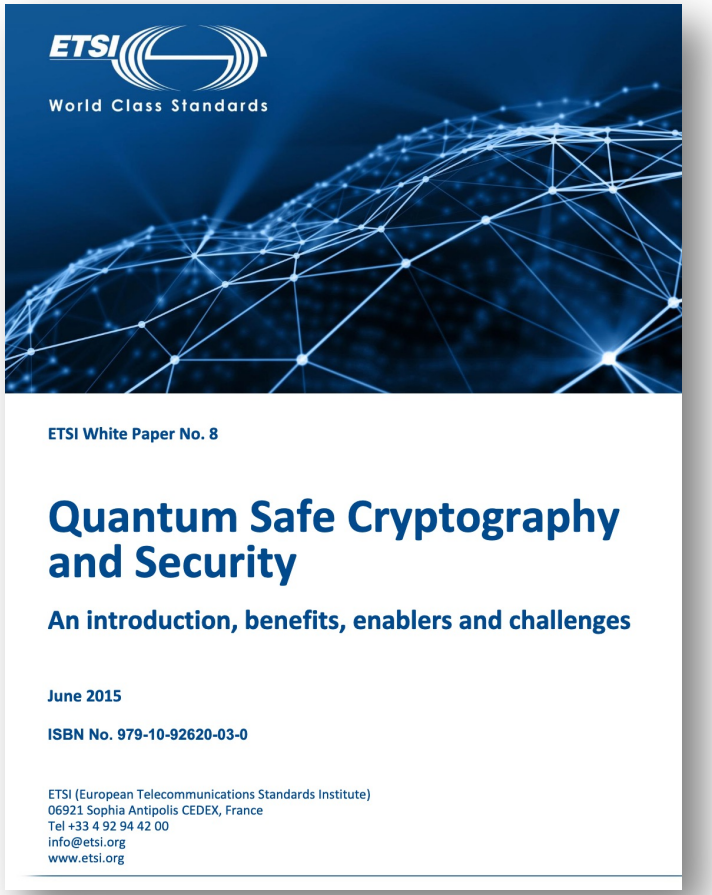
CycloneDX‘s CBOM Standard



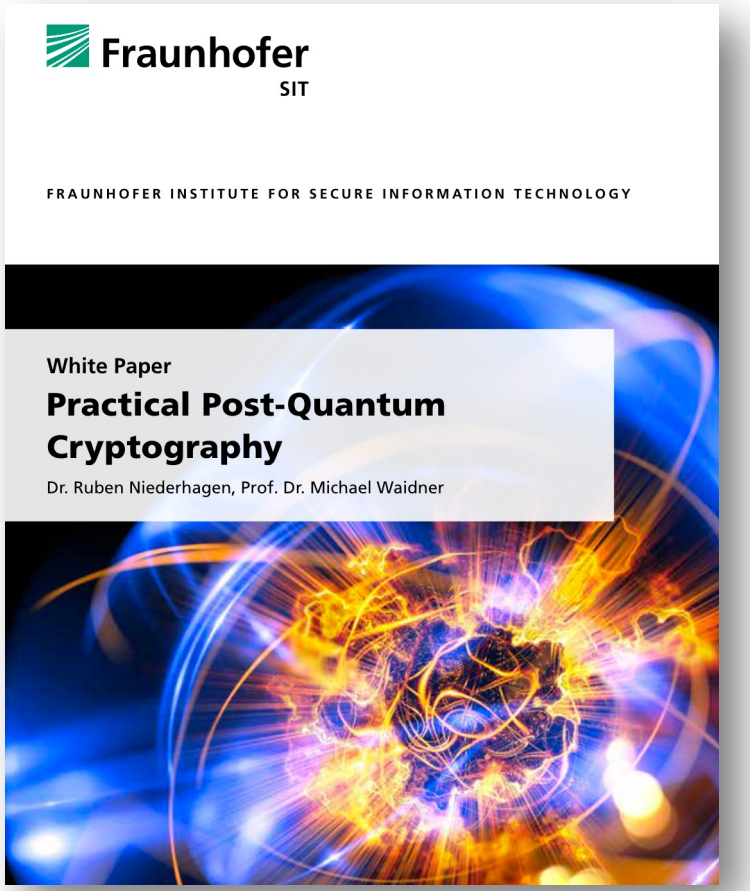
OWASP, “Authoritative Guide to CBOM: Implement Cryptography Bill of Materials for Post-Quantum Systems and Applications,” CycloneDX Feature Working Group on Cryptography, First Edition, Apr. 2024.

Appendix

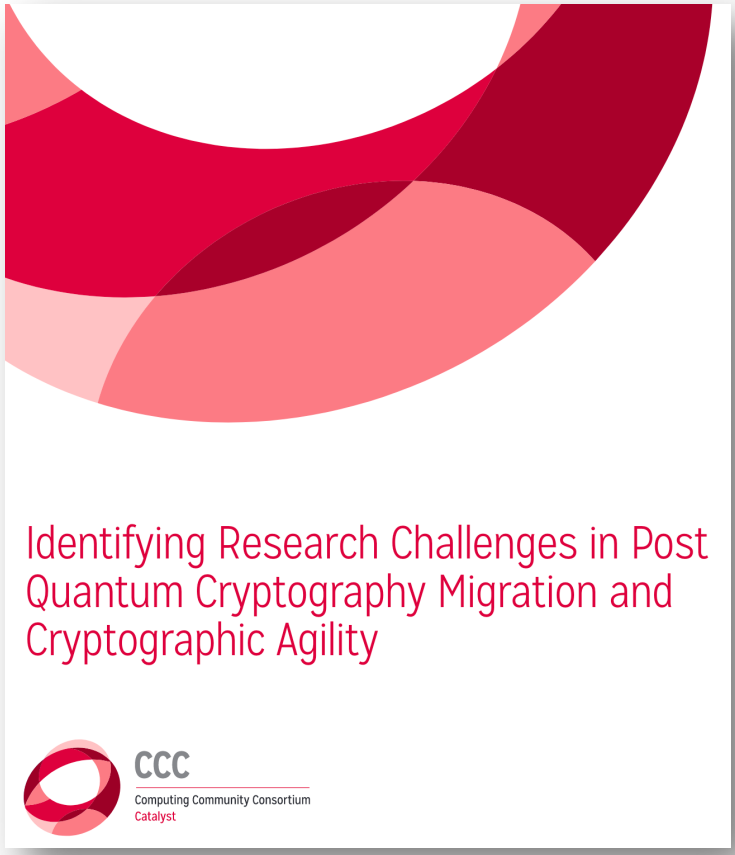
Notable Publications (1/3)



M. Campagna et al., “Quantum Safe Cryptography and Security: An introduction, benefits, enablers and challenges,” European Telecommunications Standards Institute, vol. 8, pp. 1–64, **2015**.



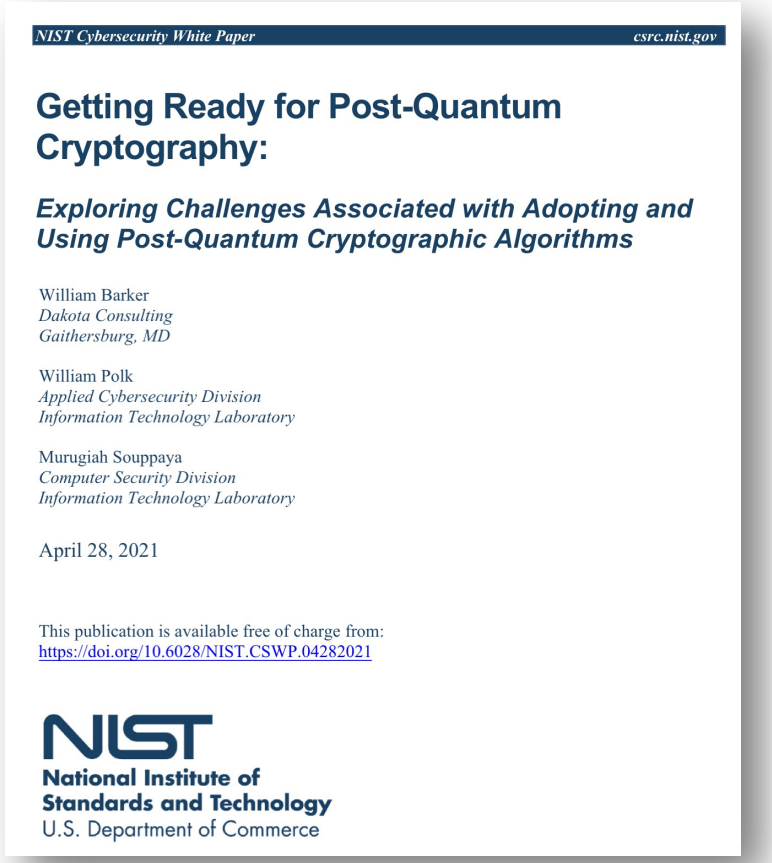
R. Niederhagen and M. Waidner, “Practical post-quantum cryptography,” Fraunhofer SIT, **2017**, [Online].



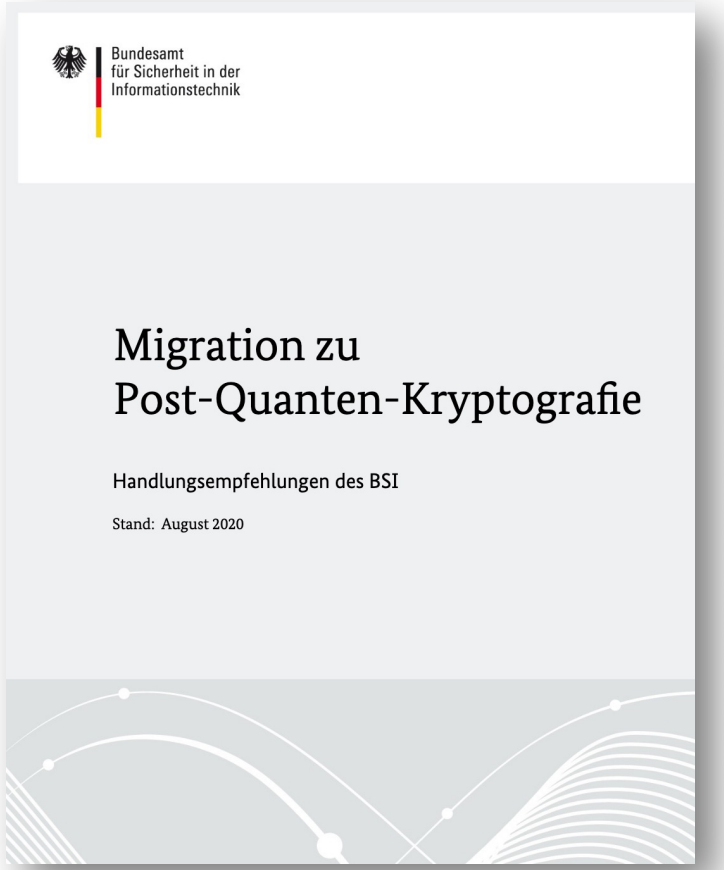
D. Ott, C. Peikert, and other workshop participants, “Identifying Research Challenges in Post Quantum Cryptography Migration and Cryptographic Agility.” arXiv, Sep. 16, **2019**. doi: 10.48550/arXiv.1909.07353.

Appendix

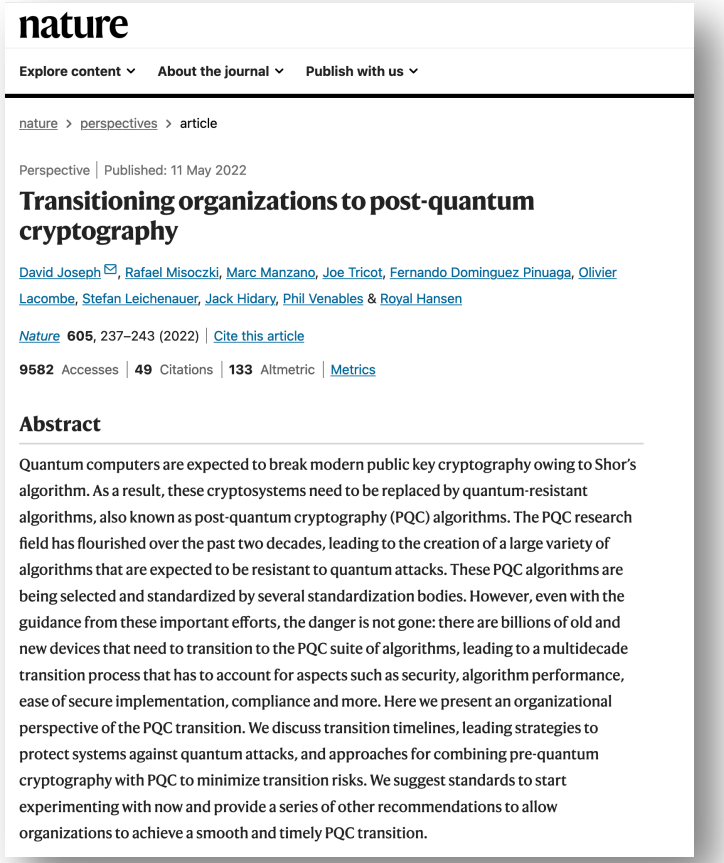
Notable Publications (2/3)



W. Barker, W. Polk, and M. Souppaya, "Getting Ready for Post-Quantum Cryptography: Explore Challenges Associated with Adoption and Use of Post-Quantum Cryptographic Algorithms," The Publications of NIST Cyber Security White Paper (DRAFT), CSRC, NIST, GOV, vol. 26, 2020, [Online].



BSI, "Migration zu Post-Quanten-Kryptografie," Bundesamt für Sicherheit in der Informationstechnik (BSI), Bonn, Aug. 2020. [Online].



D. Joseph et al., "Transitioning organizations to post-quantum cryptography," Nature, vol. 605, no. 7909, pp. 237–243, May 2022, doi: 10.1038/s41586-022-04623-2.

Appendix

Notable Publications (3/3)

- K. Petrenko, A. Mashatan, and F. Shirazi, "**Assessing the quantum-resistant cryptographic agility of routing and switching IT network infrastructure in a large-size financial organization**," *J. Inf. Secur. Appl.*, vol. 46, pp. 151–163, 2019, doi: 10.1016/J.JISA.2019.03.007.
- S. Paul and M. Niethammer, "**On the importance of cryptographic agility for industrial automation**," *Autom.*, vol. 67, no. 5, pp. 402–416, 2019, doi: 10.1515/AUTO-2019-0019.
- K. Heid, J. Heider, M. Ritscher, and J.-P. Stotz, "**Tracing cryptographic agility in android and iOS apps**," in *Proceedings of the 9th international conference on information systems security and privacy, ICISSP 2023, lisbon, portugal, february 22-24, 2023*, P. Mori, G. Lenzini, and S. Furnell, Eds., SciTePress, 2023, pp. 38–45. doi: 10.5220/0011620000003405.
- D. Ott, K. Paterson, and D. Moreau, "**Where is the research on cryptographic transition and agility?**," *Commun. ACM*, vol. 66, no. 4, pp. 29–32, 2023, doi: 10.1145/3567825.
- A. Silonosov and L. Henesey, "**Towards cryptographic agility manifesto in end-to-end encryption systems: a position paper from the perspective of crypto-consumers**," in *IEEE conference on dependable, autonomic and secure computing, DASC 2024, boracay island, philippines, november 5-8, 2024*, IEEE, 2024, pp. 65–72. doi: 10.1109/DASC64200.2024.00015.
- J. Cho, C. Lee, E. Kim, J. Lee, and B. Cho, "**Software-defined cryptography: A design feature of cryptographic agility**," *CoRR*, vol. abs/2404.01808, 2024, doi: 10.48550/ARXIV.2404.01808.

Appendix

Terminology

Cryptographic Agility (crypto-agility)	Ability to switch between multiple cryptographic primitives, enabling rapid adaptations of new cryptographic algorithms without making disruptive system changes.	[0]
Software Bill of Materials (SBOM)	Object model to what is inside a software. Helps stakeholders to keep track of software components and dependencies for better software supply chain security.	[1]
Open Worldwide Application Security Project (OWASP)	Non-profit foundation that works to improve the security of software.	[2]
CycloneDX	A standardized and machine-readable format for capturing the components that comprise software (SBOM), hardware (HBOM), services (SaaSBOM), AI/ML models (AI/ML-BOM), and cryptography (CBOM).	[3] [4]
Cryptography Bill of Materials (CBOM)	Object model to describe cryptographic assets and their dependencies. CBOM is an extension of the CycloneDX standard for SBOMs	[5] [6]

References

Additional references are cited in-line at the relevant slides.

[0] “Cryptographic agility,” Wikipedia. Dec. 30, 2023. Accessed: Apr. 03, 2024. [Online]. Available: https://en.wikipedia.org/w/iCryptographic_agility

[1] J. Fruhlinger, “What is an SBOM? Software bill of materials explained,” CSO Online. Accessed: Apr. 10, 2024. [Online]. Available: https://www.csoonline.com/article/573185/what-is-an-sbom-software-bill-of-materials-explained.html

[2] OWASP Foundation. Accessed: Apr. 10, 2024. [Online]. Available: https://owasp.org/about/

[3] S. Springett, “CycloneDX v1.6 Released, Advances Software Supply Chain Security with Cryptographic Bill of Materials and Attestations,” OWASP. Accessed: Apr. 10, 2024. [Online]. Available: https://owasp.org/blog/2024/04/09/CycloneDX-v1.6-Released.html

[4] “CycloneDX Specification Overview.” Accessed: Apr. 10, 2024. [Online]. Available: https://cyclonedx.org/specification/overview/

[5] A. Curioni and M. Osborne, “IBM’s Cryptography Bill of Materials to speed up quantum-safe assessment,” IBM Research. Accessed: Apr. 10, 2024. [Online]. Available: https://research.ibm.com/blog/cryptographic-bill-of-materials

[6] IBM, CBOM. International Business Machines, 2024. Accessed: Apr. 10, 2024. [Online]. Available: https://github.com/IBM/CBOM

[7] P. W. Shor, “Algorithms for Quantum Computation: Discrete Logarithms and Factoring,” in Proceedings 35th Annual Symposium on Foundations of Computer Science, IEEE Comput. Soc. Press, 1994, pp. 124–134. doi: 10.1109/SFCS.1994.365700.