

TOOLS PRESENTATION

LIBOQS (OPEN SAFE)

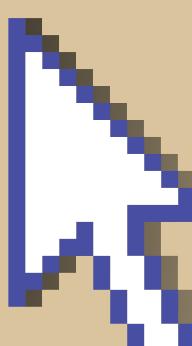
RAGHADE NAWAR

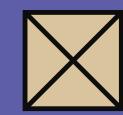


C LIBRARY FOR PROTOTYPING AND
EXPERIMENTING WITH QUANTUM-RESISTANT
CRYPTOGRAPHY



QUANTUM





PRES

ENTATION OUTLINE

1. What is liboqs?
2. Installation & Setup
3. Cloud/Open Source/Freeware Status
4. Community & Support Forums
5. Who is Using liboqs?
6. Comparison with Similar Tools
7. Tools: liboqs Live Demonstration
8. Conclusion & Resources





TOOL OVERVIEW

What is liboqs?

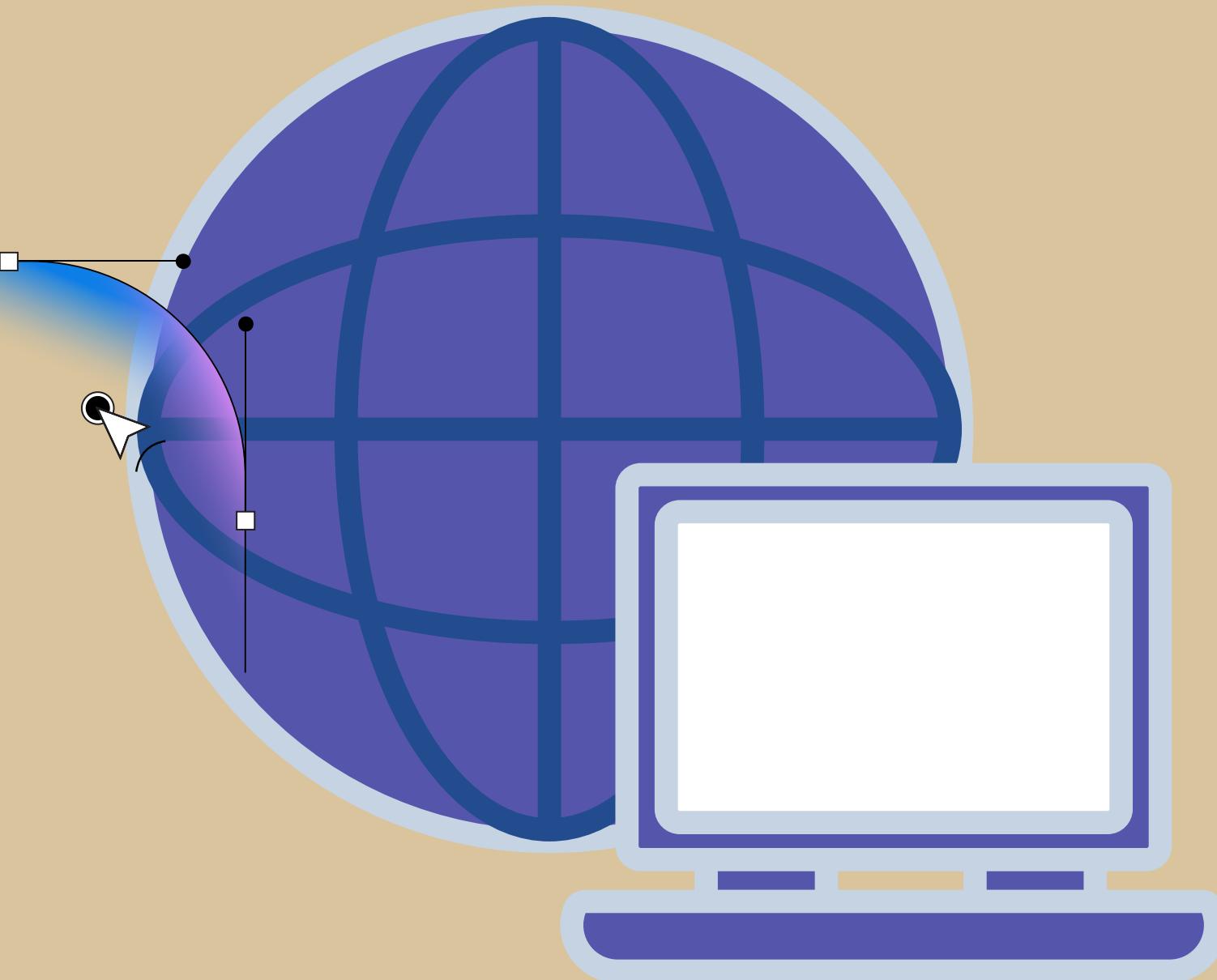
liboqs is an open-source C library for quantum-resistant cryptographic algorithms.

Full Name: Open Quantum Safe - liboqs

Purpose: Provide production-ready implementations of post-quantum cryptographic algorithms

Project: Part of the Open Quantum Safe (OQS) project





HOW TO GET/INSTALL THE TOOL

Installation Methods

1. From Source (Linux/macOS)

```
```bash
Clone the repository
git clone -b main https://github.com/open-quantum-safe/liboqs.git
cd liboqs

Create build directory
mkdir build && cd build

Configure with CMake
cmake -DCMAKE_INSTALL_PREFIX=/usr/local ..

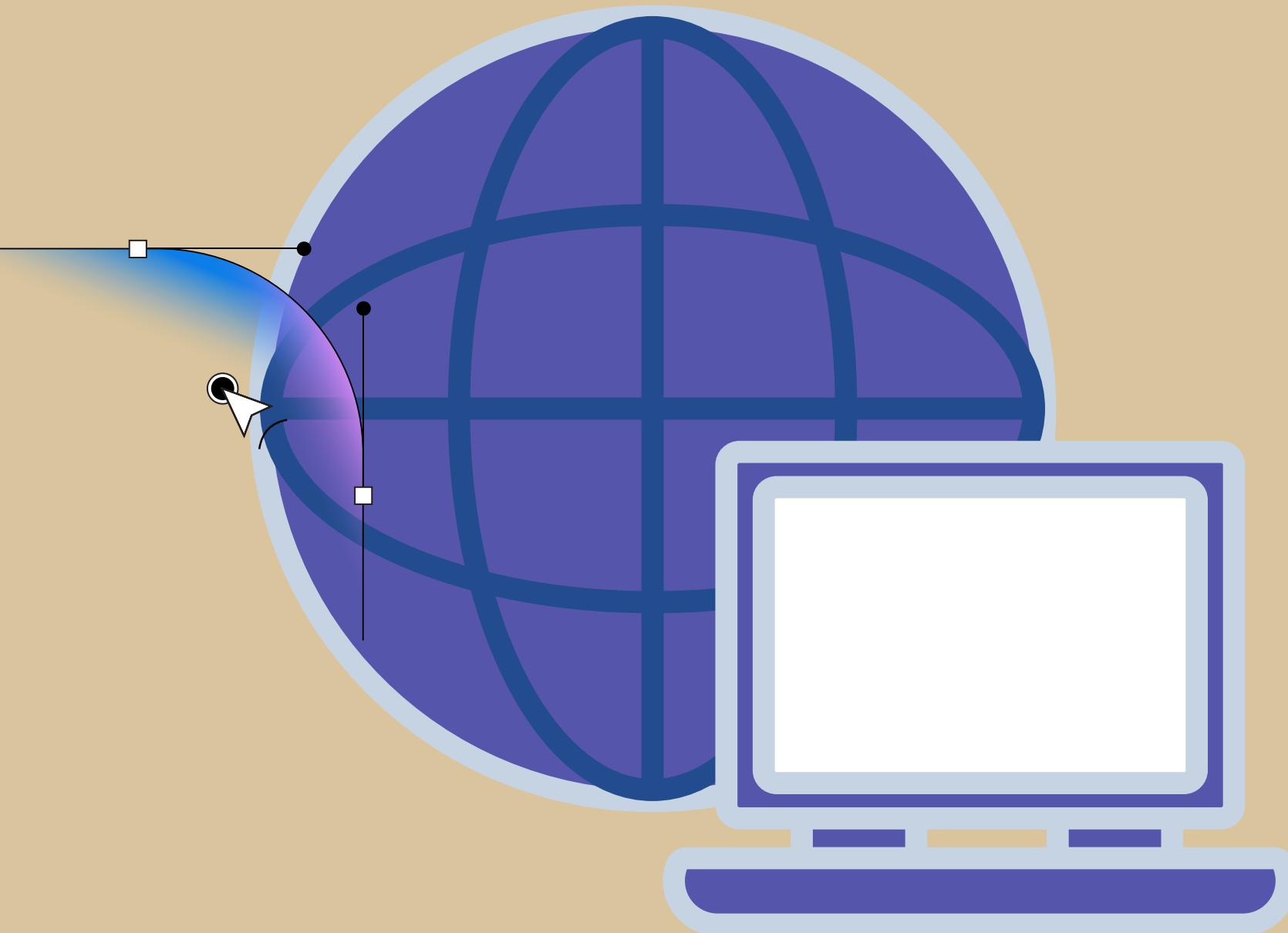
Build
make -j$(nproc)

Install (may need sudo)
sudo make install
````
```





HOW TO GET/INSTALL THE TOOL



Installation Methods

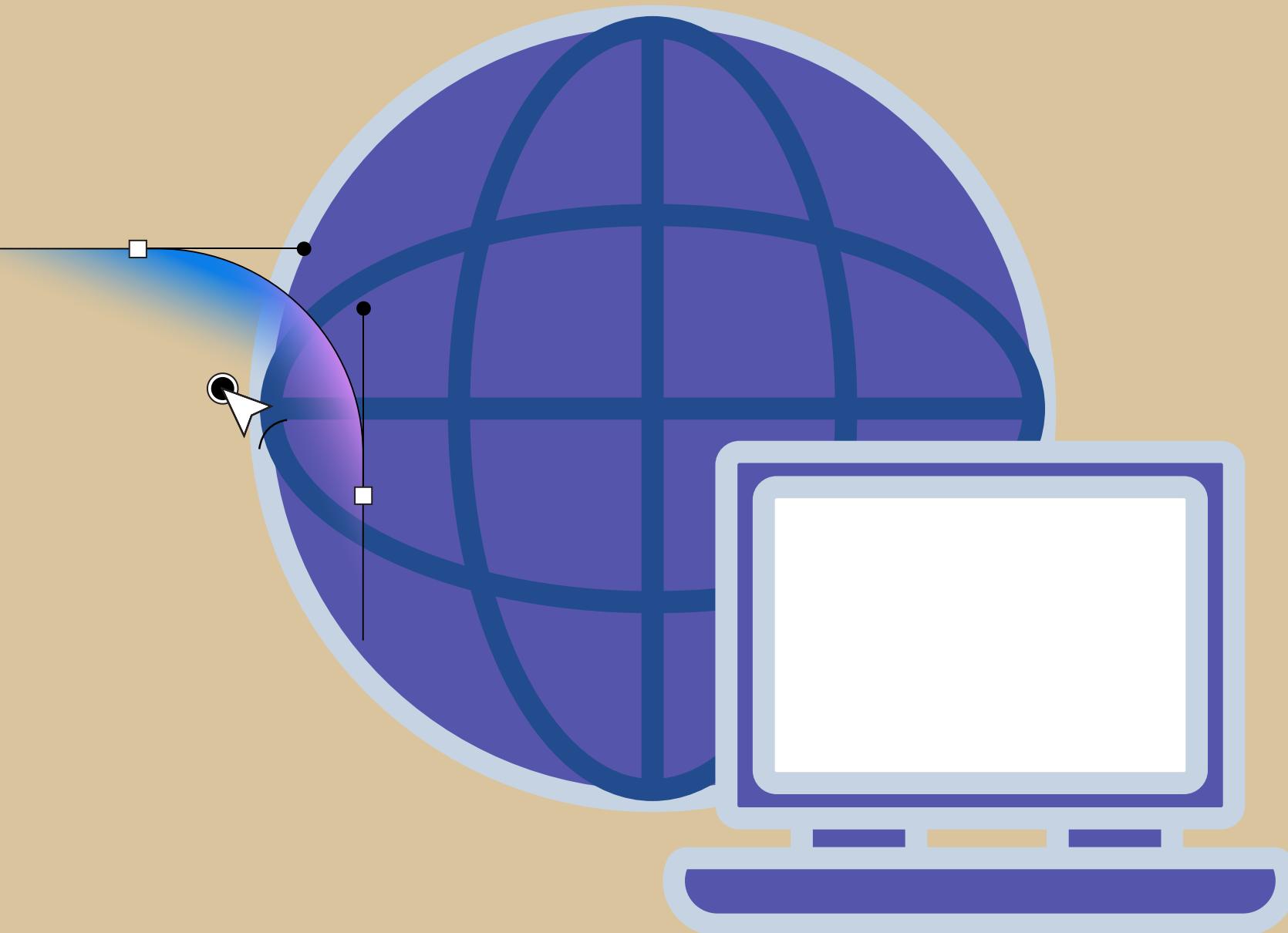
2. Using Package Managers
Ubuntu/Debian

```
``` bash
Add OQS repository (if available)
sudo apt-get update
sudo apt-get install liboqs-dev
```
```





HOW TO GET/INSTALL THE TOOL

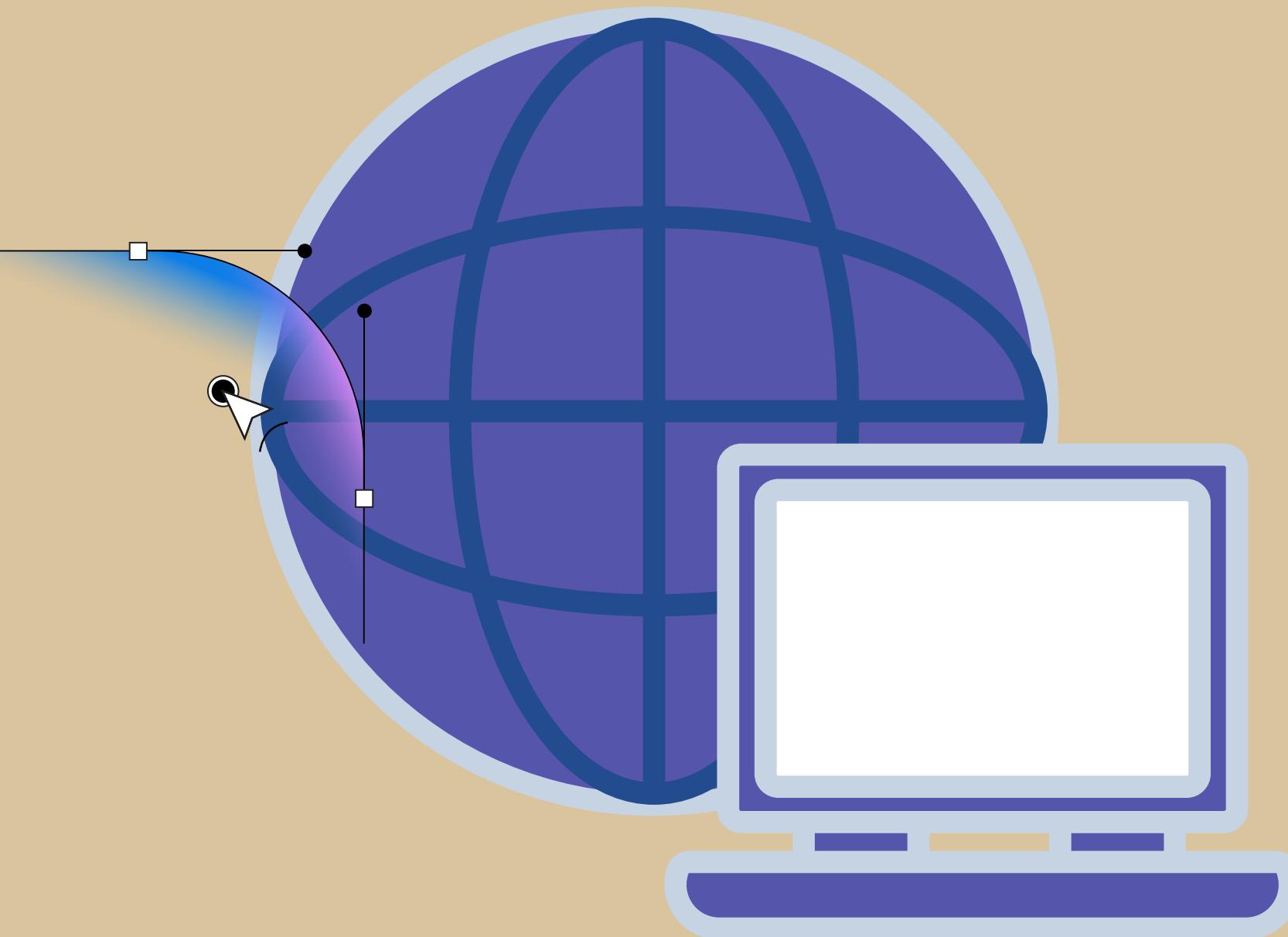


Installation Methods

2. Using Package Managers
macOS (Homebrew)

```
...  
bash  
brew tap open-quantum-safe/liboqs  
brew install liboqs  
...
```





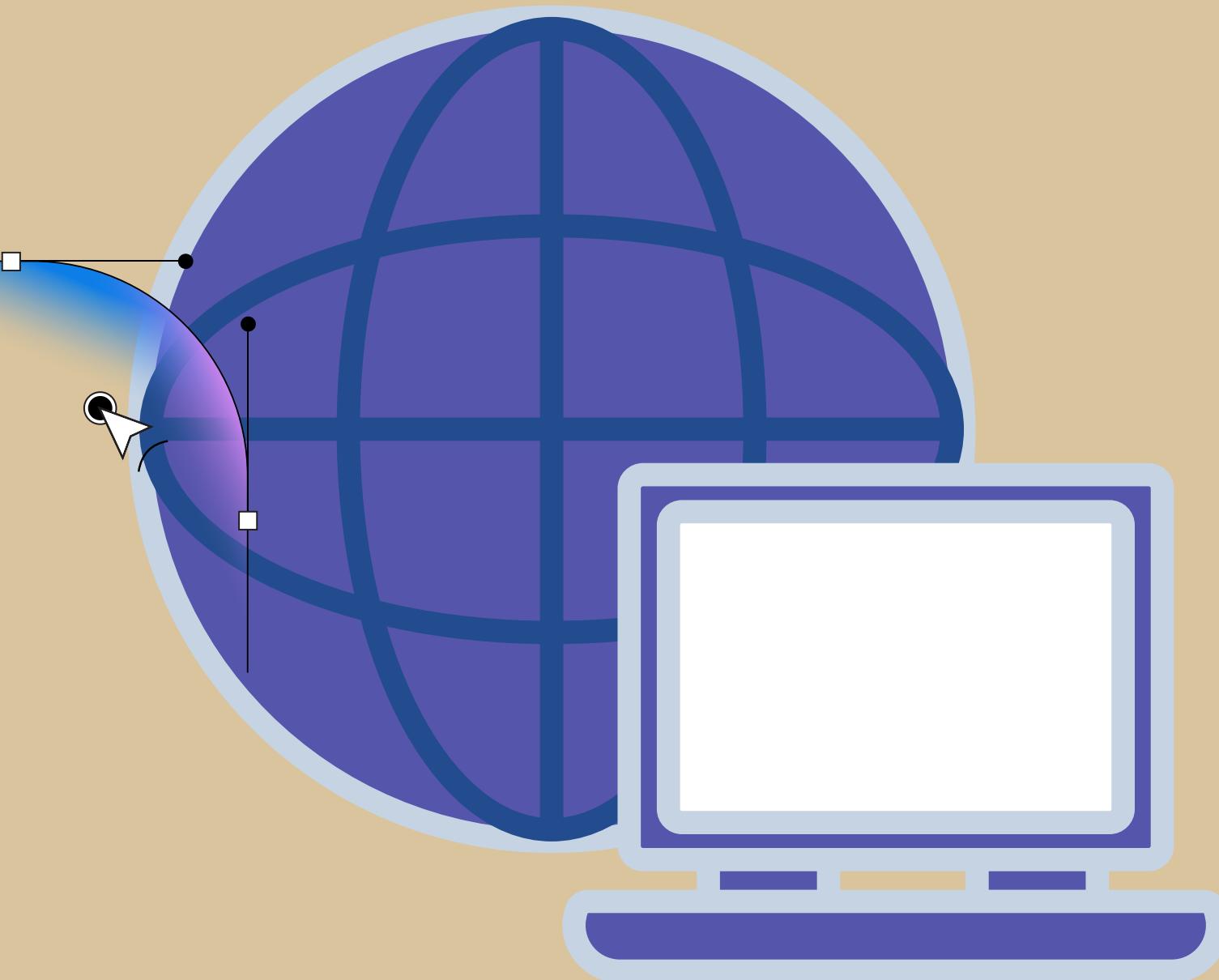
HOW TO GET/INSTALL THE TOOL

Installation Methods

2. Using Package Managers
Windows (vcpkg)

```
...  
bash  
vcpkg install liboqs  
...
```





HOW TO GET/INSTALL THE TOOL

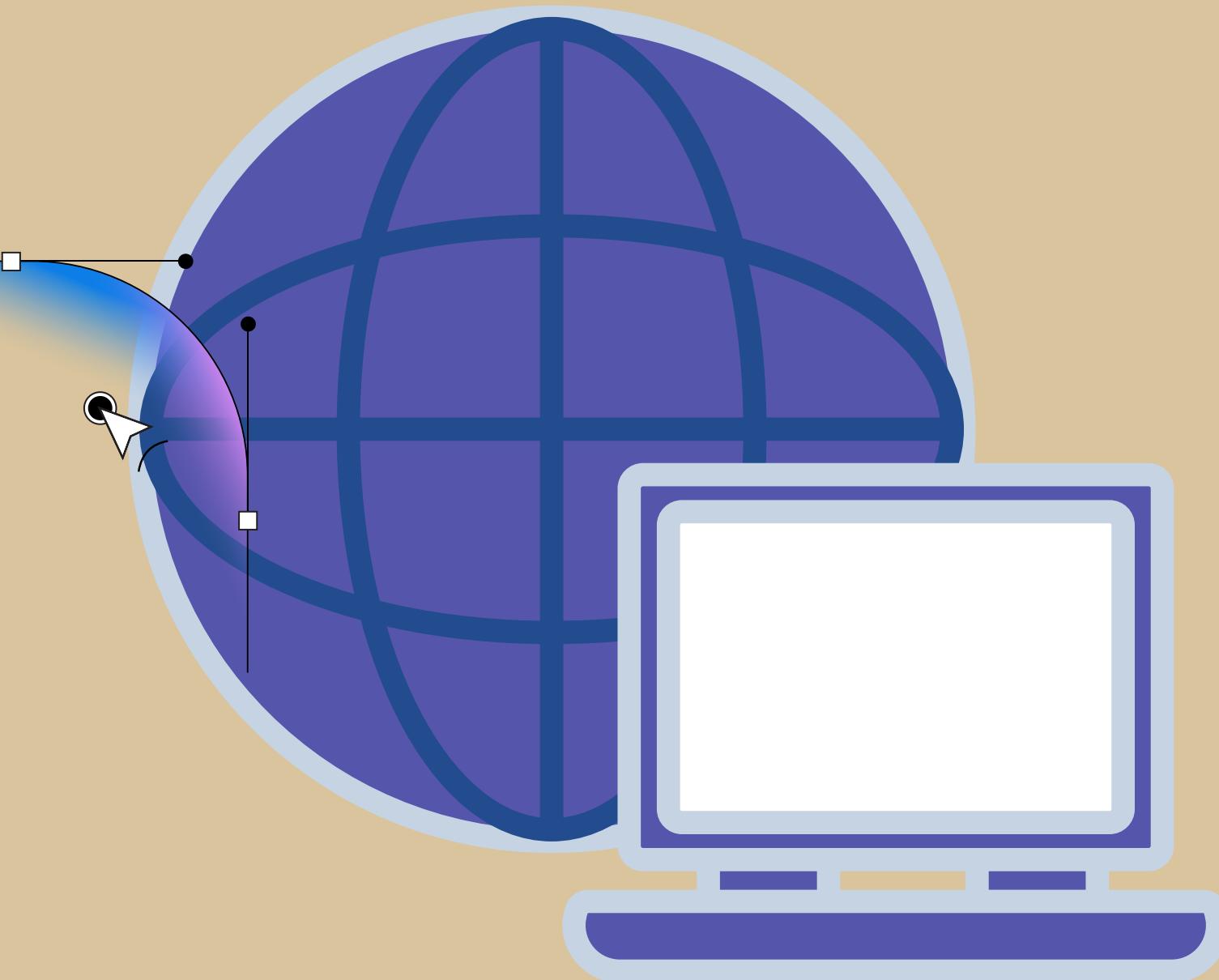
Installation Methods

3. Using Docker

```
```bash
Pull official Docker image
docker pull openquantumsafe/liboqs

Run interactive container
docker run -it openquantumsafe/liboqs /bin/bash
```
```





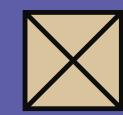
HOW TO GET/INSTALL THE TOOL

Installation Methods

4. Python Bindings

```
```bash
Install liboqs-python
pip install liboqs-python
```
```





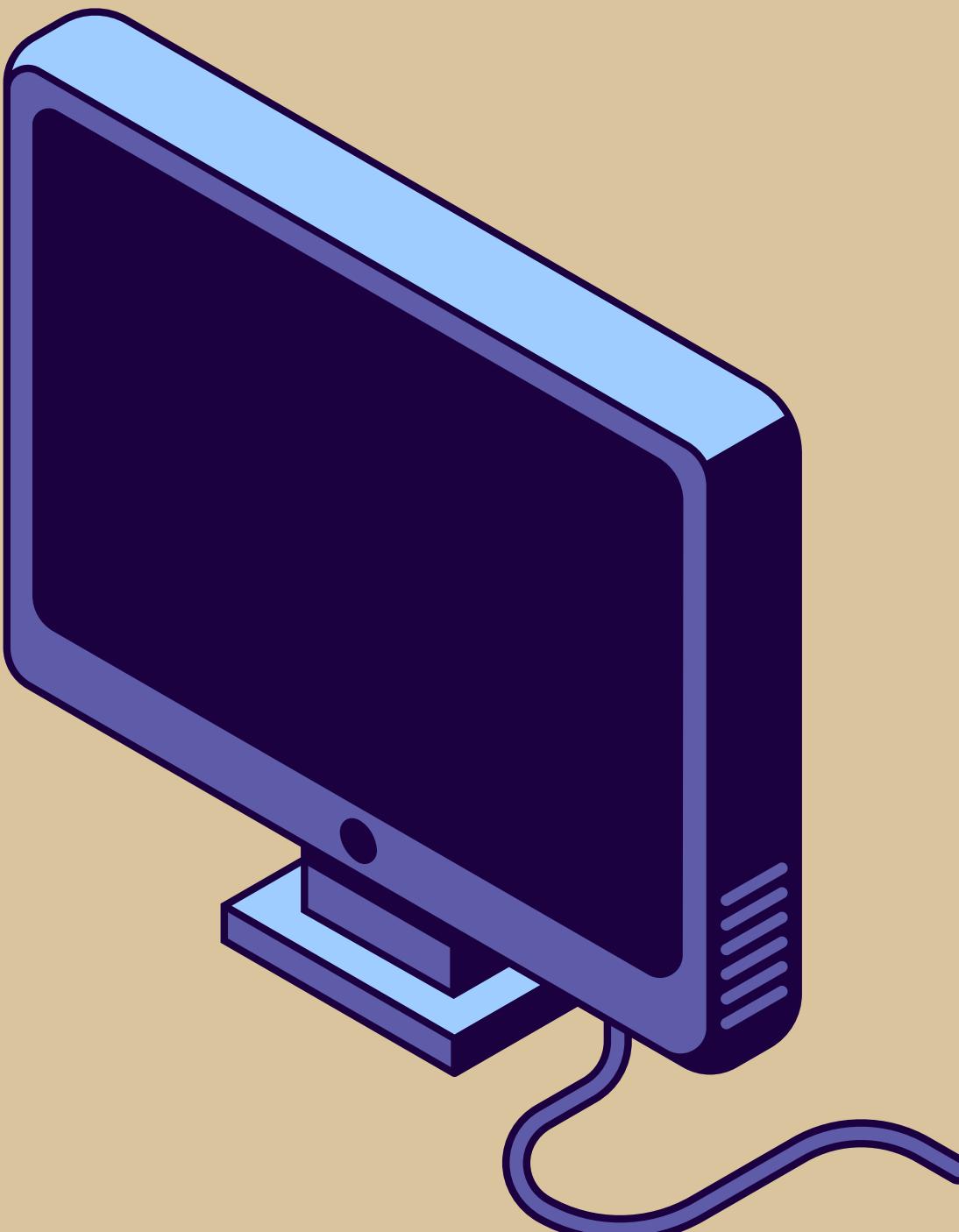
BUILD OPTIONS

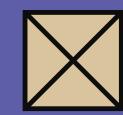
Configuration Flags

```
```bash
Enable all algorithms
cmake -DOQS_ENABLE_KEM_KYBER=ON \
 -DOQS_ENABLE_SIG_DILITHIUM=ON \
 -DOQS_ENABLE_SIG_FALCON=ON \
 -DOQS_ENABLE_SIG_SPHINCS=ON ...

Disable some algorithms to reduce size
cmake -DOQS_MINIMAL_BUILD=ON ...

Enable OpenSSL integration
cmake -DOQS_USE_OPENSSL=ON ...
````
```





CLOUD/OPEN SOURCE/FREEWWARE STATUS



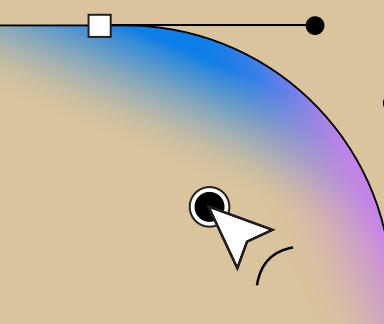
LICENSE

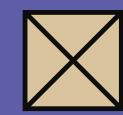
- **Type:** MIT License
- **Status:** Completely FREE and Open Source
- **Commercial Use:** Allowed
- **Modification:** Allowed
- **Distribution:** Allowed



AVAILABILITY

- **Source Code:** GitHub (public repository)
- **Binary Packages:** Available for major platforms
- **Cloud Ready:** Can be deployed on any cloud platform
- **No Cost:** Completely free to use





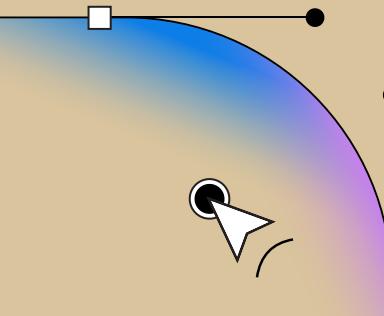
LIBOQS (OPEN QUANTUM SAFE)

CLOUD/OPEN SOURCE/FREEWWARE STATUS



REPOSITORY

- **URL:** <https://github.com/open-quantum-safe/liboqs>
- **Stars:** 1.8k+ (as of Dec 2025)
- **Forks:** 450+
- **Active Development:** Regular updates and releases



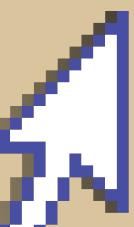


CURRENT SUPPORT FORUMS/GROUPS



Official Support Channels

1. GitHub
 - Issues: <https://github.com/open-quantum-safe/liboqs/issues>
 - Discussions: <https://github.com/open-quantum-safe/liboqs/discussions>
 - Pull Requests: Active community contributions
 - Response Time: Usually within 24-48 hours
 2. Documentation
 - Main Docs: <https://openquantumsafe.org/>
 - API Reference: <https://openquantumsafe.org/liboqs/>
 - Wiki: Comprehensive guides and tutorials
 2. Stack Overflow
 - Tag: [\[post-quantum-cryptography\]](#), [\[liboqs\]](#)
 - Community: Growing PQC community
 - Questions: ~100+ related questions
- Community Size
- Contributors: 50+ active contributors
 - Organizations: Supported by industry and academia
 - Users: Thousands of downloads per month
 - Global: Worldwide community





WHO IS UTILIZING IT AND HOW

Industry

1. Cloud Providers

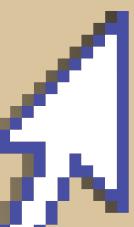
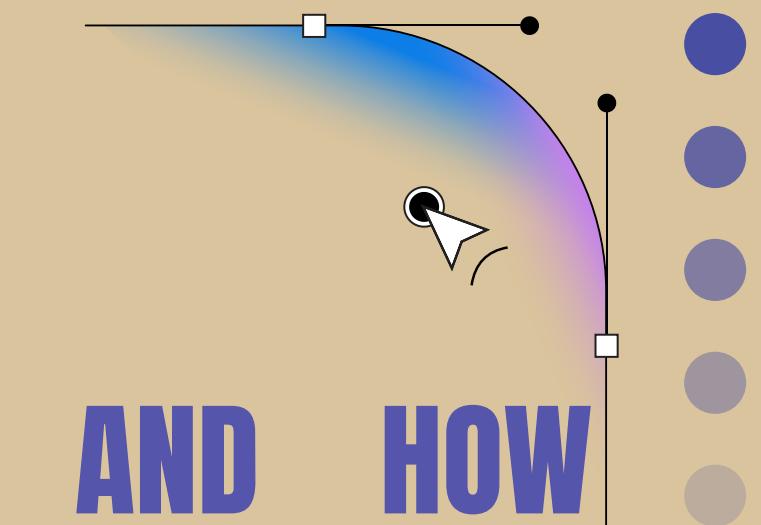
- Cloudflare: Uses OQS for HTTPS experiments
- AWS: Testing PQC integration
- Microsoft: Azure experiments with PQC

2. Software Companies

- Mozilla Firefox: PQC testing
- Google Chrome: CECPQ experiments
- Apple: iOS/macOS security research

3. Security Companies

- Cisco: Network equipment PQC
- Fortinet: Firewall PQC support
- Palo Alto Networks: Security appliances





LIBOQS (OPEN QUANTUM SAFE)



WHO IS UTILIZING IT AND HOW

Academia

Leading Universities

- MIT, Stanford, UC Berkeley
- KU Leuven, TU Eindhoven
- Waterloo (Institute for Quantum Computing)
- Used in research and teaching



TOOLS PRESENTATION





LIBOQS (OPEN QUANTUM SAFE)



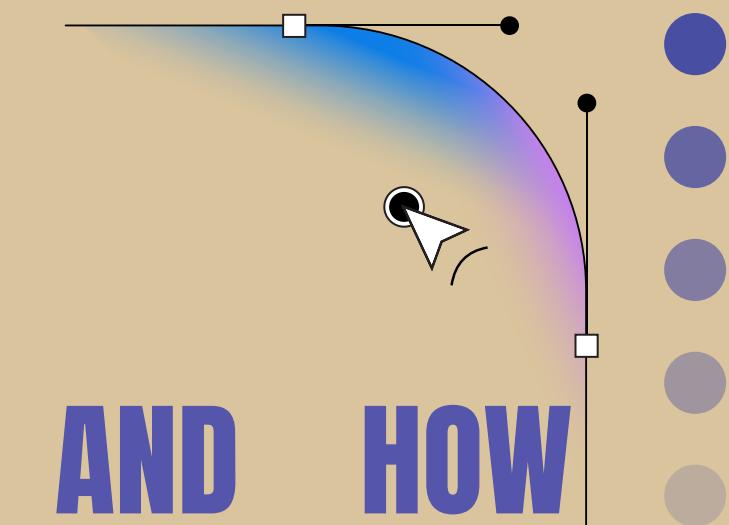
WHO IS UTILIZING IT AND HOW

Government Agencies

- NIST: Reference for PQC standardization
- NSA: Evaluation and testing
- European Commission: PQCRYPTO project
- Various national cyber agencies



TOOLS PRESENTATION





LIBOQS (OPEN QUANTUM SAFE)



WHO IS UTILIZING IT AND HOW

Fame and Recognition Statistics

- GitHub Stars : 1,800+ (Top PQC library)
- Downloads: 50,000+ monthly
- Citations: 200+ academic papers
- Integrations: 30+ projects using liboqs

Awards and Recognition

- Part of NIST PQC standardization efforts
- Featured in major security conferences - Recommended by industry standards bodies
- Widely cited in academic literature



TOOLS PRESENTATION





COMPARISON WITH SIMILAR TOOLS



vs. PQClean

- liboqs: More comprehensive, production-ready
- PQClean: Focus on clean, portable code
- Winner: liboqs for deployment, PQClean for reference

vs. Bouncy Castle PQC

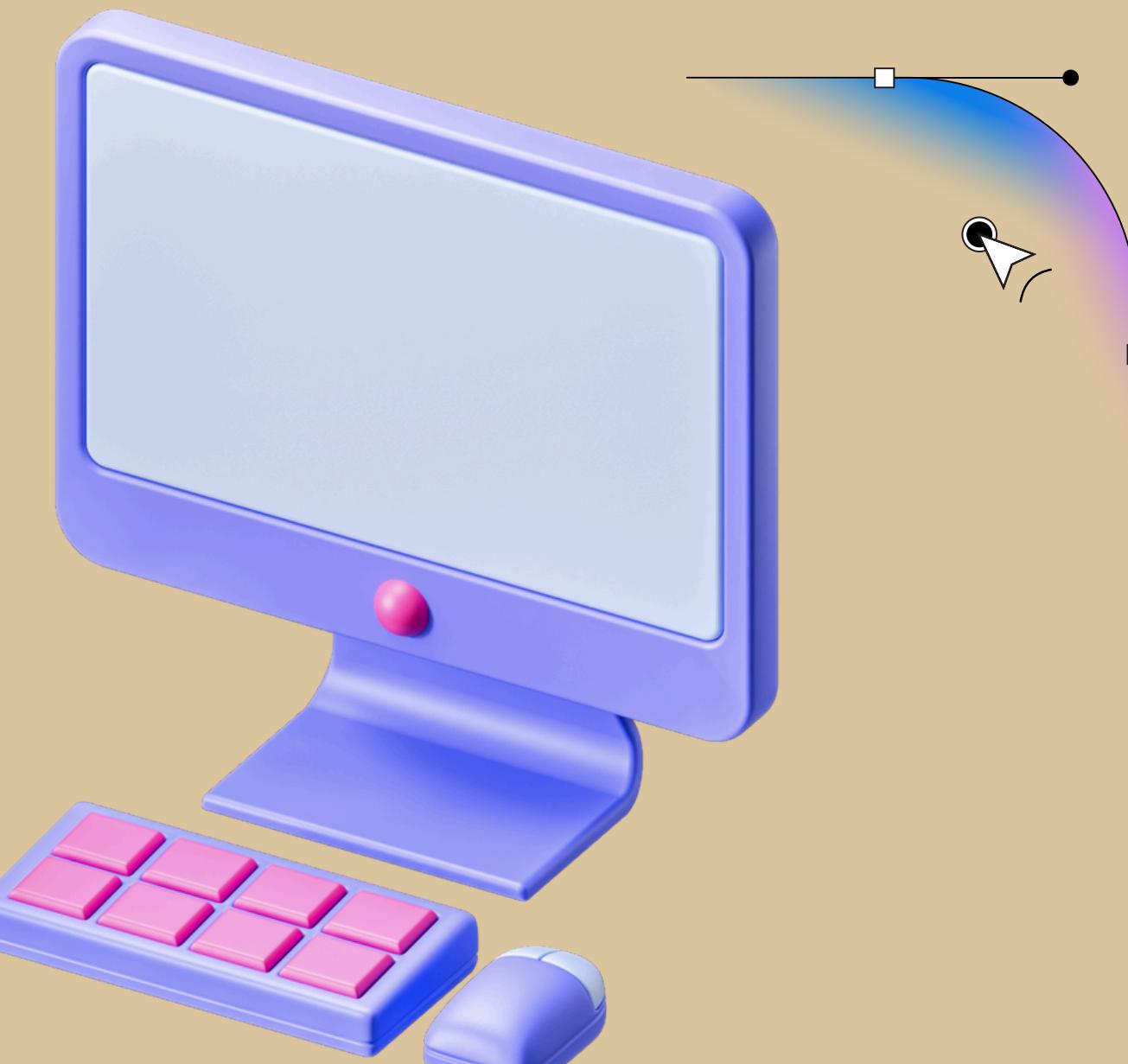
- liboqs: C library, performance-focused
- Bouncy Castle: Java, easier integration
- Winner: Depends on language preference

vs. OpenSSL PQC Fork

- liboqs: Standalone, algorithm-focused
- OpenSSL: Full SSL/TLS stack
- Winner: liboqs for flexibility, OpenSSL for protocols



COMPARISON WITH SIMILAR TOOLS



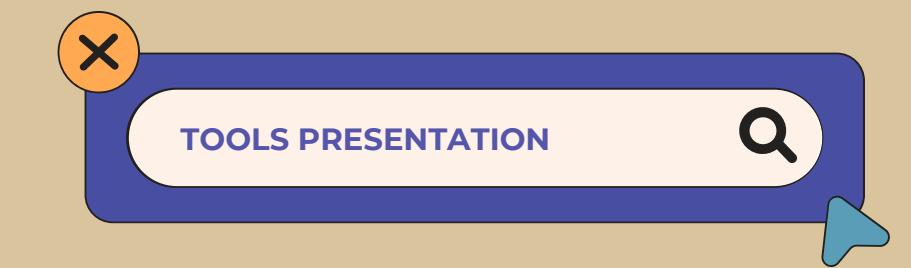
vs. libsodium

- liboqs: PQC-focused
- libsodium: Classical crypto, easy to use
- Winner: Different purposes (PQC vs classical)

Industry Adoption Level

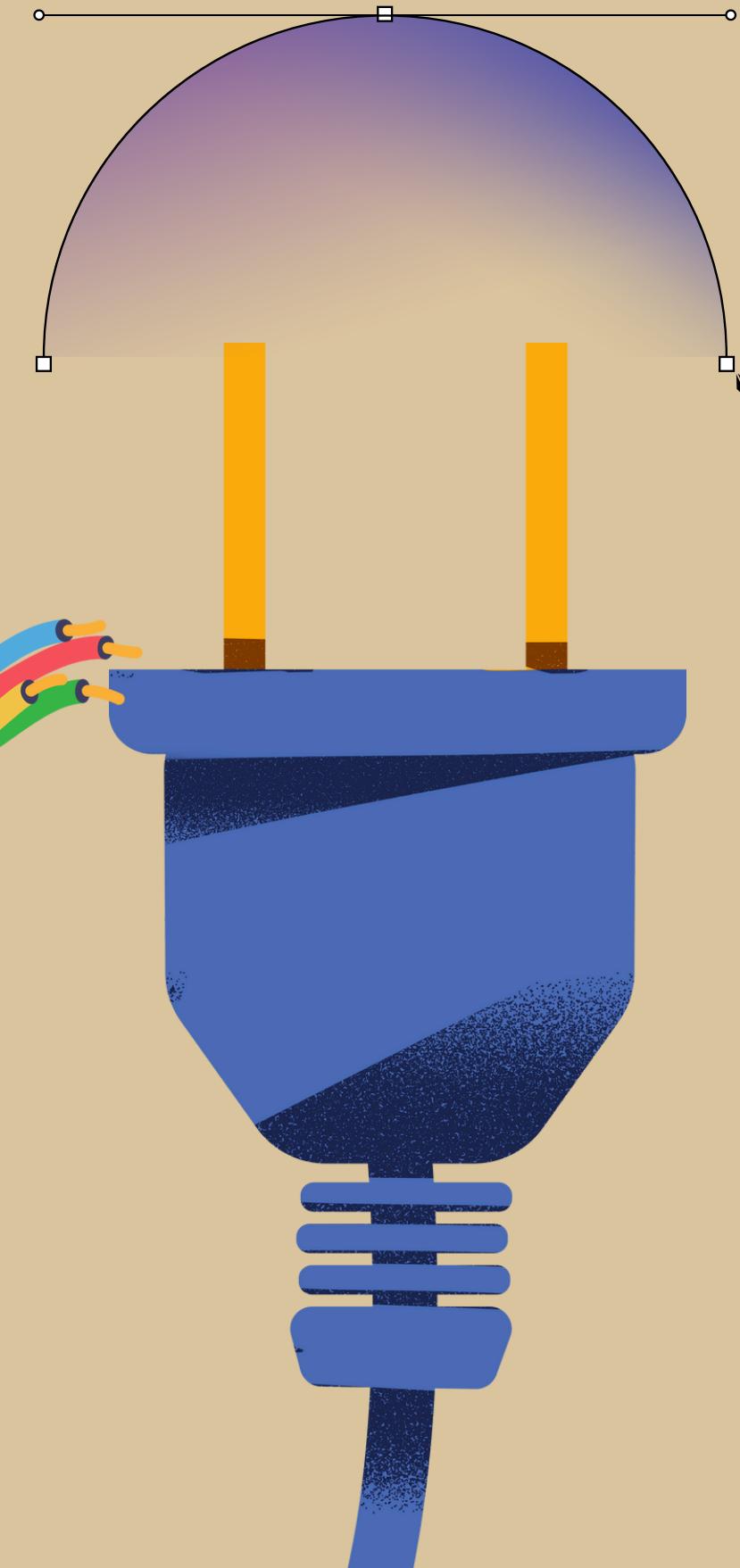
Status: LEADING PQC LIBRARY

- Most widely deployed open-source PQC library
- Industry standard for PQC integration
- Active development and maintenance
- Strong community support





LIBOQS (OPEN QUANTUM SAFE)



AVAILABLE ALGORITHMS

Key Encapsulation Mechanisms (KEMs)

- CRYSTALS-Kyber: Kyber512, Kyber768, Kyber1024
- BIKE: BIKE-L1, BIKE-L3, BIKE-L5
- Classic McEliece: Multiple parameter sets
- HQC: HQC-128, HQC-192, HQC-256
- NTRU: NTRU-HPS, NTRU-HRSS
- SABER: LightSaber, Saber, FireSaber



TOOLS PRESENTATION





LIBOQS (OPEN QUANTUM SAFE)



AVAILABLE ALGORITHMS

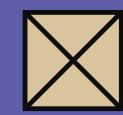
Digital Signatures

- CRYSTALS-Dilithium: Dilithium2, Dilithium3, Dilithium5
- Falcon: Falcon-512, Falcon-1024
- SPHINCS+: Multiple parameter sets
- Picnic: Various levels
- Rainbow: Multiple levels



TOOLS PRESENTATION



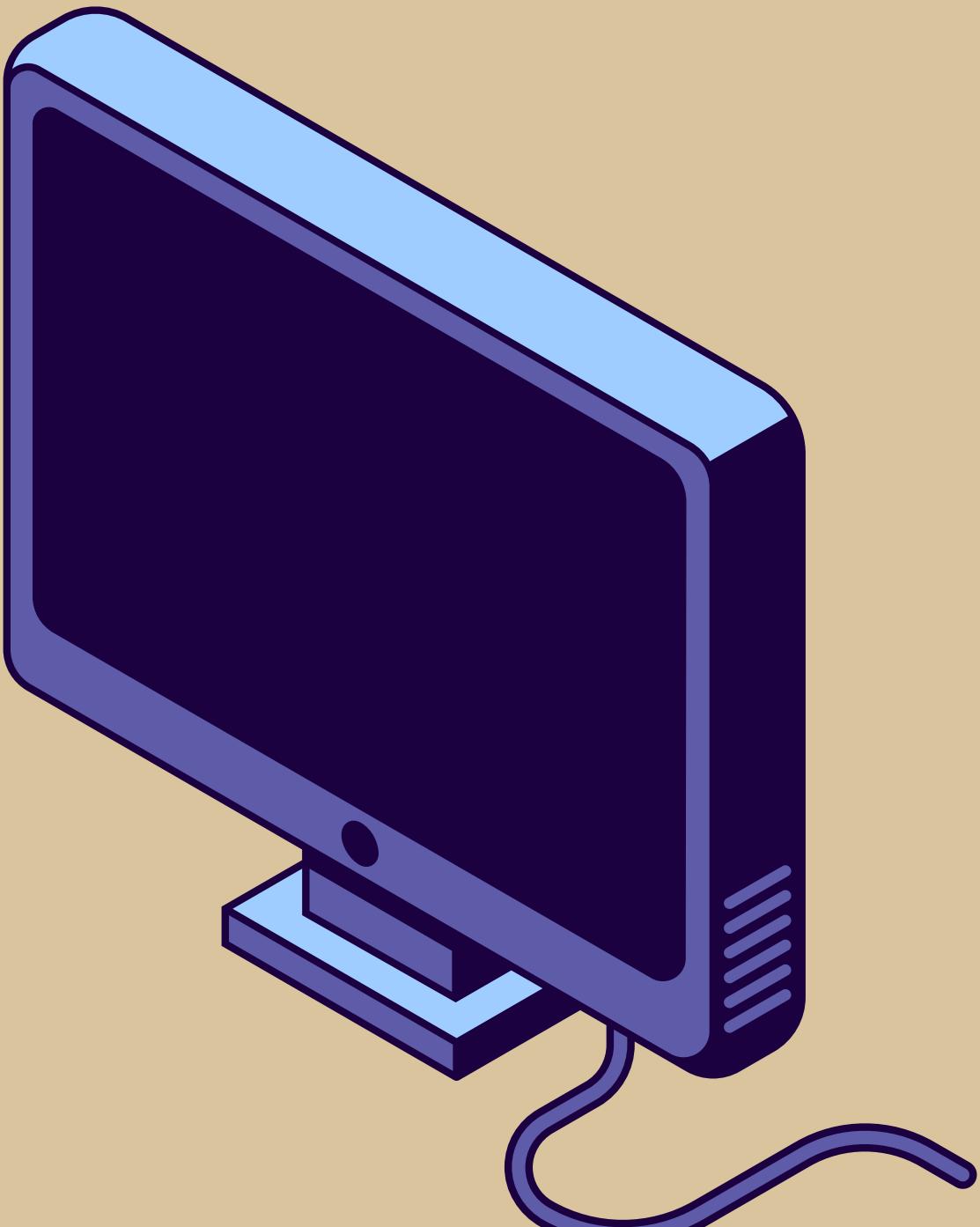


LIBOQS (OPEN QUANTUM SAFE)

TOOLS: LIBOQS LIVE DEMONSTRATION

The link of Colab notebook:

<https://colab.research.google.com/drive/1OGYULoYvtzcmPWfe7MaBbqfmCN6Mtl8A?usp=sharing>



TOOLS PRESENTATION





RESOURCES

Official Links

- Website: <https://openquantumsafe.org/>
- GitHub: <https://github.com/open-quantum-safe/liboqs>
- Documentation: <https://openquantumsafe.org/liboqs/>
- Wiki: <https://github.com/open-quantum-safe/liboqs/wiki>

Tutorials

- [Getting Started Guide](#)
- [API Reference \(Doxygen\)](#)
- [Integration Examples](#)
- [Performance Tuning](#)





RESOURCES

Official Links

Community

- [GitHub Discussions](#)
- [Stack Overflow](#)

Related Projects

- [oqs-openssl: OpenSSL with PQC](#)
- [oqs-openssh: SSH with PQC](#)
- [oqs-provider: OpenSSL 3.0 provider](#)
- [oqs-demos: Example applications](#)





CONCLUSION

Summary

- ✓ liboqs is the leading open-source PQC library
- ✓ Easy to install and use (multiple platforms)
- ✓ Completely free and open source (MIT licence)
- ✓ Excellent community support (GitHub, Slack, mailing list)
- ✓ Widely used (industry, academia, government)
- ✓ Perfect for research (reproducibility, comprehensiveness, trusted)
- ✓ Production-ready (security audited, optimized, maintained)



THANK YOU!

