Post-Quantum Cryptography (PQC) Network Instrument:

Measuring PQC Adoption Rates and Identifying Migration Pathways

Phuong Cao

Research Scientist
National Center for Supercomputing Applications
University of Illinois at Urbana-Champaign

Team:

Jakub Sowa, Bach Hoang, Advaith Yeluru, Steven Qie, Santiago Nunez Corrales, Anita Nikolich, Ravishankar Iyer

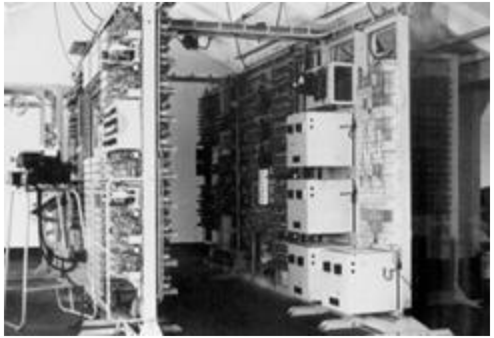
Classical Cryptography

Microchip Transistor Counts
(Impacts CPU speed)

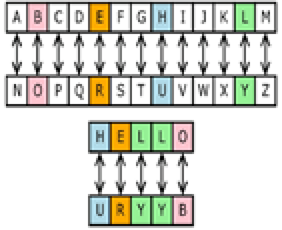
Classical Cryptography Timeline

1918: German Enigma cipher created
1932: first cracking of Enigma cipher in Poland
1940: "Bombe" machine built by Turing to speed up Allied Enigma codebreaking

1940: German Lorenz cipher created
1943: first electric programmable computer, "Colossus," built to break Lorenz cipher



Your String	Secret Message
MD5 Hash	54b0581b96c2e04197a9f4f50f1e64da Copy
SHA1 Hash	dfa5d7e19ad72db52fa54f6806a5831ac661f927



10^{10}

10^{9}

10^{6}

10^{3}

Past - 1800s: simple Substitution ciphers like the Caesar cipher dominate

1976: Diffie-Hellman key exchange
1977: RSA public key encryption

1985: ECC key exchange proposed.

1994: Secure Socket Layer (SSL) for network security released by Netscape

2001: US government adopts an Advanced Encryption Standard (AES)

2004: MD5 hash proven not secure
2005: SHA1 hash proven not secure

2010s: all versions of SSL considered not secure and deprecated in favor of TLS

Future: latest TLS version 1.3 reaches 100% adoption

Year

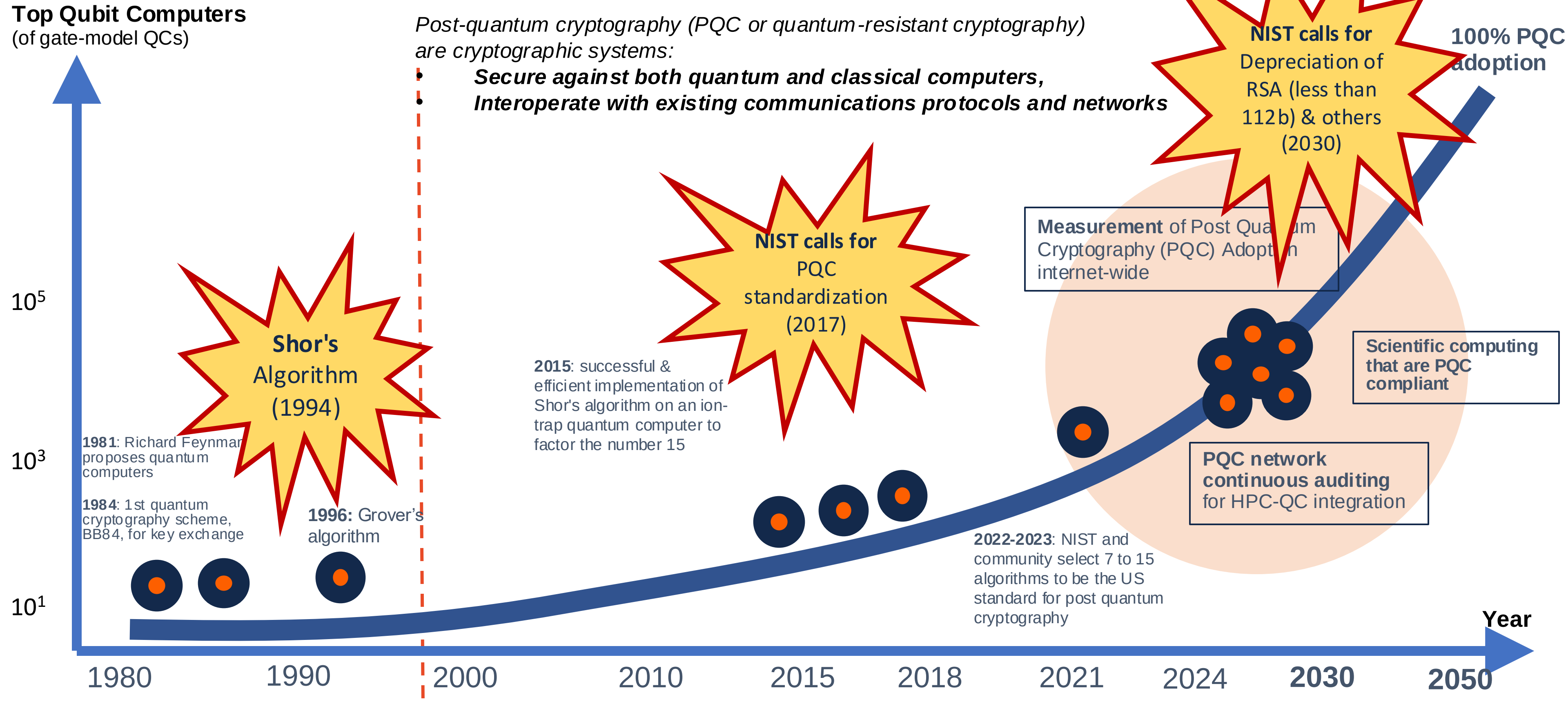
0 (AD) ... 1940 1960 1980 2000 2020 **2050**

Classical Computers

Quantum-Resistant Cryptography

Post-quantum cryptography (PQC or quantum-resistant cryptography) are cryptographic systems:

- ***Secure against both quantum and classical computers,***
- ***Interoperate with existing communications protocols and networks***



Quantum Cryptography timeline

* This is a quantum annealing QC which is not a universal QC, primarily solving optimization problems, and could not perform Shor's algorithm.

Summary and Key Takeaways

Problems

- **Inadequate guidance on migrating HPC cyberinfrastructure to be compliant.**

Standards: Key Encapsulation and Digital Signature

CRYSTALS Kyber | NTRU Prime | X25519 (hybrid) | CRYSTALS Dilithium

---

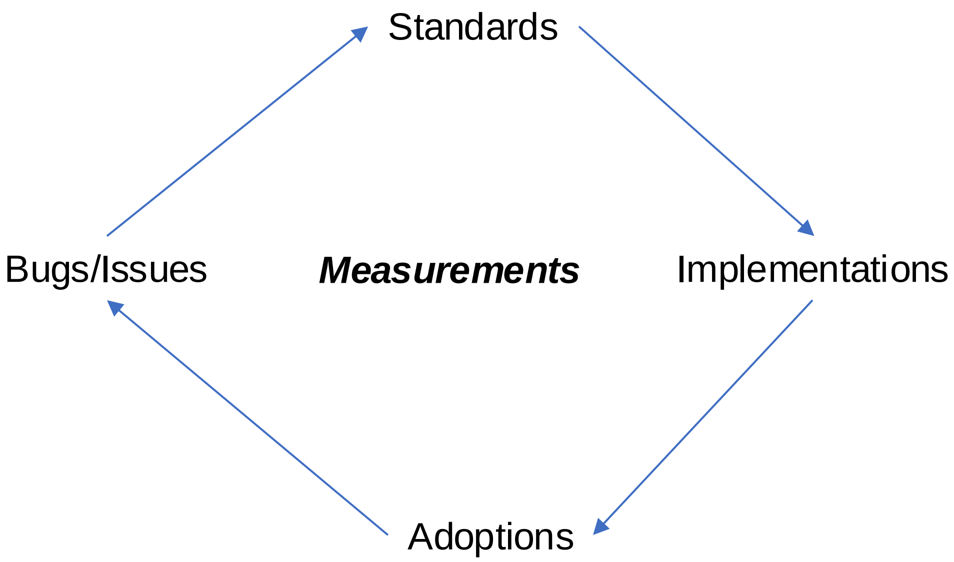
TLS 1.3 | OpenSSL | OpenSSH | … | SciTokens

Implementations

Summary and Key Takeaways

Problems

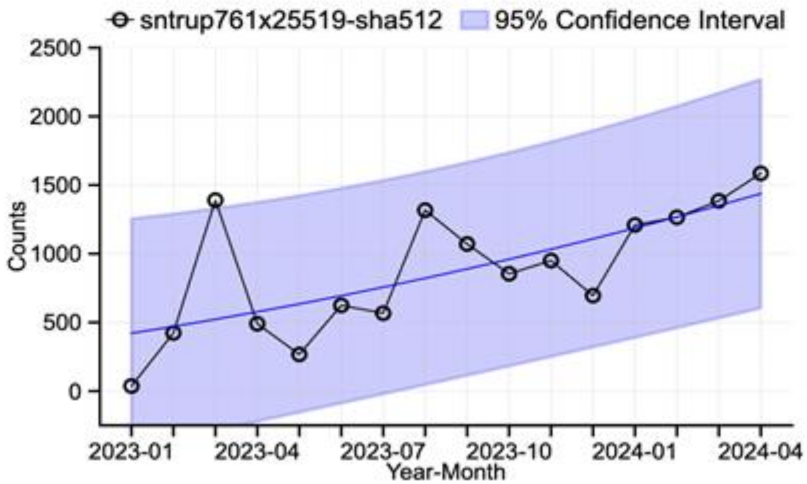
- Inadequate guidance on migrating HPC cyberinfrastructure to be compliant.
- **Insufficient feedback on PQC drafts and real-world adoption**



Summary and Key Takeaways

Problems

- Inadequate guidance on migrating HPC cyberinfrastructure to be compliant.
- Insufficient feedback on PQC drafts and real-world adoption
- **Lack of quantitative, compelling argument for increasing public awareness**



Example of SSH connections using NTRU Prime 761 and x25519 measured at NCSA (2024)

Summary and Key Takeaways

Problems

- Inadequate guidance on migrating HPC cyberinfrastructure to be compliant.
- Insufficient feedback on PQC drafts and real-world adoption
- Lack of quantitative, compelling argument for increasing public awareness

State of the Art

- Initial migration of TLS to PQC (Cloudflare, Google, Meta etc.)
- Alliance on standard PQC implementation

Need a concerted effort focusing on PQC adoption measurements on HPC environment.

Summary and Key Takeaways

Problems

- Inadequate guidance on migrating HPC cyberinfrastructure to be compliant.
- Insufficient feedback on PQC drafts and real-world adoption
- Lack of quantitative, compelling argument for increasing public awareness

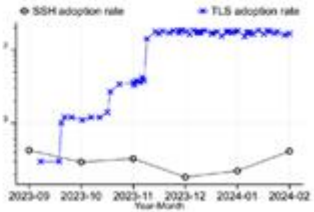
State of the Art

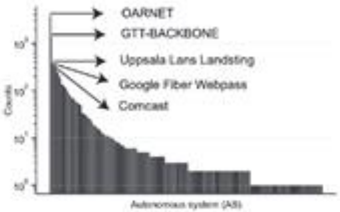
- Initial migration of TLS to PQC (Cloudflare, Google, Meta etc.)
- Alliance on standard PQC implementation

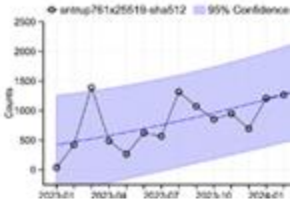
Need a concerted effort focusing on PQC adoption measurements on HPC environment.

Approach & Results

- ***Described a PQC instrument embedded in network of open-science HPC applications.***
- ***Analyzed Zeek connection metadata (SSH, TLS, RDP) collected at > 400Gbps NCSA network***
  - Avg. 0.029% adoption rate of sntrup761 for SSH (out of 20M connections from 2023-2024 at NCSA)
- **Systematically characterized current adoption of HPC authentication libraries, applications [1] (Published in IEEE QCE 2024)**







[1] Jakub Sowa, Jakub Sowa, Bach Hoang, Advaith Yeluru, Steven Qie, Santiago Nunez Corrales, Anita Nikolich, Ravishankar Iyer, **Phuong Cao** "*Post-Quantum Cryptography (PQC) Network Instrument: Measuring PQC Adoption Rates and Identifying Migration Pathways*" In 2024 IEEE International Conference on **Quantum Computing and Engineering** (**QCE**), Montreal, Canada

Summary and Key Takeaways

Problems

- Inadequate guidance on migrating HPC cyberinfrastructure to be compliant.
- Insufficient feedback on PQC drafts and real-world adoption
- Lack of quantitative, compelling argument for increasing public awareness

State of the Art

- Initial migration of TLS to PQC (Cloudflare, Google, Meta etc.)
- Alliance on standard PQC implementation

Need a concerted effort focusing on PQC adoption measurements on HPC environment.

Approach & Results

- ***Described a PQC instrument embedded in network of open-science HPC applications.***
- ***Analyzed Zeek connection metadata (SSH, TLS, RDP) collected at > 400Gbps NCSA network***
  - Avg. 0.029% adoption rate of sntrup761 for SSH (out of 20M connections from 2023-2024 at NCSA)
- **Systematically characterized current adoption of HPC authentication libraries, applications [1] (Published in IEEE QCE 2024)**

Future Work & Discussions

- Disseminating real-time and snapshot of adoption results with NIST & community
- Identify traces of novel attacks in the wild (e.g., ciphersuite downgrade attacks?)
- Work with HPC cyberinfrastructure such as SciTokens to identify and overcome challenges.

[1] Jakub Sowa, Jakub Sowa, Bach Hoang, Advaith Yeluru, Steven Qie, Santiago Nunez Corrales, Anita Nikolich, Ravishankar Iyer, **Phuong Cao**
"*Post-Quantum Cryptography (PQC) Network Instrument: Measuring PQC Adoption Rates and Identifying Migration Pathways*"
In 2024 IEEE International Conference on **Quantum Computing and Engineering** (**QCE**), Montreal, Canada

Migrating HPC application's communications to become Quantum-resistant

Research Questions:

1. To what extent are NCSA and its scientific partners ready for post-quantum cryptography (PQC)?

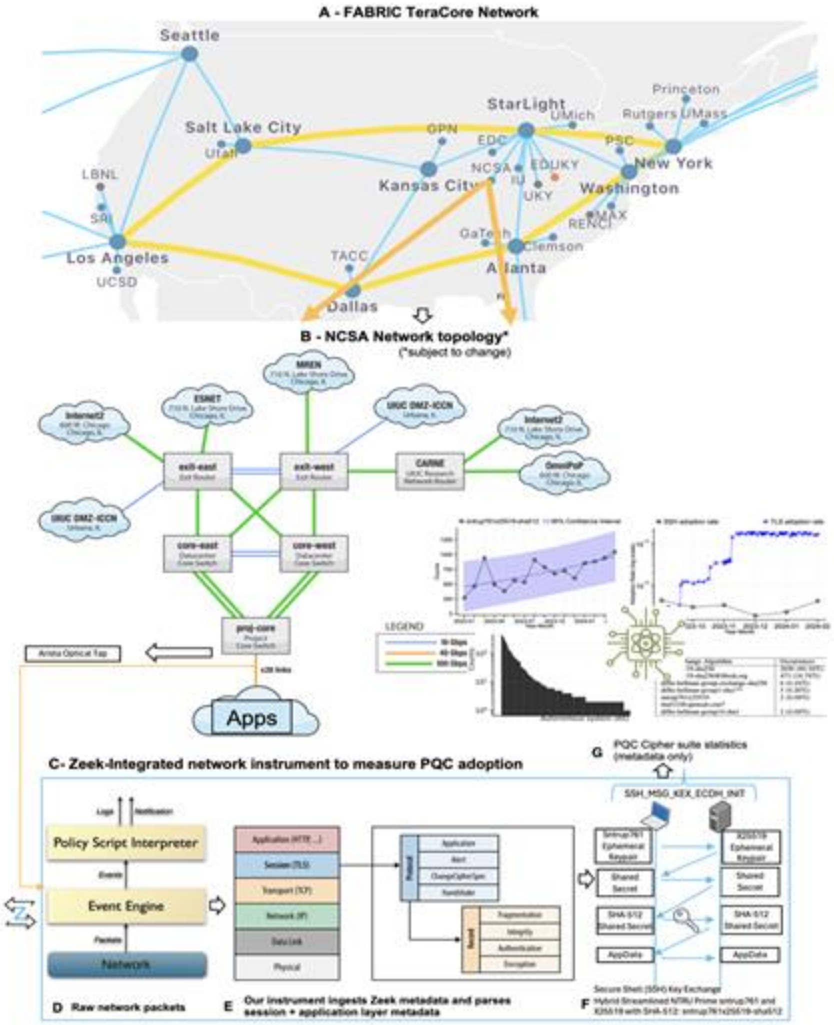
1. What can we do to better prepare for PQC standardization?

PQC measurement architecture

Harness the connectivity and
wide gamut of HPC applications
at NCSA to measure PQC adoption rate

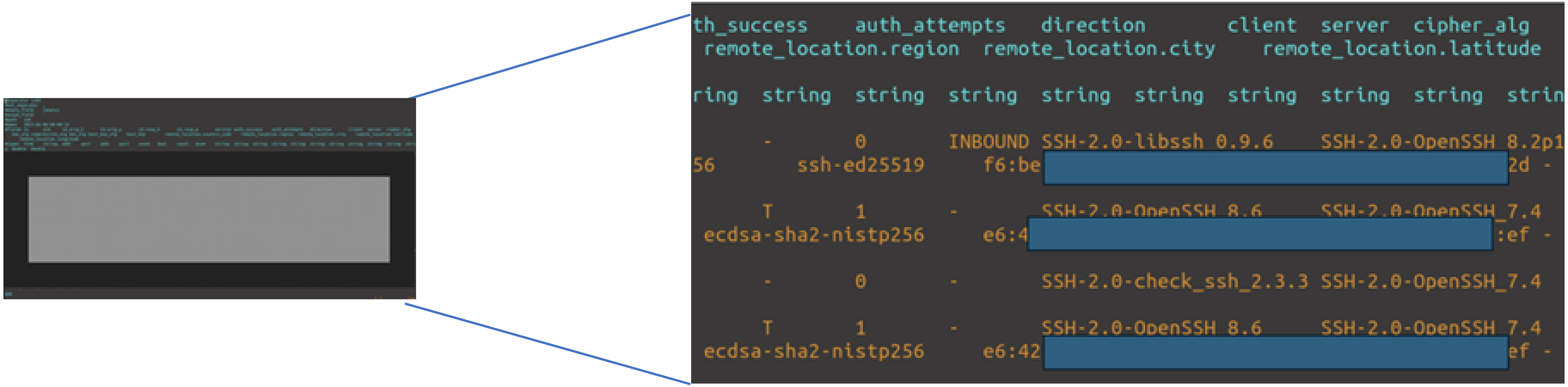
Focus on the most popular protocols:
SSH and TLS

Produce batch and real-time statistics



NCSA Network Metadata-Gathering Process

- Sampled many hours of <u>network metadata</u> generated by Zeek at NCSA
- No information beyond metadata was used
- Zeek logs were parsed in Python for analysis of network traffic of certain protocols



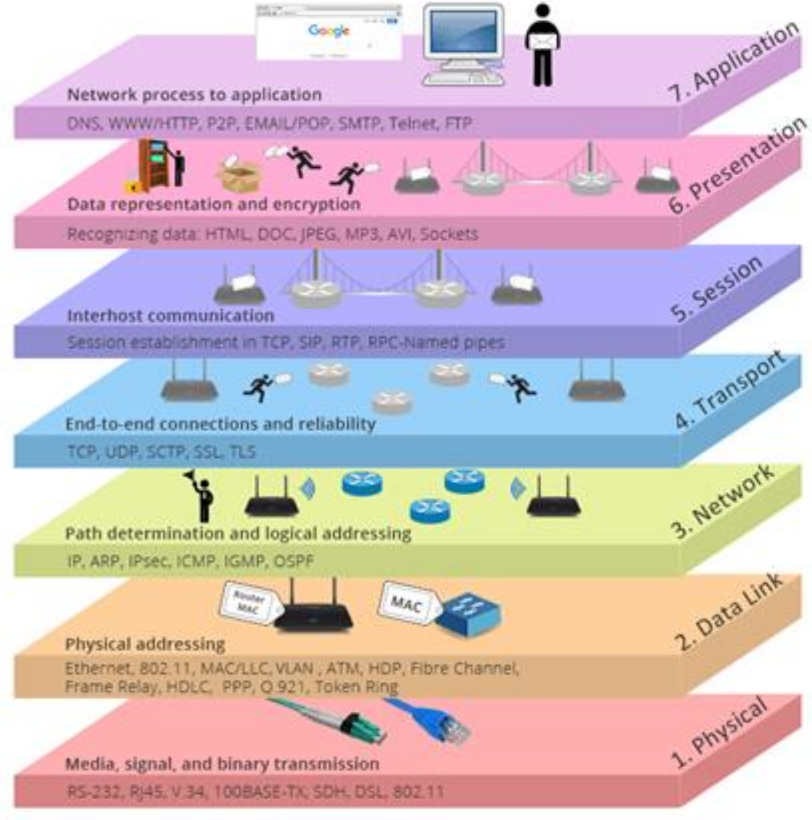
Data Overview

Data Characteristics	Data Collection	Description
Number of protocols	9 major network protocols	DNS, Kerberos, Modbus, MySQL, Radius, X509, SSL, SSH, application logs (syslog)
Data generator	Zeek	Zeek parses raw network packets and produces metadata of network connections.
Data generation rate	$\approx$ 30GB compressed logs per day	Data is compressed in to chunks every hour in the `gzip` format
Network speed	400Gbps	The network border links are 400Gbps and is connected to a TeraCore link
Data amount	13 TB	Total longitudinal data collected across all seven layers of network
Format	Tab-separated values (tsv)	Each network protocol has specific fields (source, destination, host key algorithm, etc.)
Privacy	Connection metadata	Only contain metadata of handshake, key exchange, and public certificates (no personally identifiable user data).
Workload characteristics	Batch, Real-time AI inference, large file transfer (petabytes)	These workloads make use of the above network protocols, providing a rich source for our analysis.
Source and destination	NCSA and its partner facilities (FABRIC, SDSC, Starlight, ESnet)	Diverse set of partners provide a good vantage point for our analysis.
Scientific applications	Representative applications	SciTokens [7], Kubernetes [8], Kerberos [9], Globus [10], and Slurm [11]
Time period	2023-01 to 2024-04 (present)	Data are collected in real-time and stored in a network-attached storage system
Sample PQC protocol	Secure Shell (SSH) connection	Samplg log: `73.45.xxx.yyy 22 SSH-2.0-OpenSSH_9.1p1 Debian-2 chacha20-poly1305@openssh.com umac-64-etm@openssh.com sntrup761x25519-sha512@openssh.com ecdsa-sha2-nistp256`

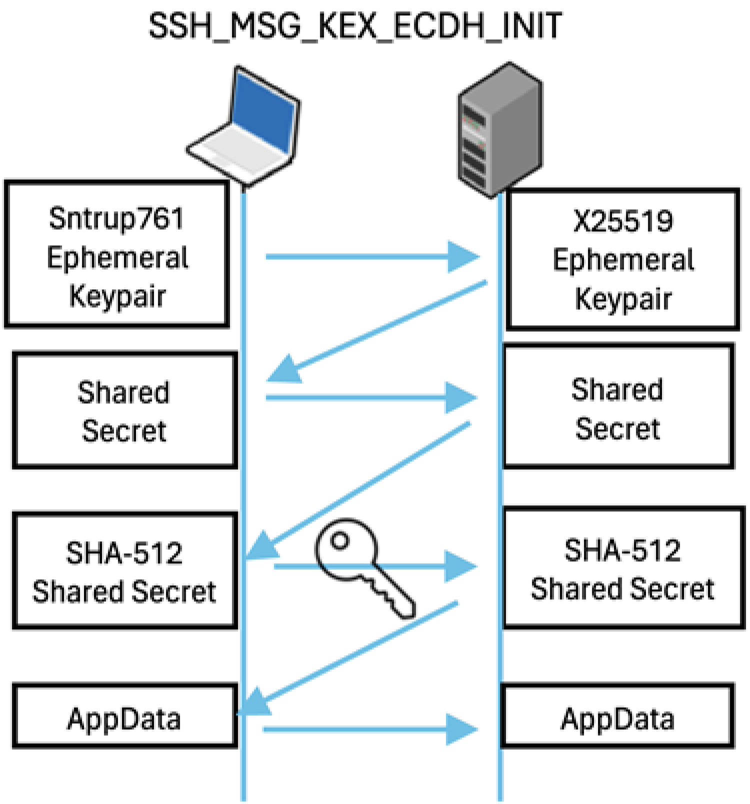
Investigated PQC of Application and Transport Layers

- Layer 7: Application layer
  - Remote Desktop Protocol (RDP)
  - Domain Name System (DNS)
  - Secure Shell (SSH)

- Layer 4: Transport layer
  - Transport Layer Security (TLS)



Current PQC implementation in SSH



Minimal Amount of PQC in the Secure Shell (SSH)

Encryption Algorithm	Occurrences
aes256-gcm@openssh.com	1686 (66.93%)
aes128-ctr	454 (18.02%)
chacha20-poly1305@openssh.com	188 (7.46%)
aes128-gcm@openssh.com	156 (6.19%)
aes256-ctr	31 (1.23%)
aes128-cbc	2 (0.08%)
3des-cbc ✖	1 (0.04%)

MAC Algorithm	Occurrences
hmac-sha2-256-etm@openssh.com	1844 (73.20%)
hmac-sha2-256	457 (18.14%)
umac-128-etm@openssh.com	154 (6.11%)
umac-64-etm@openssh.com	33 (1.31%)
hmac-sha1	17 (0.67%)
hmac-sha2-512	13 (0.52%)

Host Key Algorithm	Occurrences
ecdsa-sha2-nistp256	1275 (50.62%)
ssh-ed25519	1233 (48.95%)
ssh-rsa ✖	5 (0.20%)
rsa-sha2-512	4 (0.16%)

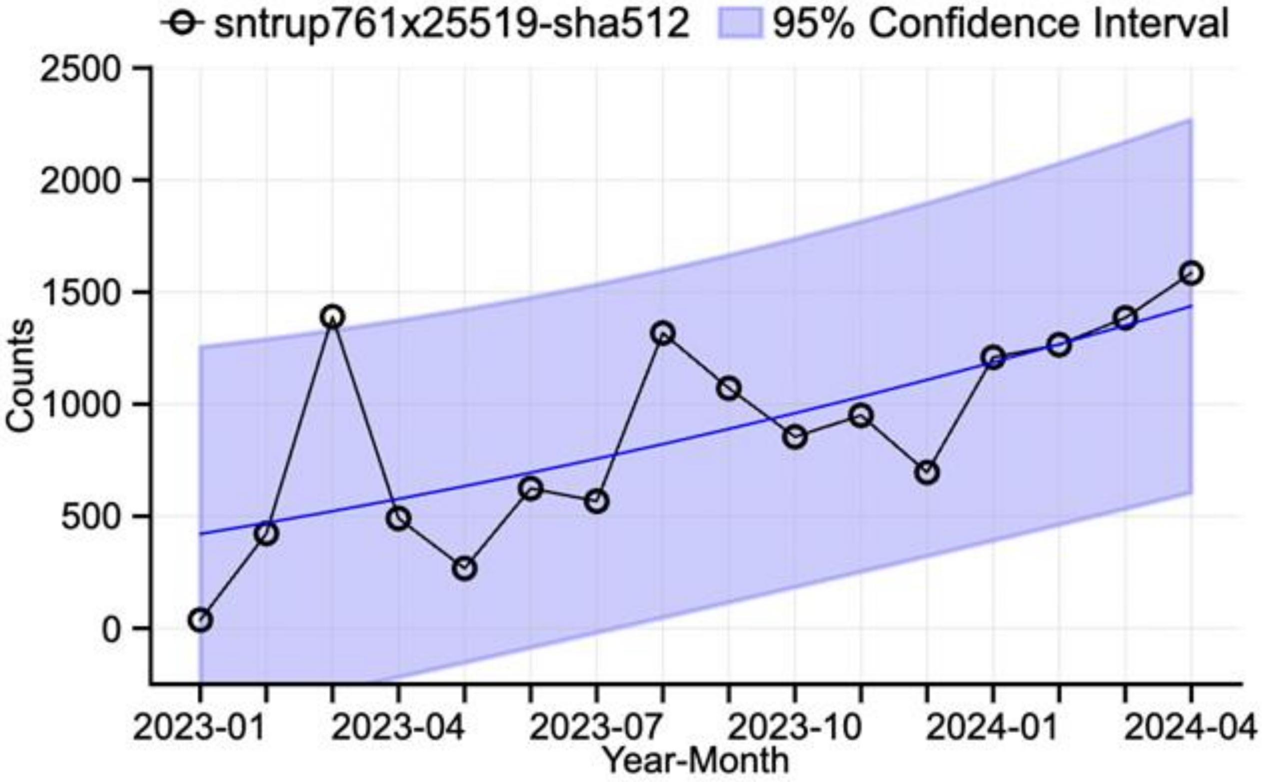
Key Exchange Algorithm	Occurrences
curve25519-sha256	2030 (80.59%)
curve25519-sha256@libssh.org	473 (18.78%)
diffie-hellman-group-exchange-sha256	6 (0.24%)
diffie-hellman-group1-sha1 ✖	5 (0.20%)
sntrup761x25519-sha512@openssh.com ★	2 (0.08%)
diffie-hellman-group14-sha1	2 (0.08%)

★ = post-quantum ✖ = not secure even now

- 99.92% of all SSH traffic was **not secure** against quantum adversaries
- <u>**sntrup761x25519**</u>: Streamlined NTRU Prime
  - A hybrid classical-PQ key exchange algorithm available by default in OpenSSH v9.0 and above as of **2022**
- Over 83% of server-side SSH protocol versions were from **2019 and earlier**

Key Exchange Algorithm	Occurrences
curve25519-sha256	2030 (80.59%)
curve25519-sha256@libssh.org	473 (18.78%)
diffie-hellman-group-exchange-sha256	6 (0.24%)
diffie-hellman-group1-sha1 ✖	5 (0.20%)
sntrup761x25519-sha512@openssh.com ★	2 (0.08%)
diffie-hellman-group14-sha1	2 (0.08%)

SSH PQC Key Exchange adoption rate is increasing over the year



Lack of PQC in Transport Layer Security (TLS)

- About 65% of connections were using TLSv1.3; about 35% were TLSv1.2
- Many unsecure cipher suites were in use – 4 had over 1000 connections!

- Many designs in the works by the IETF & NIST; some companies even trying to integrate PQC into their TLS

- The difficulty to even adopt TLS v1.3 internet-wide foreshadows PQC adoption as well

TLS Ciphersuites	Occurrences
TLS-AES-128-GCM-SHA256*	416447 (53.02%)
TLS-ECDHE-RSA-WITH-AES-256-GCM-SHA384	117788 (15.00%)
TLS-AES-256-GCM-SHA384*	100708 (12.82%)
TLS-ECDHE-RSA-WITH-AES-128-GCM-SHA256	79171 (10.08%)
TLS-DH-ANON-WITH-AES-256-GCM-SHA384**	42261 (5.38%)
TLS-ECDH-ANON-WITH-AES-256-CBC-SHA**	14787 (1.88%)
TLS-ECDHE-RSA-WITH-NULL-SHA**	5612 (0.71%)
TLS-ECDHE-ECDSA-WITH-AES-128-GCM-SHA256	3382 (0.43%)
TLS-ECDHE-RSA-WITH-CHACHA20-POLY1305-SHA256	2787 (0.35%)
TLS-ECDHE-ECDSA-WITH-AES-256-GCM-SHA384	2497 (0.32%)

Table 2: A list of the top 10 cipher suites found in sample TLS connection data (*in TLSv1.3, **considered non-secure).

Updating Zeek network security monitor to integrate parsing of recent TLS cipher suites.

Putting SSH and TLS adoption rate in perspective

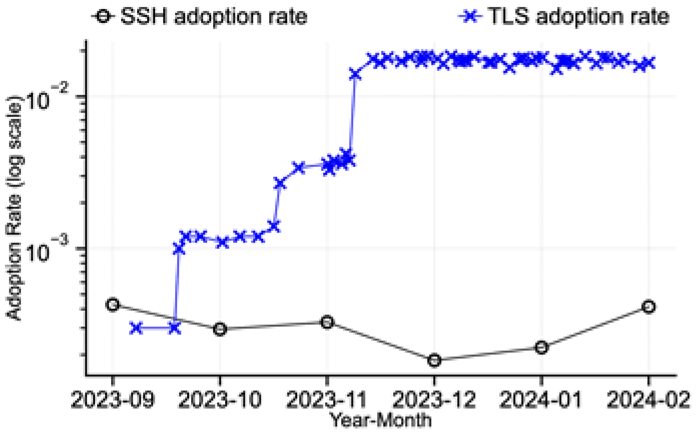


Fig. 2. Cross-protocol and cross-site comparison of adoption rate between SSH protocol at NCSA (our analysis) compared with publicly available TLS adoption rate at Cloudflare [6]. NCSA records an average of 0.029% (6044 out of 20,556,816 SSH connections) adoption rate for SSH, while Cloudflare recorded ≈ 1.78 percent adoption rate for TLS; more than 99% adoption came from Chrome [6].

Nearly Nonexistent Cryptography in RDP/DNS

- Remote Desktop Protocol (RDP):
  - Can be configured to use TLS encryption and Network-layer authentication but only on Windows 11
  - Out of 26 connections in sample data, only 2 used both encryption and authentication

- Domain Name System (DNS):
  - Not encrypted at all by default -- anyone can see what websites you try to visit, even on the NCSA network
    - Can enable HTTPS encryption on some browsers (Firefox, Chrome etc.)
    - Can also configure DNS to encrypt DNS-over-TLS (DoT)

Top Autonomous Systems with PQC traffic

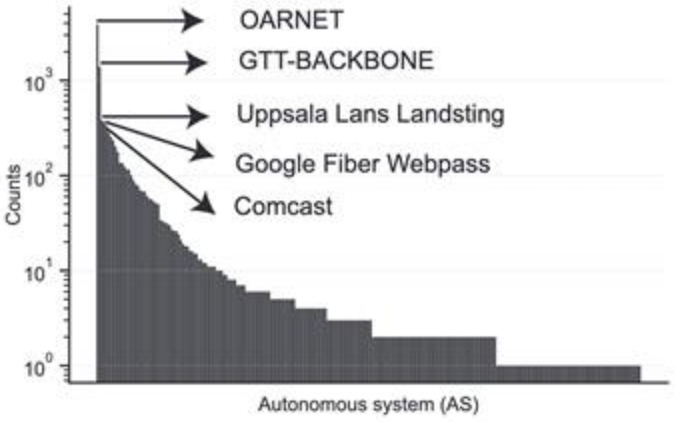


Fig. 3. A histogram of autonomous systems adopting PQC in SSH showing that top 5 ASes (OARNE, GTT, Google Fiber, Comcast, etc.) from U.S. and Uppsala Lans Landsting (Sweden) accounted for the majority of PQC in the head of the distribution. A long list of ASes is shown in the long tail.

Taxonomy of PQC adoptions in HPC applications

THE CURRENT STATE OF ADOPTION OF SCIENTIFIC APPLICATIONS AND PROTOCOLS REGARDING POST-QUANTUM CRYPTOGRAPHY. N/A ITEMS SHOW IN-PROGRESS OR INCOMPLETE INFORMATION TO DETERMINE PQC READINESS.

Protocols	Applications/ Libraries	Descriptions	Quantum Resistant Implemetation
BHR	ncsa/bhr [19]	Black Hole Router	N/A
DHCP	Internet Protocol	Dynamic Host Configuration Protocol	N/A
DNS	Internet Protocol Suite	Domain Name Service	N/A
DPD	Internet Key Exchange	Dead Peer Detection	N/A
HTTP	Internet Protocol Suite	Hypertext Transfer Protocol	Implement through SSL/TLS
FTP	SFTP	File Transfer Protocol	Implement through OpenSSH SCP
Kerberos	krb5 [20],	Network Authentication Protocol	N/A
	GSSAPI [21]		
Modbus	Modbus/TCP Security [22]	Client/Server Data Communication Protocol	N/A
MySQL	mysql-server [23]	Relational Database Protocol	N/A
NTLM		New Technology LAN Manager	N/A
RADIUS	FreeRADIUS [24]	Remote Authentication Dial-In User Service	N/A
RDP	FreeRDP [25]	Remote Desktop Protocol	N/A
SIP	RTP, SRTP	Session Initiation Protocol	N/A
SMB	Samba [26]	Server Message Block	N/A
SSH	openssh	Secure Shell	sntrup761x25519-sha512@openssh.com key exchange method
	libssh		
SSL/TLS	Open Quantum Safe [27]	Secure Sockets Layer	KEM (BIKE, CRYSTALS-Kyber), Signature (CRYSTALS-Dilithium)
SMTP		Simple Mail Transfer Protocol	N/A
SciTokens	scitokens	Federated Authorization for Distributed Scientific Computing	N/A

Future Work

- Creating a web tool that measures Post-Quantum Cryptography at NCSA and other organizations
  - "Network of PQC telescopes"

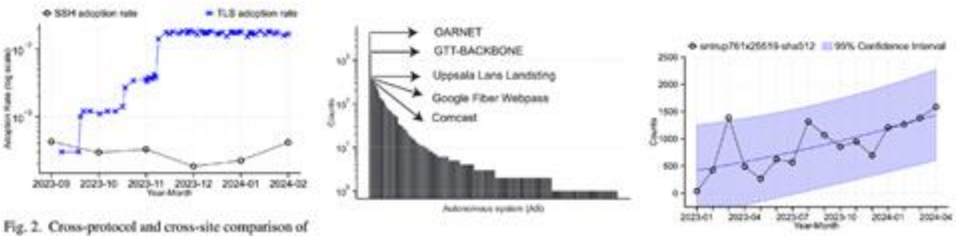
- Creating a tool to quickly scan a network and analyze its usage of PQC
  - As opposed to what we did - manually selecting logs and creating readable output

- Analyzing the risk of and figuring how to mitigate Post-Quantum "cipher suite downgrade attacks"
  - Even modern cryptography deals with downgrade attacks, where an adversary tries to force a connection to use less secure cryptography

Summary and Key Takeaways

Approach & Results

- ***Described a PQC instrument embedded in network of open-science HPC applications.***
- ***Analyzed Zeek connection metadata (SSH, TLS, RDP) collected at > 400Gbps NCSA network***
  - Avg. 0.029% adoption rate of sntrup761 for SSH (out of 20M connections from 2023-2024 at NCSA)
- **Systematically characterized current adoption of HPC authentication libraries, applications [1] (Published in IEEE QCE 2024)**



Adoption rate in SSH, TLS, and ASes

Future Work & Discussions

- Disseminating real-time and snapshot of adoption results with NIST & community
- Identify traces of novel attacks in the wild (e.g., ciphersuite downgrade attacks?)
- Work with HPC cyberinfrastructure such as SciTokens to identify and overcome challenges.

Acknowledgements

TrustedCI, NSF, NCSA, UIUC

[1] Jakub Sowa, Jakub Sowa, Bach Hoang, Advaith Yeluru, Steven Qie, Santiago Nunez Corrales, Anita Nikolich, Ravishankar Iyer, **Phuong Cao**
"*Post-Quantum Cryptography (PQC) Network Instrument: Measuring PQC Adoption Rates and Identifying Migration Pathways*"
In 2024 IEEE International Conference on **Quantum Computing and Engineering** (**QCE**), Montreal, Canada

Potential Solutions

- A **TLS v2.0**, introducing PQC by default, focusing on HPC communications
  - Securing all network traffic even against quantum adversaries

- Potentially configure most network protocols to run over this TLS 2.0
  - A **TLS Termination Proxy** can be used as a wrapper around current infrastructure to make it easier to secure traffic
  - Streamlining and simplifying cryptography and security

- More generally, make sure to keep software like SSH protocols and browsers updated to use the safest cryptography