Google | Privacy, Safety & Security

Key Rotation Done Right:

How to Improve Your Security Posture and Migrate to PQC in One Go

Sophie Schmieg (sschmieg@google.com)

OSCW 2024

March 24, 2024

Threat Model

The four main areas of cryptography

Asymmetric Encryption

Used mainly for encryption in transit, allows sending confidential messages to another party, by negotiating a shared key.

Digital Signatures

Used very widely, allows for proof of documents being genuine.

Symmetric Cryptography

Used very widely, especially for encryption at rest and for actually transferring data for encryption in transit, allows to encrypt data with a key.

Fancy Cryptography

Various other uses of cryptography, often to accomplish complicated privacy guarantees.

The four main areas of cryptography

Asymmetric Encryption

Used mainly for encryption in transit, allows sending confidential messages to another party by negotiating a shared key.

Vulnerable to Store Now Decrypt Later

Digital Signatures

Used very widely, allows for proof of documents being genuine.

Symmetric Cryptography

Used very widely, especially for encryption at rest and for actually transferring data for encryption in transit, allows to encrypt data with a key.

Fancy Cryptography

Various other uses of cryptography, often to accomplish complicated privacy guarantees.

?

Asymmetric Encryption

- Encryption in Transit
  - S/MIME
  - HPKE
  - Other

Digital Signatures

PKI

- Very Complex
- Might require Merkle-tree Certificates

Tokens

- Some complexities
- Stateful and symmetric alternatives
- UOV & friends

Software Signatures

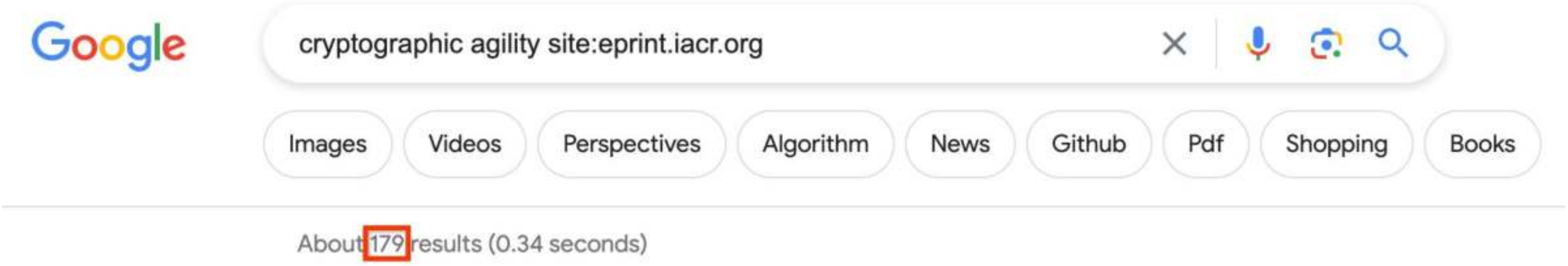
- Likely straightforward

Firmware Signatures

- Urgent
- Prefer Conservative Choices

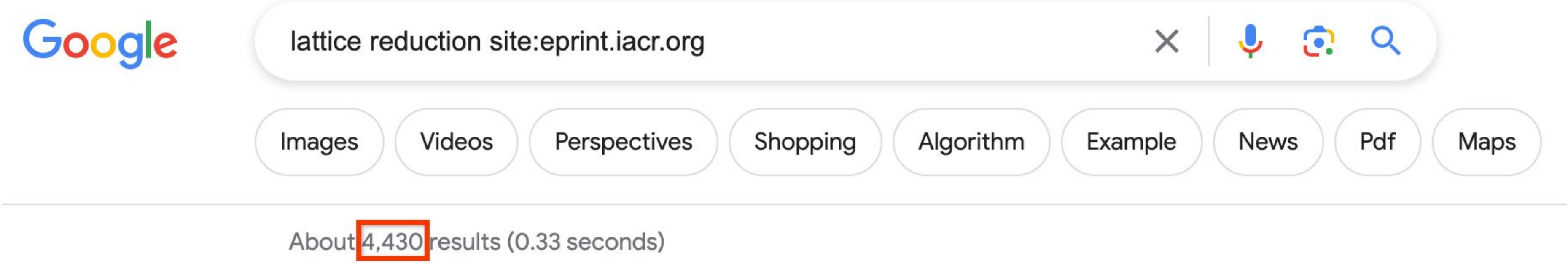
Cryptographic Agility

Definition



cryptographic agility site:eprint.iacr.org

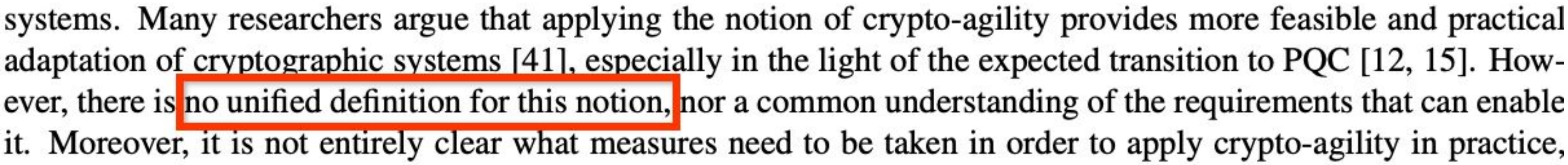
About 179 results (0.34 seconds)



lattice reduction site:eprint.iacr.org

About 4,430 results (0.33 seconds)

Definition



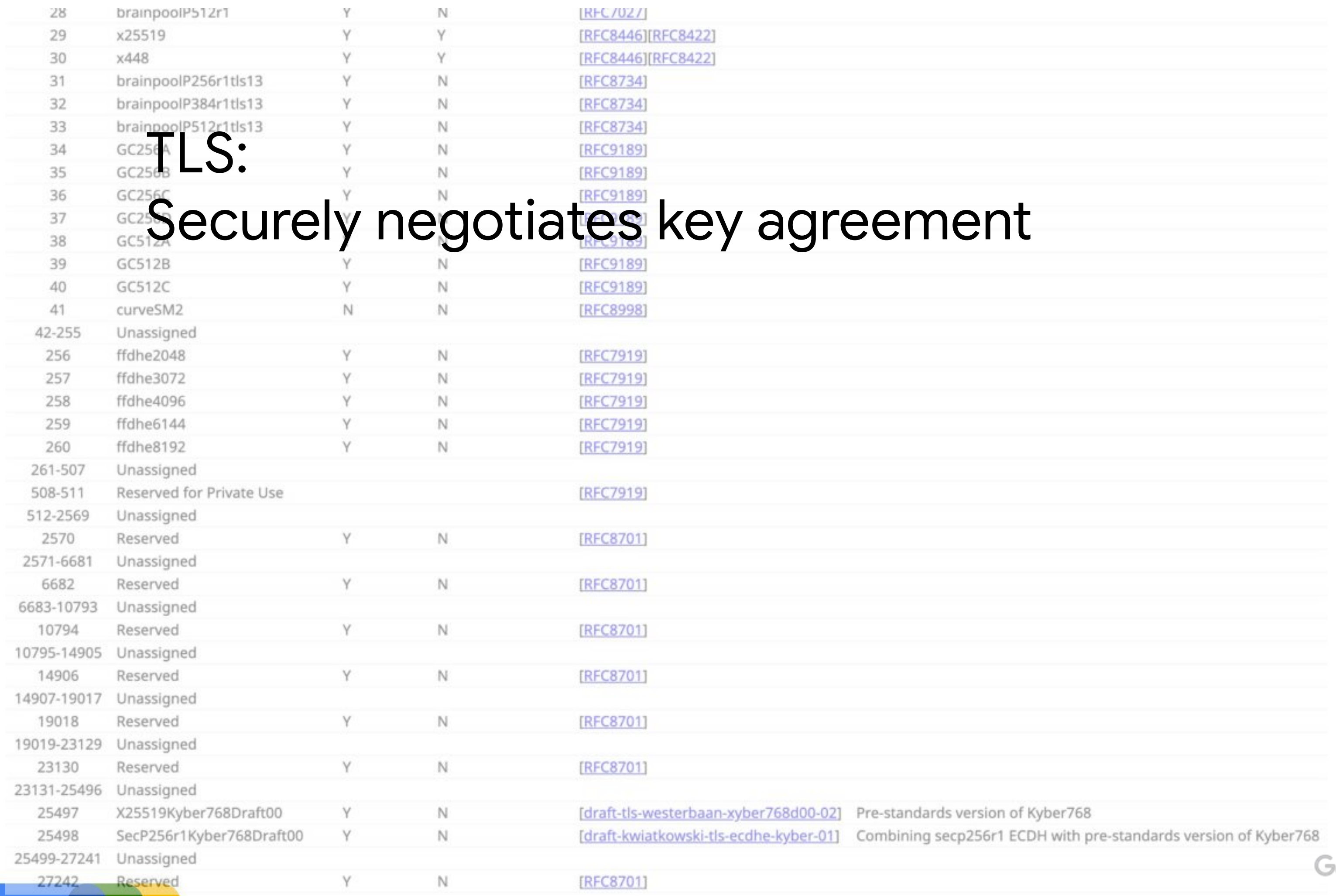
systems. Many researchers argue that applying the notion of crypto-agility provides more feasible and practical adaptation of cryptographic systems [41], especially in the light of the expected transition to PQC [12, 15]. However, there is no unified definition for this notion, nor a common understanding of the requirements that can enable it. Moreover, it is not entirely clear what measures need to be taken in order to apply crypto-agility in practice,

[1] "On the State of Crypto-Agility", https://eprint.iacr.org/2023/487

A first definition

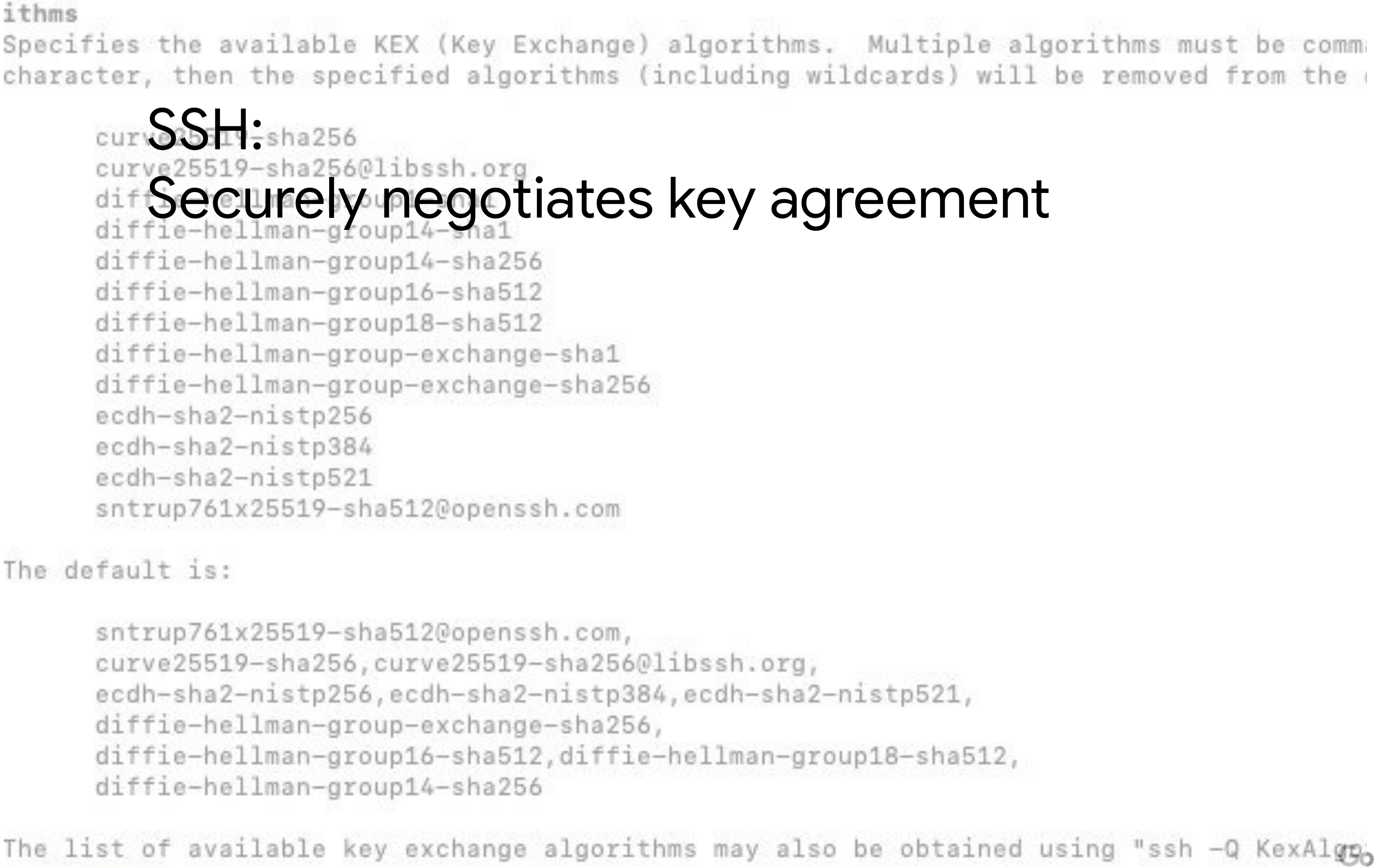
Crypto agility means the ability to change algorithms or parameter sets without major engineering effort.

TLS:
Securely negotiates key agreement



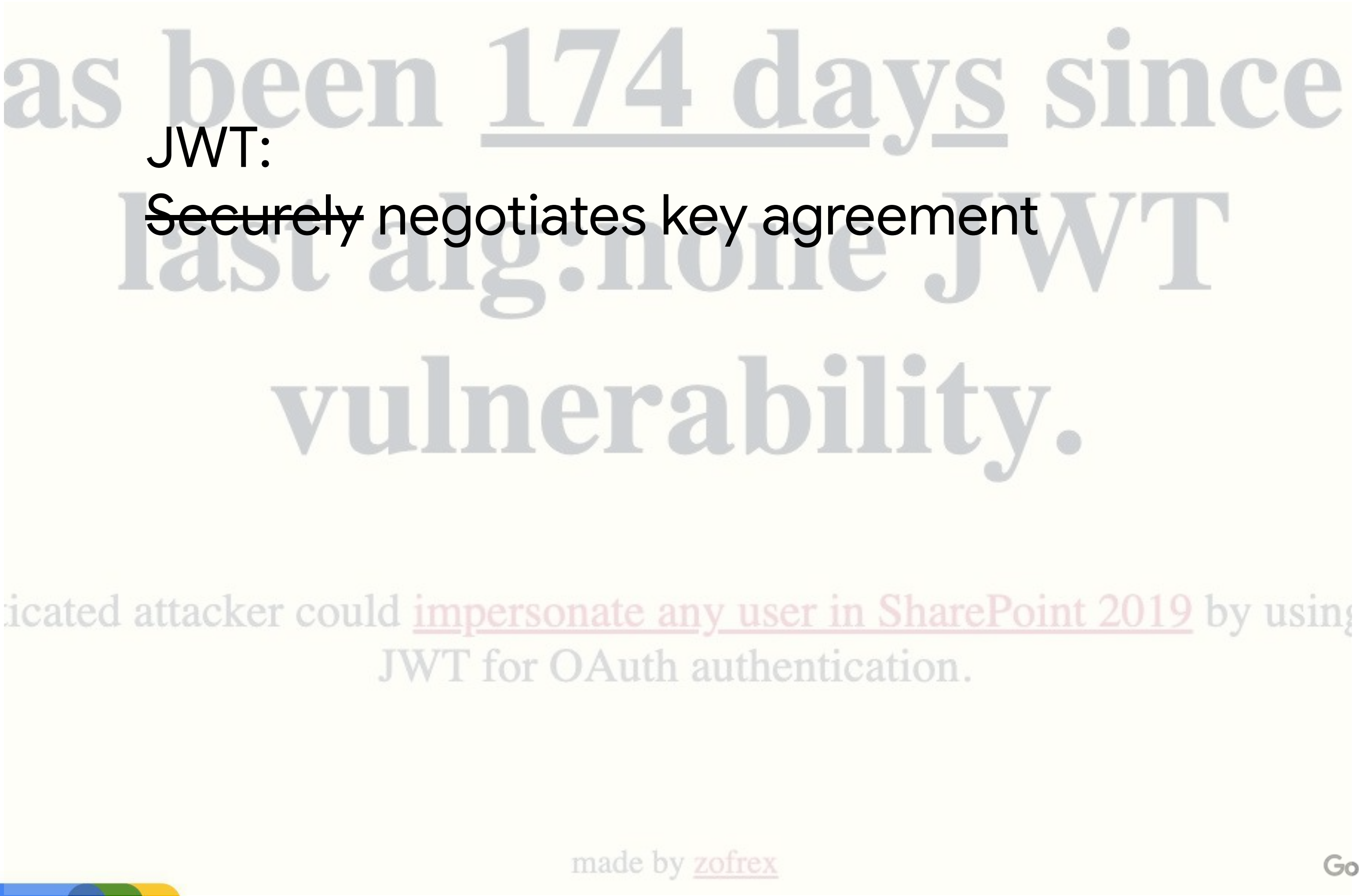
28	brainpoolP512r1	Y	N	[RFC7027]	
29	x25519	Y	Y	[RFC8446][RFC8422]	
30	x448	Y	Y	[RFC8446][RFC8422]	
31	brainpoolP256r1tls13	Y	N	[RFC8734]	
32	brainpoolP384r1tls13	Y	N	[RFC8734]	
33	brainpoolP512r1tls13	Y	N	[RFC8734]	
34	GC256A	Y	N	[RFC9189]	
35	GC256B	Y	N	[RFC9189]	
36	GC256C	Y	N	[RFC9189]	
37	GC256D	Y	N	[RFC9189]	
38	GC512A	Y	N	[RFC9189]	
39	GC512B	Y	N	[RFC9189]	
40	GC512C	Y	N	[RFC9189]	
41	curveSM2	N	N	[RFC8998]	
42-255	Unassigned				
256	ffdhe2048	Y	N	[RFC7919]	
257	ffdhe3072	Y	N	[RFC7919]	
258	ffdhe4096	Y	N	[RFC7919]	
259	ffdhe6144	Y	N	[RFC7919]	
260	ffdhe8192	Y	N	[RFC7919]	
261-507	Unassigned				
508-511	Reserved for Private Use			[RFC7919]	
512-2569	Unassigned				
2570	Reserved	Y	N	[RFC8701]	
2571-6681	Unassigned				
6682	Reserved	Y	N	[RFC8701]	
6683-10793	Unassigned				
10794	Reserved	Y	N	[RFC8701]	
10795-14905	Unassigned				
14906	Reserved	Y	N	[RFC8701]	
14907-19017	Unassigned				
19018	Reserved	Y	N	[RFC8701]	
19019-23129	Unassigned				
23130	Reserved	Y	N	[RFC8701]	
23131-25496	Unassigned				
25497	X25519Kyber768Draft00	Y	N	[draft-tls-westerbaan-xyber768d00-02]	Pre-standards version of Kyber768
25498	SecP256r1Kyber768Draft00	Y	N	[draft-kwiatkowski-tls-ecdhe-kyber-01]	Combining secp256r1 ECDH with pre-standards version of Kyber768
25499-27241	Unassigned				
27242	Reserved	Y	N	[RFC8701]	

SSH:
Securely negotiates key agreement



```
ithms
Specifies the available KEX (Key Exchange) algorithms.  Multiple algorithms must be comm
character, then the specified algorithms (including wildcards) will be removed from the

      curve25519-sha256
      curve25519-sha256@libssh.org
      diffie-hellman-group1-sha1
      diffie-hellman-group14-sha1
      diffie-hellman-group14-sha256
      diffie-hellman-group16-sha512
      diffie-hellman-group18-sha512
      diffie-hellman-group-exchange-sha1
      diffie-hellman-group-exchange-sha256
      ecdh-sha2-nistp256
      ecdh-sha2-nistp384
      ecdh-sha2-nistp521
      sntrup761x25519-sha512@openssh.com

The default is:

      sntrup761x25519-sha512@openssh.com,
      curve25519-sha256,curve25519-sha256@libssh.org,
      ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,
      diffie-hellman-group-exchange-sha256,
      diffie-hellman-group16-sha512,diffie-hellman-group18-sha512,
      diffie-hellman-group14-sha256

The list of available key exchange algorithms may also be obtained using "ssh -Q KexAlgo
```

JWT:
~~Securely~~ negotiates key agreement



as been 174 days since last alg:none JWT vulnerability.

icated attacker could impersonate any user in SharePoint 2019 by usin JWT for OAuth authentication.

made by zofrex

MISSION ACCOMPLISHED

A first definition

Crypto agility means the ability to change algorithms or parameter sets without major engineering effort.

A refined definition

Crypto agility means the ability to change algorithms or parameter sets of a deployed system without major engineering effort.

Corollary

Crypto Agility in practice is a Key Rotation problem!

Tink Keys

Tink Keys

ECDSA

P256/SHA256

```
x: 04f3…
y: 85cd…
s: 09fa…
```

Tink Keys

Keyset, Type: PublicKeySign

34ae Primary	ECDSA	P256/SHA256	x: 04f3… y: 85cd… s: 09fa…
a25f	ECDSA	P256/SHA256	x: e78a… y: 13fa… s: 98ee…
843b	ECDSA	P521/SHA512	x: 7c53… y: 9e9f… s: 8afc…
da3c	RSA-PKCS1	2048 bit, SHA256	n: 98f7… e: 10001 d: affe…

Tink Keys

Keyset, Type: PublicKeySign

34ae Primary	ECDSA	P256/SHA256	x: 04f3… y: 85cd… s: 09fa…
a25f	ECDSA	P256/SHA256	x: e78a… y: 13fa… s: 98ee…
843b	ECDSA	P521/SHA512	x: 7c53… y: 9e9f… s: 8afc…
da3c	RSA-PKCS1	2048 bit, SHA256	n: 98f7… e: 10001 d: affe…

Sample Signature:

```
01a25f9da0eb…
```

Tink Keys

Keyset, Type: PublicKeySign

34ae	Primary	ECDSA	P256/SHA256	x: 04f3… y: 85cd… s: 09fa…
a25f		ECDSA	P256/SHA256	x: e78a… y: 13fa… s: 98ee…
843b		ECDSA	P521/SHA512	x: 7c53… y: 9e9f… s: 8afc…
da3c		RSA-PKCS1	2048 bit, SHA256	n: 98f7… e: 10001 d: affe…

Sample Signature:

01a25f9da0eb…

Tink Keys

Keyset, Type: PublicKeySign

34ae	Primary	ECDSA	P256/SHA256	x: 04f3… y: 85cd… s: 09fa…
a25f		ECDSA	P256/SHA256	x: e78a… y: 13fa… s: 98ee…
843b		ECDSA	P521/SHA512	x: 7c53… y: 9e9f… s: 8afc…
da3c		RSA-PKCS1	2048 bit, SHA256	n: 98f7… e: 10001 d: affe…

Sample Signature:

`01a25f9da0eb…`

Tink Keys

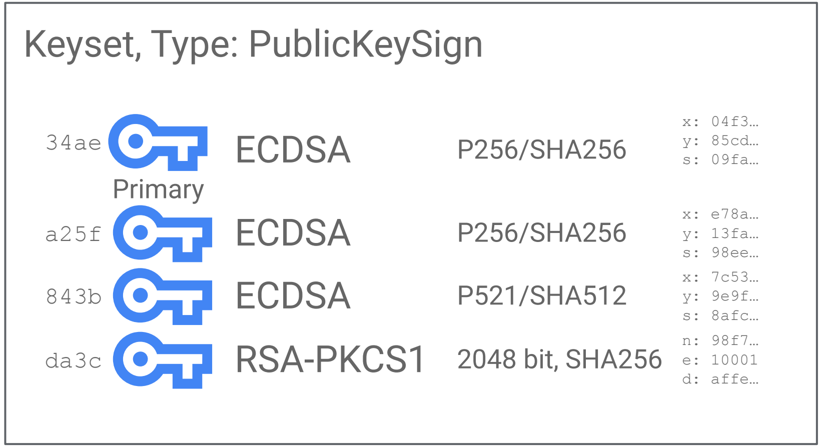
Keyset, Type: PublicKeySign

34ae Primary	ECDSA	P256/SHA256	x: 04f3… y: 85cd… s: 09fa…
a25f	ECDSA	P256/SHA256	x: e78a… y: 13fa… s: 98ee…
843b	ECDSA	P521/SHA512	x: 7c53… y: 9e9f… s: 8afc…
da3c	RSA-PKCS1	2048 bit, SHA256	n: 98f7… e: 10001 d: affe…

Sample Signature:

`01a25f9da0eb…`

Tink Keys



Keyset, Type: PublicKeySign

34ae	🔑 Primary	ECDSA	P256/SHA256	x: 04f3… y: 85cd… s: 09fa…
a25f	🔑	ECDSA	P256/SHA256	x: e78a… y: 13fa… s: 98ee…
843b	🔑	ECDSA	P521/SHA512	x: 7c53… y: 9e9f… s: 8afc…
da3c	🔑	RSA-PKCS1	2048 bit, SHA256	n: 98f7… e: 10001 d: affe…

Tink Keys

Keyset, Type: PublicKeySign

34ae Primary	ECDSA	P256/SHA256	x: 04f3… y: 85cd… s: 09fa…
a25f	ECDSA	P256/SHA256	x: e78a… y: 13fa… s: 98ee…
843b	ECDSA	P521/SHA512	x: 7c53… y: 9e9f… s: 8afc…

Tink Keys

Keyset, Type: PublicKeySign

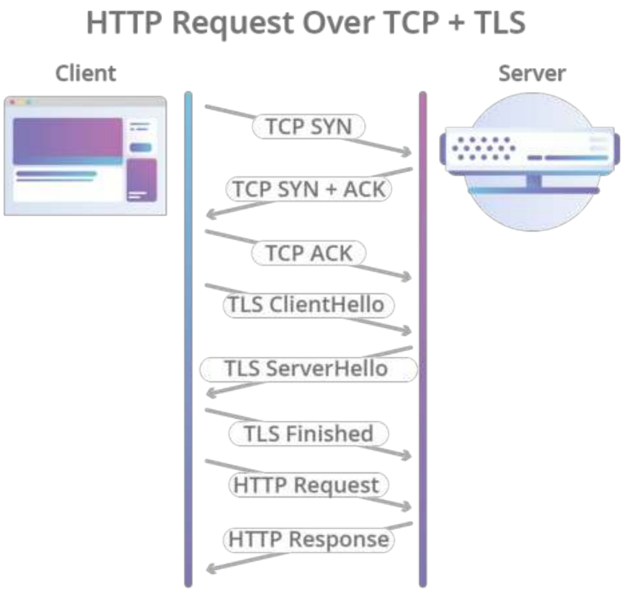
34ae	Primary 🔑	ECDSA	P256/SHA256	x: 04f3… y: 85cd… s: 09fa…
a25f	🔑	ECDSA	P256/SHA256	x: e78a… y: 13fa… s: 98ee…
843b	🔑	ECDSA	P521/SHA512	x: 7c53… y: 9e9f… s: 8afc…

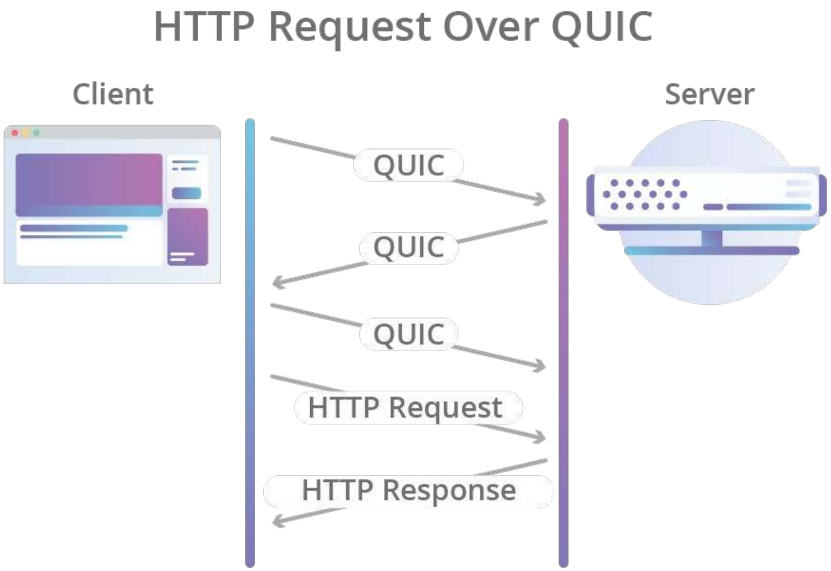
Tink Keys

Keyset, Type: PublicKeySign

fe71	🔑 Primary	ECDSA + Dilithium	P256/SHA256 Dilithum3	x: 04f3… y: 85cd… s: 09fa… ρ: 0a2b… s_1: 1e4f… …
34ae	🔑	ECDSA	P256/SHA256	x: 04f3… y: 85cd… s: 09fa…
a25f	🔑	ECDSA	P256/SHA256	x: e78a… y: 13fa… s: 98ee…
843b	🔑	ECDSA	P521/SHA512	x: 7c53… y: 9e9f… s: 8afc…

The Dark Side of Cryptographic Agility





Agility Takeaways

The Good:

- Protocols support agility
- Tink can make agility easier

The Bad:

- Agility is inherently and first and foremost a key rotation problem
- Rotating keys is hard
- Agility can be actively harmful to performance

Thank you