MPLS JAPAN 2026

WAN 品質別に見た 耐量子暗号の TLS 性能

ML-KEM・ML-DSA を 3 段階の WAN 品質で実測

2026 年 10 月 26 日 樽林 勇気
日本アイ・ビー・エム システムズ・エンジニアリング株式会社

アジェンダ

- 背景
- 検証
- 結果
- まとめ

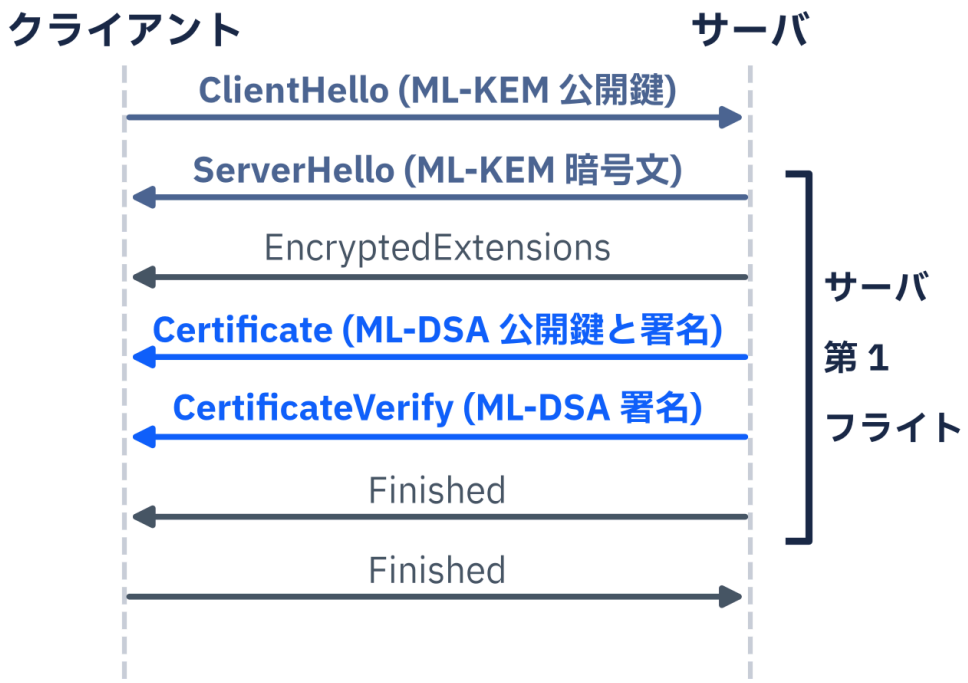
アジェンダ

- 背景
- 検証
- 結果
- まとめ

TLS1.3 における PQC アルゴリズム

アルゴリズムによって影響を受けるメッセージが異なる

- **鍵共有アルゴリズム → ML-KEM**
  通信の暗号鍵を共有する
  `ClientHello` と `ServerHello` が大きくなる
- **署名アルゴリズム → ML-DSA**
  サーバが本物であることを示す
  `Certificate` と `CertificateVerify` が大きくなる



古典アルゴリズムと PQC アルゴリズムの鍵・署名サイズ (規格値)

ML-KEM-768 の公開鍵は X25519 の約 37 倍、ML-DSA-87 の署名は P-256 の約 64 倍

アルゴリズム	種別	用途	公開鍵 [B]	暗号文・署名 [B]
X25519	古典	鍵共有	32	32
ML-KEM-768	PQC	鍵共有	1,184	1,088
ECDSA P-256	古典	署名	65	約 72
ML-DSA-44	PQC	署名	1,312	2,420
ML-DSA-87	PQC	署名	2,592	4,627

PQC アルゴリズムの値は使用した OpenSSL の出力と一致することを確認済

アジェンダ

- 背景
- 検証
- 結果
- まとめ

検証

検証環境と WAN 条件



`ClientHello` → / ← `ServerHello` (鍵共有)

← `Certificate` と `CertificateVerify` (署名)

WAN 品質	遅延	ジッタ	損失率	帯域	実測 RTT	想定
良好	20 ms	10 ms	0.1 %	100 Mbit/s	20.6 ms	社内・近距離
劣化	80 ms	30 ms	1.0 %	50 Mbit/s	80.0 ms	混雑した中距離
不良	200 ms	80 ms	3.0 %	10 Mbit/s	199.2 ms	遠距離・モバイル

検証

評価対象と測定方法

鍵共有アルゴリズムと署名アルゴリズムを独立に入れ替え、基準アルゴリズムに対して評価する

- 鍵共有 5 方式: **X25519 (基準)**、ML-KEM-512 / 768 / 1024、X25519MLKEM768
- 署名 6 方式: **ECDSA P-256 (基準)**、P-384、P-521、ML-DSA-44 / 65 / 87
  - 証明書は Root CA → 中間 CA → EE(End-Entity) の 3 階層で、全階層とも同じ署名方式で作成
  - サーバは EE 証明書と中間 CA 証明書の 2 枚を送る
  - EE 証明書に SCT (Certificate Transparency のログの署名) は含まない
- 接続時間 = `openssl s_client` 1 回の実行時間 (TCP 接続・TLS ハンドシェイク・HTTP 1 往復)
- 1 組合せ 1,000 回を同時接続 1 で測り、中央値の比と 95 % 信頼区間で比べる

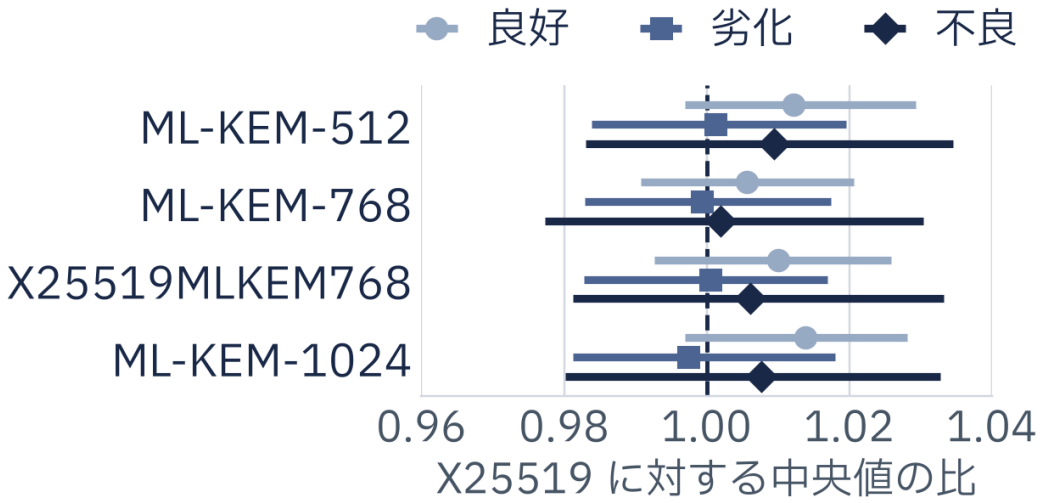
アジェンダ

- 背景
- 検証
- 結果
- まとめ

鍵共有アルゴリズム：接続時間の中央値の比

古典アルゴリズムから PQC アルゴリズムへ変更しても接続時間は変わらない

- X25519 に対する比は、PQC 4 方式・3 WAN で 0.997〜1.014 倍
- 95 % 信頼区間はすべて 1.0 を含み、WAN 品質や p95 で見ても変わらない
- ML-KEM-1024 は `ClientHello` が 1,754 B (TCP ペイロード) で 2 セグメントに分かれる
- それでも initcwnd 10 (14,480 B) に収まり、往復は増えない



X25519MLKEM768	比	95 % 信頼区間
良好	1.010	0.993〜1.026
劣化	1.000	0.983〜1.017
不良	1.006	0.981〜1.033

結果

鍵共有アルゴリズム：HelloRetryRequest

`HelloRetryRequest` が発生すると、1 RTT 増加する

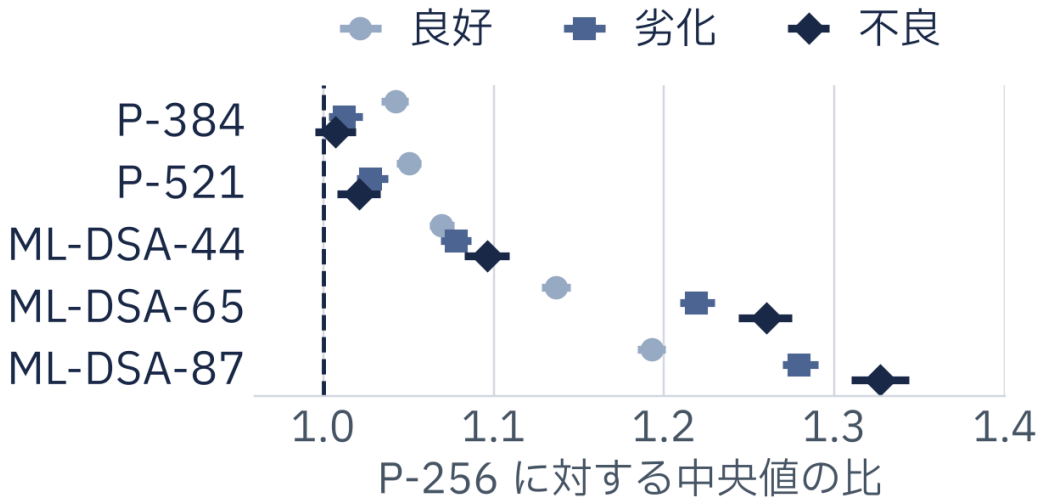
WAN 品質 (RTT)	PQC のみ	古典+PQC	古典のみ (HRR)	HRR の増分	増分 / RTT
良好 (20.6 ms)	108.3 ms	107.3 ms	129.8 ms	+21.6 ms	1.05
劣化 (80.0 ms)	308.8 ms	307.3 ms	386.1 ms	+77.3 ms	0.97
不良 (199.2 ms)	741.9 ms	731.6 ms	984.0 ms	+242.1 ms	1.22

- 初回の `ClientHello` の `key_share` に PQC の公開鍵を含めれば回避できる
- クラウド WAF とオリジンサーバ間などで PQC アルゴリズムを利用する場合などに考慮が必要

署名アルゴリズム: 接続時間の中央値の比

古典アルゴリズムから PQC アルゴリズムへ変更すると、遅くなる

- P-256 に対する比は、PQC 3 方式・3 WAN 品質で 1.07〜1.33 倍
- 95 % 信頼区間はすべて 1.0 を含まない
- 最も軽い ML-DSA-44 でも 1.07〜1.10 倍
- ML-DSA は 3 方式とも、WAN 品質が悪いほど比が大きい
- 比の順は `Certificate` (EE + 中間 CA 証明書) のサイズ順 (8.5 / 11.6 / 15.5 KB) と一致する



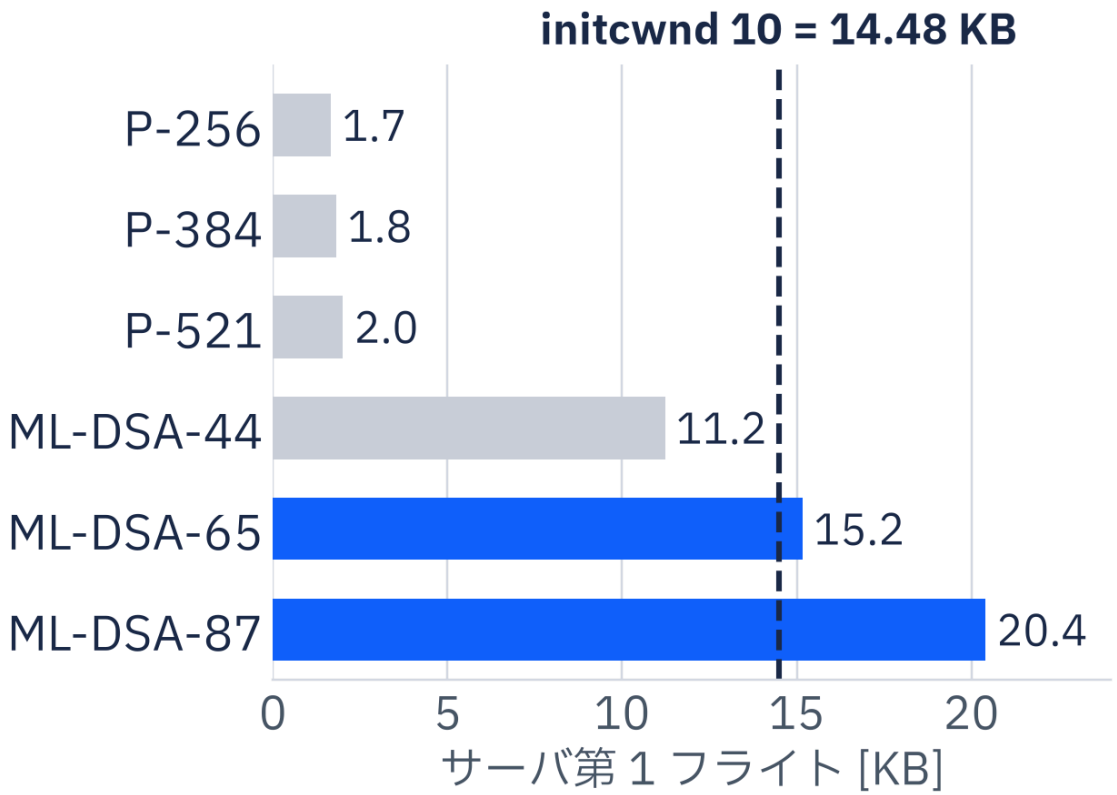
ML-DSA-87	比	95 % 信頼区間
良好	1.193	1.184〜1.201
劣化	1.279	1.269〜1.290
不良	1.327	1.310〜1.344

結果

署名アルゴリズム: initcwnd

第 1 フライトが initcwnd 10 (14,480 B) を超えると 1 RTT 増加する

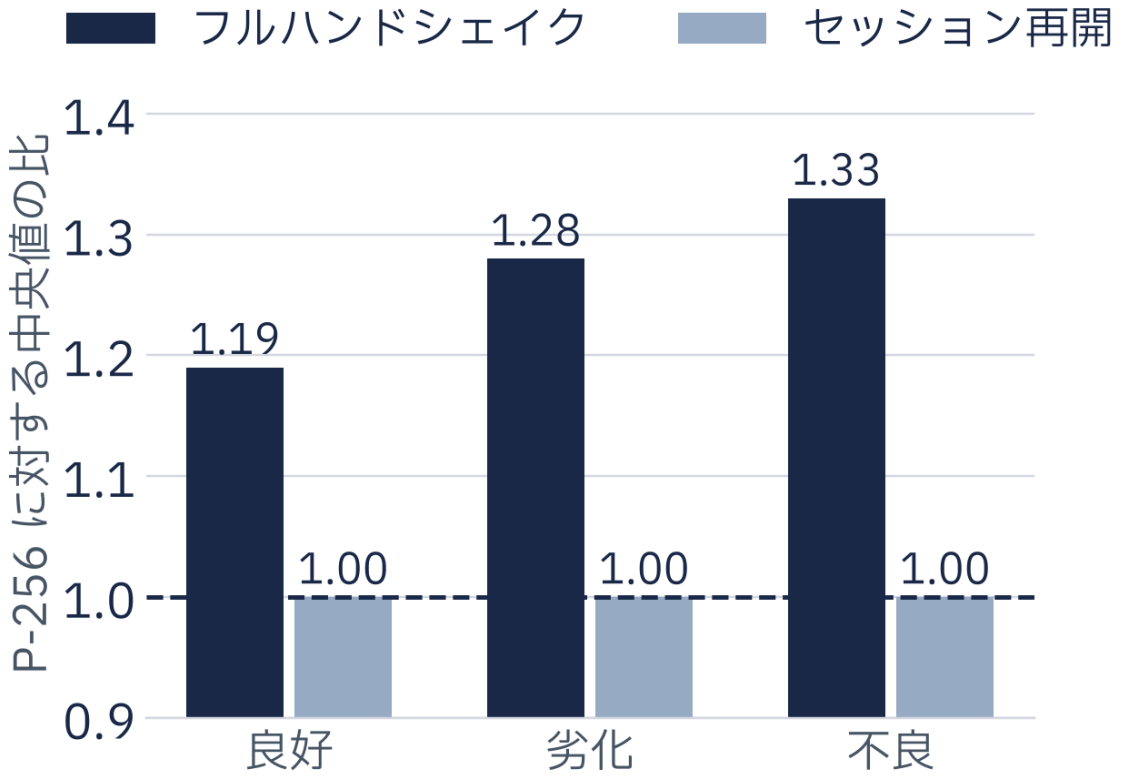
- `Certificate` は P-256 の 1.3 KB に対し、ML-DSA-87 では 15.5 KB (第 1 フライト全体では 20.4 KB)
- EE 証明書と中間 CA 証明書の両方に、PQC の公開鍵と署名が入る
- **initcwnd 10 (10 × MSS 1,448 B) を ML-DSA-65 / 87 が超える**
- 超えた分は、ACK を待ち、1 RTT 増加する
- ML-DSA-44 (11.2 KB) も、SCT 2 つ分 (約 5 KB) を足すと 14,480 B を超える (約 16.2 KB)



署名アルゴリズム: セッション再開

セッション再開時は `Certificate` と `CertificateVerify` が省略されるため、差が出ない

- ML-DSA-87 の P-256 比は
  フル 1.19〜1.33 倍 → 再開 1.00 倍
- サーバの送信量は 20,406 B から 241 B に減る
- **初回接続とチケット失効後はフルハンドシェイクになる**



結果

署名アルゴリズム: Merkle Tree Certificates 利用時の接続時間

MTC のサーバ第 1 フライトは 6.2 KB で、X.509 (12.3 KB) の約半分

WAN 品質 (RTT)	X.509 + SCT x2 > initcwnd 10	X.509 + SCT x2 < initcwnd 20	X.509 < initcwnd 10	MTC < initcwnd 10
良好 (20.6 ms)	**110.2 ms**	100.3 ms	98.4 ms	97.0 ms
劣化 (80.0 ms)	**361.5 ms**	309.2 ms	306.4 ms	300.5 ms
不良 (199.2 ms)	**888.1 ms**	759.7 ms	748.0 ms	729.1 ms

- X.509 が initcwnd を超える場合には、MTC で効果有り
- initcwnd に収まった場合、サイズが半分以下になるので、WAN 品質の影響が小さい

署名は全構成で ML-DSA-44、鍵共有は ML-KEM-768 で実施
X.509 は EE + 中間 CA

アジェンダ

- 背景
- 検証
- 結果
- まとめ

まとめ

移行の判断

WAN 品質が悪い拠点ほど、往復を減らす対策の効果が大きい

鍵共有アルゴリズム

- X25519MLKEM768 の実装も進んでいる
- WAN 品質の影響は小さく、PQC 移行できる
- `HelloRetryRequest` が発生すると
  1 RTT 増える
- 最初の `ClientHello` に PQC の `key_share`
  を載せるよう調整する

署名アルゴリズム

- WAN 品質の影響を受ける
- ML-DSA は第 1 フライトが
  initcwnd 10 を超える可能性が高いので注意する
- initcwnd を引き上げる
  (initcwnd 20 なら 1 RTT の増加は消える)
- セッション再開を活用する
- MTC の標準化と実装を注視する

Thank you!