Controller of Certifying Authorities
Ministry of Electronics & Information Technology
Government of India

IEEE COMPUTER SOCIETY Bangalore Chapter

IAS IEEE INDUSTRY APPLICATIONS SOCIETY Linking Research to Practice Bangalore Chapter

6th INTERNATIONAL CONFERENCE ON

PUBLIC KEY INFRASTRUCTURE AND ITS APPLICATIONS

(PKIA 2025)

SEPTEMBER 3-4th, 2025

Validation Framework for Module Lattice-based Post-quantum Digital Signature

Smitha G Havanur, Rahul Kumar,

Dr Abey Jacob

CDAC Bangalore

AGENDA

- Introduction
- Overview of ML-DSA
  - Parameter set for ML-DSA Algorithm
  - Auxiliary Functions ML-DSA Algorithm
  - ML-DSA Key Generation,
  - Signature generation
  - Signature verification Algorithm
- Test procedure and Validation results
- References

Overview of ML-DSA

- ML-DSA is an optimized Schnorr-like lattice-based signature scheme, Standardized by NIST in FIPS-204, called ML-DSA Module-Lattice-based Digital Signature Algorithm.
- Relies on hard lattice problems: Module Learning With Errors (M-LWE) problem and Module Short Integer Solution (MSIS) .
- ML-DSA standard defines three parameter sets with increasing security levels: ML-DSA-44, ML-DSA-65, and ML-DSA-87
- This research contributes to detailed validation procedures and systematic test vector generation with intermediate parameter verification to identify precise sources of implementation errors.

Parameter set for ML-DSA Algorithm

Parameter	ML-DSA-44	ML-DSA-65	ML-DSA-87
Modulus q	8380417	8380417	8380417
512th root of unity in Zq	1753	1753	1753
Matrix dimensions (k,ℓ)	(4, 4)	(6, 5)	(8, 7)
η (eta)	2	4	2
Number of ±1's in polynomial c = τ (tau)	39	49	60
Hint sparsity ω	80	55	75
z_threshold (γ_1-β)	130994	524092	524168
r0_threshold (γ_2-β)	95154	261692	261768
Claimed security strength	Category 2	Category 3	Category 5
Private Key (bytes)	2560	4032	4896
Public Key (bytes)	1312	1952	2592
Seed (bytes)	32	32	32
Signature Size (bytes)	2420	3309	4627

Auxiliary Functions ML-DSA Algorithm

SHAKE-128/256 and SHA3-256/512 used for expanding seeds, creating challenges and fixed-length cryptographic digests

NTT Point wise multiplication operations

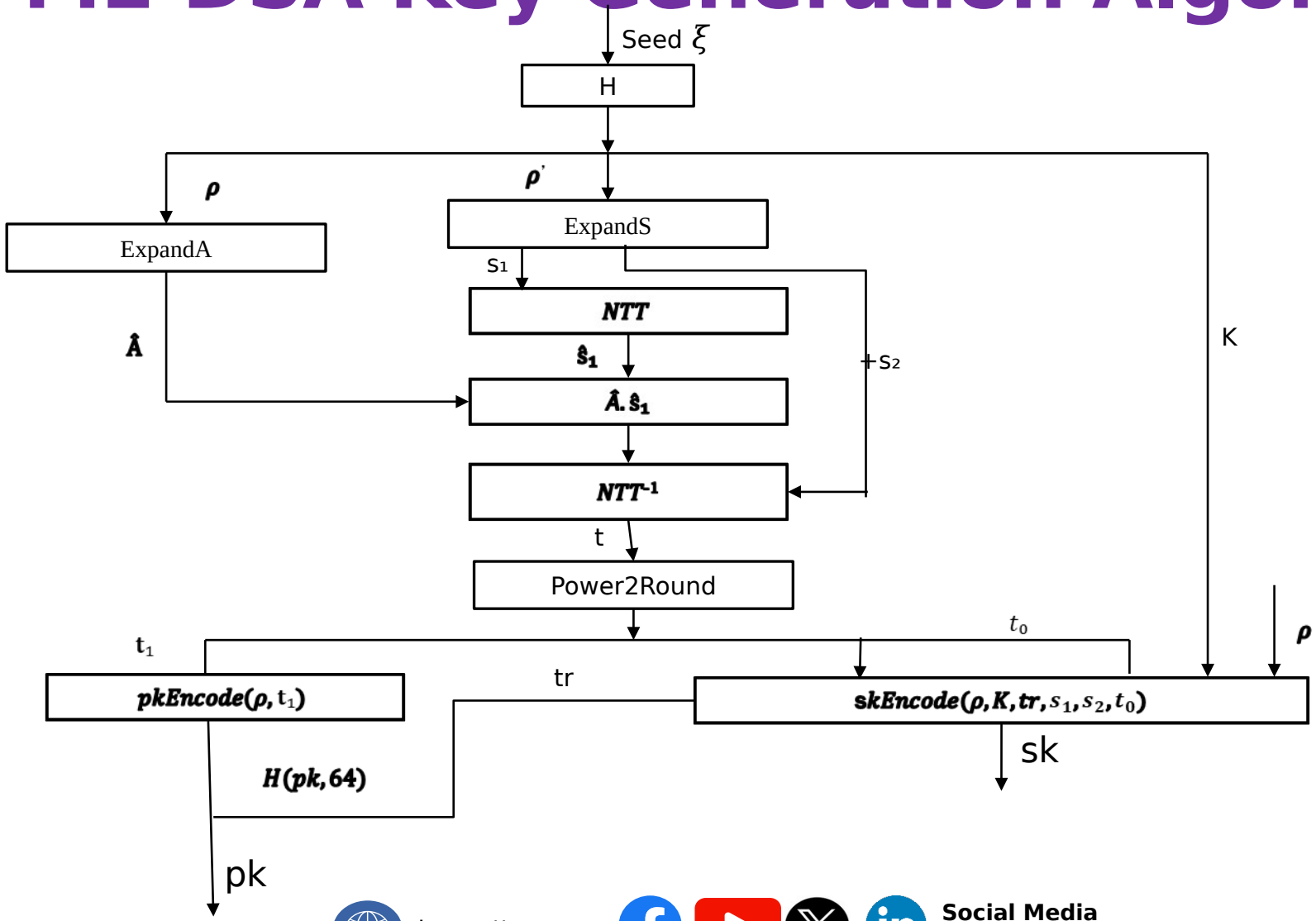
Encoding and Decoding translate ML-DSA public/private keys into byte strings and vice versa

ExpandA: Generates $k \times \ell$ public matrix

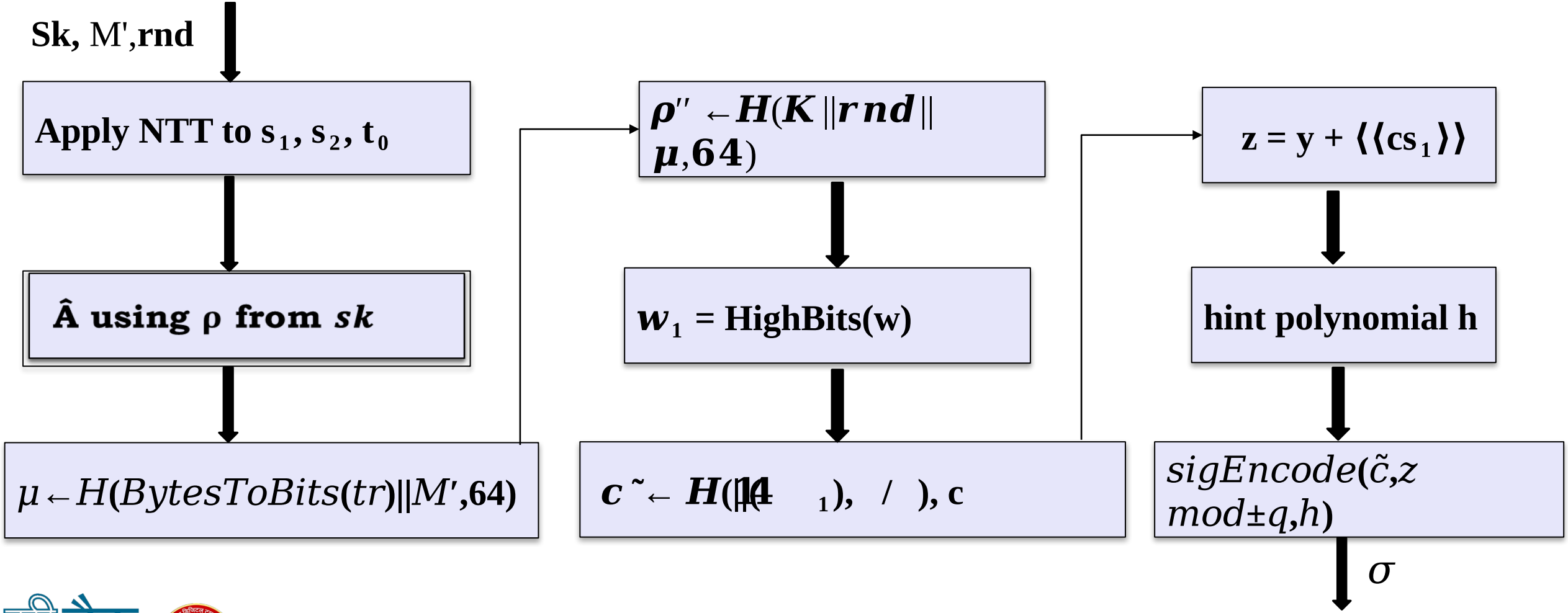
ExpandMask: Produces masking vectors using seed

Hints are created during signing and used during verification

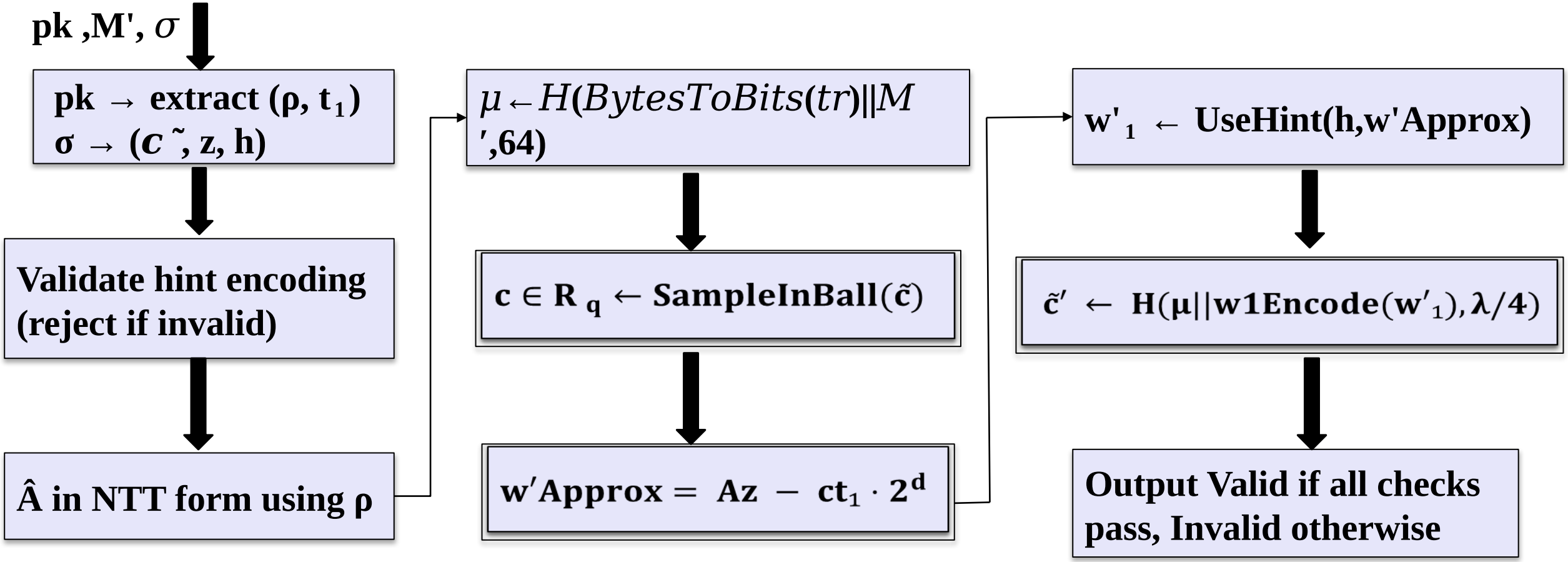
ML-DSA Key Generation Algorithm



ML-DSA Signing

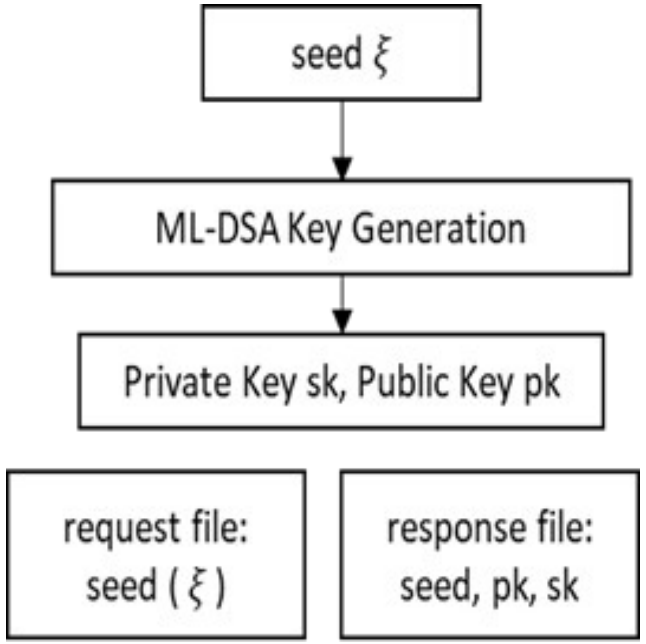


ML-DSA Signature Verification Algorithm

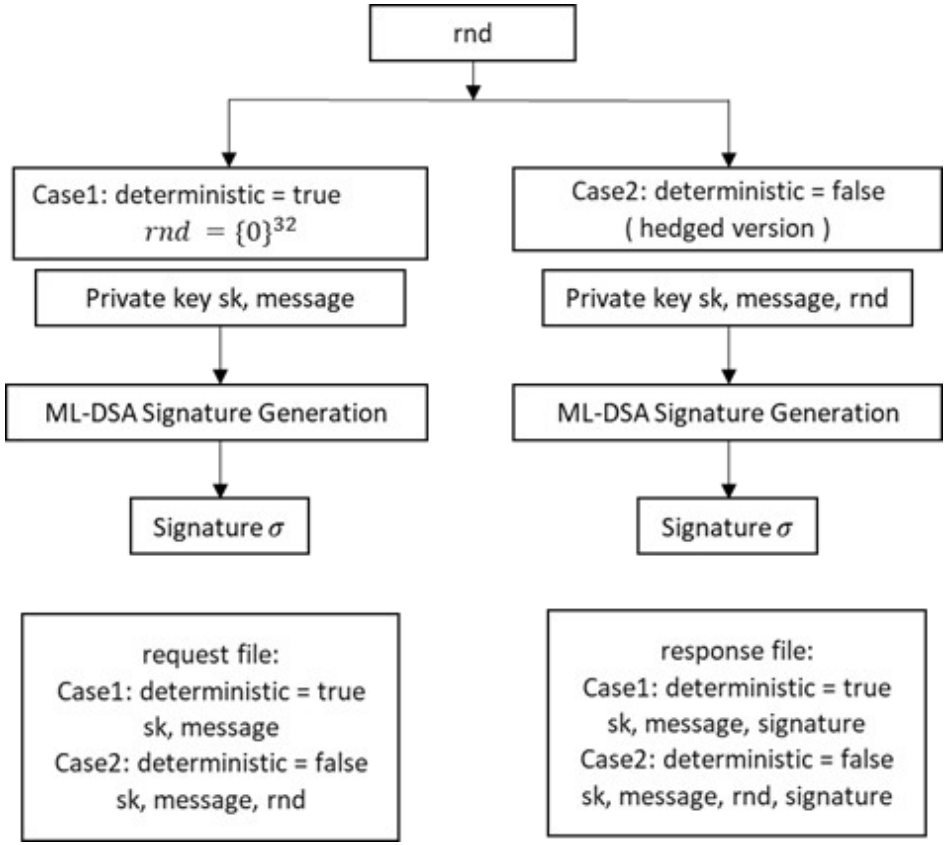


VALIDATION TEST IMPLEMENTATION FOR ML-DSA

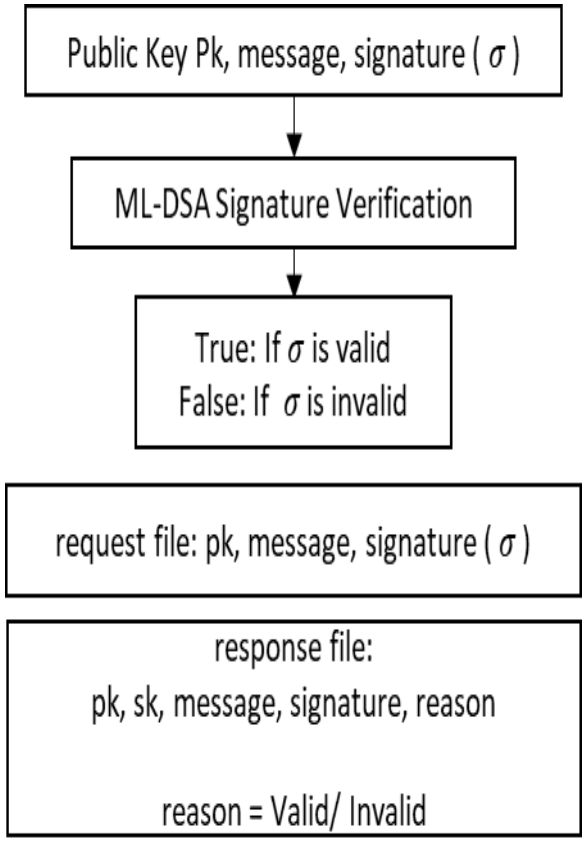
Key Generation



Signature Generation



Signature Verification



ML-DSA-44 Signature Verification

```
# Custom Test Vectors for ML-DSA-44
# ML-DSA-44 Signature Verification Response File
# Generated On: Monday, 01 September 2025 at 01:15:40 PM

count = 0
pk = fa9a819e87d67f57bcb1581f4e138908fbffa0688b672c56cd0975fd9a351381cddaaf0bb6d73a2d15450423fb5e9e285e5a4474dc39d951a84ecbe7ad16d4d6c3741d0a5fd14ce8faae2cdf1018953e823db24c746c92f7de
sk = fa9a819e87d67f57bcb1581f4e138908fbffa0688b672c56cd0975fd9a3513812b08f2d9a6d2b563dd1e0348c24ac78973ff88b70e11fc2018e63164d0085029dd55ab6e00b9c70621ee7e8cc75a9047edd617fef749947567
message = 62dfa50a4394b20d88948cb16c69dc94ae736cf6c9d4642151f2dc28f1c813a15a46f6d05ea4536c811f74c2e6e5a776f05479059643372dd421661f0a38d083
signature = d509fb7bfb0755051ce8283ab7e41795e80cdb3533dd00d3c72b9b3c06760e86cb4ce889022f11420d05702e6c66cb4a958cff75f0325563d6a6cbd5c8ea079464deec22065c63ecdfe3a2329d50cb290a4a334d600
signature verification passed

count = 1
pk = 9a2656b61ed5f49761929530b5255073f6263b2010f6223b9481c6a7b4b525049c81cf6a23a6b32618adaaa1501b6d48d234c4bcc98806935b615e1b391ad6ca89ac1b5d3c8c489a9afb17a216fc59ca77819a35e7daced9db
sk = 9a2656b61ed5f49761929530b5255073f6263b2010f6223b9481c6a7b4b52504793ed4ae97c31f638f77f57813269c74[illegible]fa733f76db553176dabf46ae1e3d44b8ace1c1c8c36522cb92961f53ad7b06e56bd7769b3e9aa4e4f
message = 8abcfb0a58cd4eb84ca7d2d272125b9534446e95f428729400de41d0eaacb465241c3af5d30106969bdbf44afa[illegible]7fc11acfa5022c729829c51bf7f6d05758a
signature = 72e319797656d17c8be024cad979f94521996a1061a751d86157aa2cb839cb80e911f9c10959e[illegible]502f0d8f82a0de38396b5ba5591769df002682173e9e175595ecc50d167fe9e2088efeae38a9f231
signature verification passed

count = 2
pk = d4075669560c50f86875ea8145258c7508dfa7fde54eb0edcda3a01a4f765[illegible]a071f7d5d153cb37ada51c3f4a1bd7692e50c3a28b2471fcd486091ce81624658d8be7a6eb557d7aec1bb30868817a4263
sk = d4075669560c50f86875ea8145258c7508dfa7fde54eb0edcda3a01a4f765[illegible]a071f7d5d153cb37ada51c3f4a1bd7692e50c3a28b2471fcd486091ce81624658d8be7a6eb557d7aec1bb30868817a4263
message = 2883db42c46b834b64ae7e7b856a6078b6e9437fe7e34a[illegible]2974a69ce7a61195f55aeadc002b284375202874d34ddd279d6dbb6f2
signature = 6e5c19062c85e17a129598fb88e3fc9810de0e6f9075[illegible]91e1a2fdbc6f69c61a7f82f0b9b59195e4bf7368f7d6f865e5195351b7d74fce9f01fbff80c51ad2efbc7af8d1193df078526a116c
signature verification failed

count = 3
pk = a601b70b69073b7476a649e1f338a[illegible]f2ec975f74776527882d049c4207de9afdf464d9d4e2db1e24916207c7a13cb0ba80f5c9f488e01b1e51bd7d42d6b69e1e2098f58e3c4a5d814927ce296b468dcb2
sk = a601b70b69073b7476a649e1f338a[illegible]a2d8f2ec975f747765278d909726df6406193a8f3a28b75af4399a96f2696aee56cedb34bfc61371f792644c4da64c183d4339578b4a3ee0cbbf47ccff05850835d9931
message = 61e734d76e74c0199618ce9d0[illegible]7185e7786b75c4c2f1501334e30edf3a54fdbb75acca69432a29a3b93797005d8168ed36c3d4e9d53bc261837316afa5
signature = ff440bf94e4aea1b844d469e8dc4bfa35d0783347c1098bd3c6a4934df53e125f232485f367a0c73e8f9920fb4dd3b464c0512c45451c85b7be3d2a189527bd7b3b857a61a2aa774104f13e717b3d1036c2e5c2e4de
signature verification passed

count = 4
pk = a3578f30fc95ba0feb0c5068746eacabe4337b67ebf0aff885647039ccbca95f9e928c3cd692aaefb1e048c1ed6f7073773df1f7a75e973e248850dc45f1b31e9983a1bc0bd351b2a06c549358a0a1d9fe90664d9d802bf410
sk = a3578f30fc95ba0feb0c5068746eacabe4337b67ebf0aff885647039ccbca95f52a0256e401b61c2ff681ed849909b5f4662c7035c075bd894b1ab6da5ab9a66c76e1e256cc0c4aafc07513ebed381a474be2f8b5f4edec3ba
message = f3f61ee26c7f55c1bdf931e4fd03842992c06c5492494a30be3ea59f0a21512d9707125cbe660fefec71277a372094c94d6b081bae72be8526c917fcda841424
signature = d46ed26217140ba99feee5db1ca38cc9d3631d842e7134cb069b99624673b4aaa6bc624faac754265468dd7581d9bd2ba23a12e59764706c0e22b046f6d6a15fd6317ac788f1e81cd81b3f177f9b2474c8c71d7ceb0
signature verification failed
```

References

1. National Institute of Standards and Technology (US), “Module-lattice-based digital signature standard,” National Institute of Standards and Technology (U.S.), Washington, D.C., NIST FIPS 204, Aug. 2024
2. ISO (International Organization for Standardization) and IEC (the International Electrotechnical Commission), “ISO/IEC 19790:2012(en), Information technology — Security techniques — Security requirements for cryptographic modules.
3. National Institute of Standards and Technology (US), “Digital Signature Standard (DSS),” National Institute of Standards and Technology (U.S.), Washington, D.C., NIST FIPS 186-5, Feb. 2023.

THANK YOU