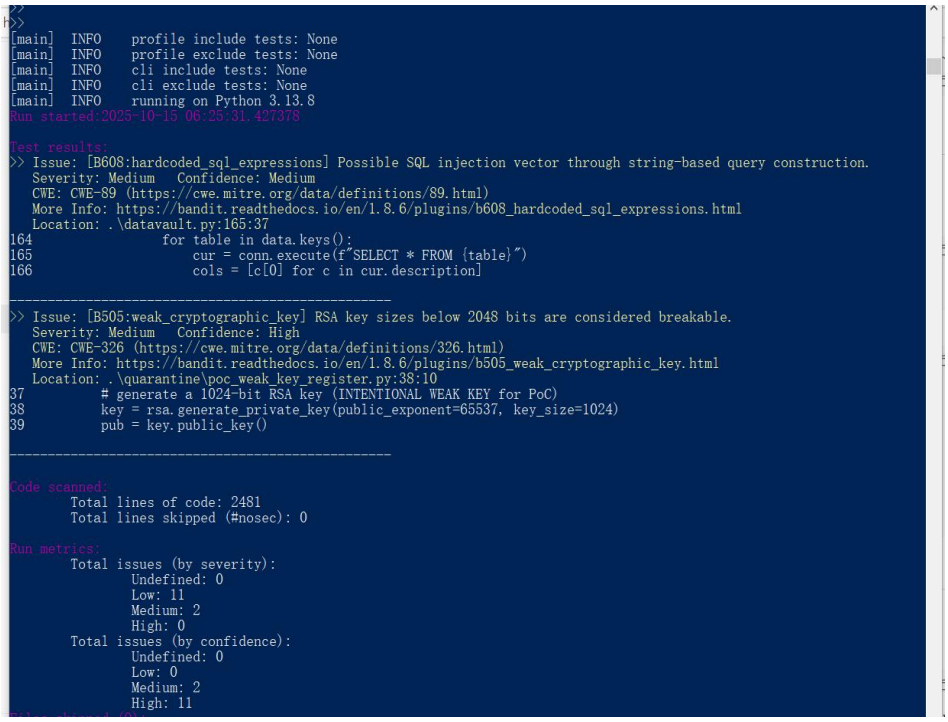
Reviewer name: Mingxu XUE a1933289

Group: project group 68

Testing approaches / tools used

Static analysis:



High-level findings

[High] Private key files (.pem) committed to the repository.

[High] Built-in backdoor flags (weak RSA keys and unsigned message acceptance).

[Medium] Non-standard password hashing (single SHA-256 with salt, no iterations).

Detailed findings, impact, and remediation

[High] Private keys included in repository

Impact: Private and public key pairs are stored in the '.keys/' directory, allowing impersonation and data decryption.

Remediation: Remove all keys from version control, regenerate keys securely at deployment, and add CI secret scanning.

[High] Backdoor flags enabled

Impact: Code defines BACKDOOR_WEAK_KEYS and BACKDOOR_TRUST_GOSSIP that bypass security validation.

Remediation: Disable these flags in production, separate them into a lab-only branch, and enforce strict key validation.

[Medium] Weak password hashing scheme
Impact: Single SHA-256 + salt used in datavault.py without iterations.
Remediation: Replace with Argon2id or PBKDF2 (≥200,000 iterations) and migrate existing hashes gradually.

Recommendations

1. Remove all private keys and regenerate securely using KMS or Vault.
2. Eliminate or isolate backdoor logic; require build-time flags for any lab use.
3. Adopt modern password hashing (Argon2id or PBKDF2).
4. Replace all try/except: pass with specific error handling and logging.
5. Enable TLS (wss://) for all client-server communication.