Cryptanalysis on Asymmetric Ciphers — RSA & RSA-Based Signatures

Lê Trí Đức
Phạm Nguyễn Thành Long

Ngày 20 tháng 10 năm 2025

Tóm tắt nội dung

Đề tài nghiên cứu các kỹ thuật tấn công và các điểm yếu khi triển khai thực tế trên RSA và các sơ đồ chữ ký dựa trên RSA (ví dụ PKCS#1 v1.5, RSASSA-PSS). Mục tiêu là khảo sát lý thuyết (factorization, low-exponent attacks, chosen-ciphertext attacks như Bleichenbacher), thực hiện các PoC an toàn trong môi trường kiểm soát (padding oracle, timing, RSA-CRT fault), đánh giá rủi ro trong các kịch bản triển khai (TLS, code signing, JWT/ID tokens, smartcards/HSMs), và đề xuất các biện pháp khắc phục và vận hành.

1 Mục tiêu học thuật & kỹ năng

1. Hiểu cơ chế toán học nền tảng của RSA (độ khó phân tích thừa số, modular exponentiation) và các biến thể chữ ký (PKCS#1 v1.5 vs RSASSA-PSS).
2. Khảo sát các lớp tấn công: factoring (GNFS overview), low-exponent attacks (Håstad), CCA (Bleichenbacher), padding oracle, Wiener's small-d, fault attacks trên RSA-CRT, timing và side-channel attacks.
3. Triển khai PoC an toàn trong lab: Bleichenbacher PoC, timing/leakage measurement, mô phỏng RSA-CRT fault.
4. Phân tích các điểm yếu khi RSA được dùng trong các hệ thống thực tế (TLS, JWT, code signing, smartcards) và đề xuất mitigations.
5. Viết báo cáo reproducible: PoC code, experiment logs, mitigation checklist.

2 Tính cấp thiết và động lực

RSA vẫn được sử dụng rộng rãi trong nhiều hệ thống mặc dù ngành đã khuyến nghị các kích thước khóa lớn hơn hoặc chuyển sang ECC/PQC. Nhiều cuộc tấn công thực tế khai thác lỗi triển khai (padding handling, timing leaks, flawed RNG) hơn là phá vỡ RSA về mặt toán học. Do đó, hiểu các vector tấn công thực tế giúp tổ chức đánh giá rủi ro và lập lộ trình migration.

3 Câu hỏi nghiên cứu và giả thuyết

1. RQ1: Trong kịch bản triển khai thực tế (TLS, JWT, code signing, HSMs), đâu là các điểm yếu phổ biến nhất liên quan tới RSA và chúng dẫn tới hậu quả gì (forgery, key recovery, signature replay)?

2. RQ2: Các tấn công Bleichenbacher (padding oracle) và timing attacks còn khả thi trên các stack hiện đại (OpenSSL/LibreSSL/BoringSSL) khi gặp cấu hình cũ hoặc lỗi không?

Giả thuyết: Các vụ compromise thực tế thường xảy ra do: xử lý padding không an toàn (PKCS#1 v1.5), RNG yếu tạo primes dễ đoán, misuse của small exponent. Việc chuyển sang RSASSA-PSS, áp dụng kích thước khóa lớn ($\geq 3072bit$), sử dụng HSM và library thực hiện constant-time sẽ giảm đáng kể rủi ro.

4 Tổng quan kĩ thuật

1. RSA primitives: quy trình tạo khóa (p, q là các số nguyên tố), $n = p \times q$, e là public exponent, d là private exponent. Các phép toán chính: encryption/signature $m^e \bmod n$ và decryption/verification $c^d \bmod n$.

2. RSA optimizations: CRT được dùng để tăng tốc decryption/signing nhưng mở ra bề mặt tấn công fault nếu không có countermeasures.

3. Các loại tấn công: factoring (GNFS), low exponent, Bleichenbacher padding oracle, timing, side-channel.

4. Signature schemes: PKCS#1 v1.5 (legacy) và RSASSA-PSS (hiện đại, có tính bảo đảm hình thức cao hơn).

5 Timeline

- Tuần 1–2: literature review và thiết lập lab (Docker images, SoftHSM).
- Tuần 3–4: triển khai Attack A (Bleichenbacher PoC) và thu thập metrics.
- Tuần 5–6: triển khai Attack B (timing) và đánh giá countermeasures.
- Tuần 7–8: triển khai Attack C (Wiener/low-d) và keygen experiments.
- Tuần 9: tổng hợp kết quả và chạy mitigation tests.
- Tuần 10: hoàn thiện báo cáo, repo reproducible, video demo và presentation.

6 References

- Bleichenbacher D., "Chosen Ciphertext Attacks against Protocols Based on the RSA Encryption Standard (PKCS#1)" — nền tảng cho padding oracle PoC.
- Kocher P., "Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS," — kinh điển về timing attacks và các countermeasures.
- Wiener M., "Cryptanalysis of short RSA secret exponents" — mô tả Wiener's attack trên small-d.
- Håstad J., các công trình về low-exponent / broadcast attacks. Boneh, các bài survey (ví dụ "Twenty Years of Attacks on the RSA Cryptosystem") để đặt bối cảnh lịch sử và các class attack.
- Tools gợi ý: OpenSSL (nhiều phiên bản), RsaCtfTool, SoftHSM, ChipWhisperer

7 Phân tích điểm yếu theo kịch bản triển khai

TLS / HTTPS

1. Bleichenbacher padding oracle: các server xử lý RSA PKCS#1 v1.5 cho PreMasterSecret (TLS $\leqslant$1.2) nếu trả về lỗi/độ trễ khác nhau sẽ tạo oracle cho việc phục hồi session secrets; TLS 1.3 loại bỏ RSA key exchange, nhưng các stack/cấu hình cũ vẫn tồn tại.
2. Khuyến nghị: tắt RSA key exchange, dùng ECDHE cho forward secrecy; sử dụng RSASSA-PSS cho chữ ký.

JWT / token signing (RS256)

1. Vấn đề: alg confusion hoặc xử lý alg/kid sai có thể dẫn tới forgery; RS256 mặc định tương ứng PKCS#1 v1.5 cho chữ ký — RSASSA-PSS nên ưu tiên cho thiết kế mới.
2. Khuyến nghị: kiểm tra chặt chẽ alg, validate JWK selection, rotate key thường xuyên.

Code signing & package ecosystems

1. Nguy cơ: máy build bị compromise hoặc lưu khoá không an toàn sẽ dẫn tới key leakage; một số hệ sinh thái chấp nhận định dạng chữ ký cũ dễ bị replay/malleability.
2. Khuyến nghị: lưu khoá trong HSM offline, kiểm soát build pipeline.

Implementation bugs & side channels

1. Timing leaks: modular exponentiation không constant-time; cần exponent blinding và constant-time libs. ư
2. CRT recombination faults: lỗi khi recombining CRT sẽ rò rỉ p hoặc q. Padding oracle: thông báo lỗi hoặc timing khác biệt tạo oracle.

8 Triển khai

Các thư viện / công cụ: OpenSSL (nhiều phiên bản), RsaCtfTool, MPrime/pari/gmp cho toán lớn, SoftHSM, Docker, Python (pycryptodome) và các script PoC cho từng attack.

9 Phương pháp nghiên cứu & thí nghiệm

- Các mục tiêu thí nghiệm (mỗi attack có PoC + measurement plan):
  - Attack A — Bleichenbacher padding oracle PoC: dựng server TLS-like dùng PKCS#1 v1.5 decryption cho PreMasterSecret, tạo oracle, phục hồi session secret; sau đó vá (uniform error, PSS) và kiểm chứng.
  - Attack B — Timing attack: đo local và remote timing để dò rò rỉ thông tin về d; đánh giá countermeasures (exponent blinding, constant-time lib).
  - Attack C — Wiener / low-d: sinh khóa vulnerable (small d) và dùng thuật toán Wiener để phục hồi private key.

- Lab setup & safety
  - Môi trường cô lập (isolated lab network), Docker images cho các phiên bản OpenSSL cũ nếu cần, SoftHSM cho PKCS#1 emulation, scripts tự động hóa PoC, và logging. Chỉ thực hiện PoC trên hệ thống lab và tuân thủ ethical rules của trường.

- Data collection & metrics
  - Success criteria: private key recovered? session secret recovered? signature forgery accepted? Effort metrics: số oracle queries, thời gian tấn công, tài nguyên tính toán, yêu cầu truy cập vật lý.
  - Đánh giá mitigation: xác minh rằng blinding/padding changes loại bỏ exploitability.

10 Biện pháp khắc phục và best practices

- Ưu tiên RSASSA-PSS cho chữ ký và tránh PKCS#1 v1.5 khi có thể.

- Dùng kích thước khóa lớn ($\geq$3072 bit) hoặc chuyển sang ECC/PQC cho bảo vệ dài hạn.

- Vô hiệu hóa RSA key exchange trong TLS, ưu tiên ECDHE cho forward secrecy.

- Áp dụng exponent & CRT blinding, constant-time modular exponentiation, và CSPRNG mạnh cho keygen.

- Lưu private key trong HSM/TPM, giới hạn API và dùng uniform error messages.

11 Mở rộng và hướng nghiên cứu tiếp theo

1. Khảo sát hệ sinh thái mã nguồn mở để đo prevalence của cấu hình vulnerable (scan có kiểm soát trên repo opensource).

2. Lập lộ trình migration sang ECC/PQC cho artifacts quan trọng (code signing, certificates, tokens).

12 Công cụ & tài nguyên gợi ý

OpenSSL, RsaCtfTool, MPrime/pari/gmp, SoftHSM, Docker, Python (pycryptodome), và script PoC cho Bleichenbacher, timing, Wiener, fault simulation.

13 Appendix: cấu trúc repository mẫu

```
project-root/
   docker/             # containers cho các phiên bản OpenSSL vulnerable, SoftHSM
   poc/                # PoC scripts cho Bleichenbacher, timing, Wiener, fault sim
   tools/              # wrappers (RsaCtfTool configs), measurement harness
   docs/               # report, slides, runbook, responsible disclosure template
   logs/               # experiment outputs (lưu non-sensitive)
```