The RSA one way permutation

- Parameters: N=pq. N ≈1024 bits. p,q ≈512 bits.
  e – encryption exponent. gcd(e, φ(N)) = 1 .
- Permutation: **RSA(M) = M^e** (mod N) where $M \in Z_N$

- Trapdoor: **d** – decryption exponent.
  Where $e \cdot d = 1$ (mod φ(N))
- Inversion: **RSA(M)^d** = $M^{ed} = M^{k\varphi(N)+1}$ = **M** (mod N)

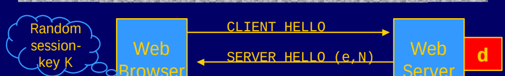
- "Assumption":
  no efficient alg. can invert RSA without trapdoor.

Textbook RSA is insecure

- Textbook RSA encryption:
  - public key: **(N,e)** Encrypt: **C = M^e** (mod N)
  - private key: **d** Decrypt: **C^d = M** (mod N)
    (**M** Î Z_N)
- Completely insecure cryptosystem:
  - Does not satisfy basic definitions of security.
  - Many attacks exist.

- The RSA one-way permutation is not a cryptosystem.

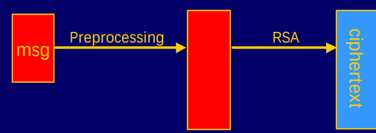
A simple attack on textbook RSA



- Session-key K is 64 bits. View $K \in \{0,\dots,2^{64}\}$
  Eavesdropper sees: $C = K^e$ (mod N) .
- Suppose **$K = K_1 \cdot K_2$** where $K_1, K_2 < 2^{34}$. (prob. ≈20%)
  Then: **$C/K_1^e = K_2^e$** (mod N)
- Build table: $C/0^e, C/1^e, C/2^e, C/3^e, \dots, C/2^{34e}$. time: 2^{34}
  For $K_2 = 0,\dots, 2^{34}$ test if K_2^e is in table. time: $2^{34} \cdot 34$
- Attack time: $\approx 2^{40} \ll 2^{64}$

Common RSA encryption

- Never use textbook RSA.
- RSA in practice:



- Main question:
  - How should the preprocessing be done?
  - Can we argue about security of resulting system?

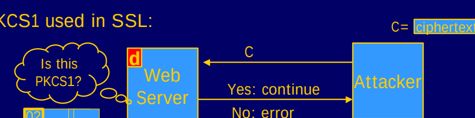
PKCS1 V1.5

- PKCS1 mode 2: (encryption)

- Resulting value is RSA encrypted.
- Widely deployed in web servers and browsers.
- No security analysis !!

Attack on PKCS1

- Bleichenbacher 98. Chosen-ciphertext attack.
- PKCS1 used in SSL:

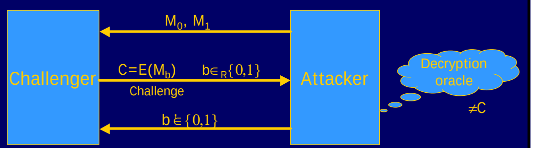


⇒ attacker can test if 16 MSBs of plaintext = '02'.

- Attack: to decrypt a given ciphertext C do:
  - Pick random $r \in Z_N$. Compute $C' = r^e \cdot C = (rM)^e$.
  - Send C' to web server and use response.

Chosen ciphertext security (CCS)

- No efficient attacker can win the following game:
  (with non-negligible probability)



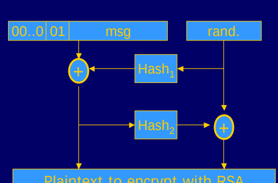
Attacker wins if b=b'

Chosen-ciphertext secure RSA

- Are there CCS cryptosystems based on RSA?
  - RSA-PKCS1 is not CCS !
- Answer: Yes! Dolev-Dwork-Naor (DDN). 1991.
  - Problem: inefficient.
- Open problem: efficient CCS system based on RSA.
- What to do? Cheat!
  - Build RSA system that is CCS in imaginary world.
  - "Assume" our-world = imaginary-world.

PKCS V2.0 - OAEP

- New preprocessing function: OAEP (BR94).
- RSA one-way permutation ⇒ RSA-OAEP is CCS
  when Hashes are "*random oracles*".



$Hash_1$, $Hash_2$ assumed to be random oracles.

SHA1?? MD5??

Improving RSA's performance

- To speed up RSA decryption use small private key d. $M^d = C \pmod N$
  - Wiener87: if $d < N^{0.25}$ then RSA is insecure.
  - B98: if $d < N^{0.292}$ then RSA is insecure (open: $d < N^{0.5}$)
  - Insecure: priv. key d can be found from (N,e).
  - Small d should never be used.

Wiener's attack

- Recall: $e\cdot d = 1 \pmod{\varphi(N)}$

$$\Rightarrow \exists k \in Z : \quad e\cdot d = k\cdot\varphi(N) + 1$$

$$\Rightarrow \left| \frac{e}{\varphi(N)} - \frac{k}{d} \right| \le \frac{1}{d\varphi(N)}$$

$\varphi(N) = N\text{-}p\text{-}q+1 \Rightarrow |N - \varphi(N)| \le p+q \le 3\sqrt{N}$

$d \le N^{0.25}/3 \Rightarrow \left| \frac{e}{N} - \frac{k}{d} \right| \le \frac{1}{2d^2}$

Continued fraction expansion of e/N gives k/d.

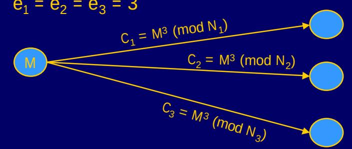
$e\cdot d = 1 \pmod k \Rightarrow \gcd(d,k)=1$

Low public exponent

- To speed up RSA encryption (and sig. verify) use a small e. $C = M^e \pmod N$
- Minimal value: e=3 ($\gcd(e, \varphi(N)) = 1$)
- Recommended value: $e=65537=2^{16}+1$
  Encryption: 17 mod. multiplies.
- Several weak attacks. Non known on RSA-OAEP.
- Asymmetry of RSA: fast enc. / slow dec.

Example: broadcast attack

- A user sends the encryption of M to three different people.
  Their RSA public keys are (N_1,e_1) (N_2,e_2) (N_3,e_3)
  where $e_1 = e_2 = e_3 = 3$



- Let $N = N_1 \cdot N_2 \cdot N_3$. Observe that $M^3 < N$.
- Using CRT Eve can build $C \in Z_N$ s.t. $C = C_i \pmod{N_i}$.
- But then, $C = M^3 \pmod{N}$, i.e. $C = M^3$.
- So, Eve can find M by computing cube root of C .

Implementation attacks

- Attack the implementation of RSA.
- Timing attack: (Kocher 97)
  The time it takes to compute $C^d \pmod{N}$ can expose d.
- Power attack: (Kocher 99)
  The power consumption of a smartcard while it is computing $C^d \pmod{N}$ can expose d.
- Faults attack: (BDL 97)
  A computer error during $C^d \pmod{N}$ can expose d. One error is enough.

Key lengths

- Security of public key system should be comparable to security of block cipher.

NIST:

Cipher key-size	Modulus size
≤ 64 bits	512 bits.
80 bits	1024 bits
128 bits	3072 bits.
256 bits (AES)	**15360** bits

- High security ⇒ very large moduli.
  Not necessary with elliptic curves.

Future...

- Low-public exponent RSA is excellent for digital signatures.
  - Good for certificate management.
  - Public Key Infrastructure (PKI)
- Key exchange/Authentication is difficult with RSA on small devices and loaded servers.
  - PalmPilot: RSA sig. gen: **30 sec**. (1024 bit)
    RSA sig. ver: 0.7 sec (1024 bit, e=3)