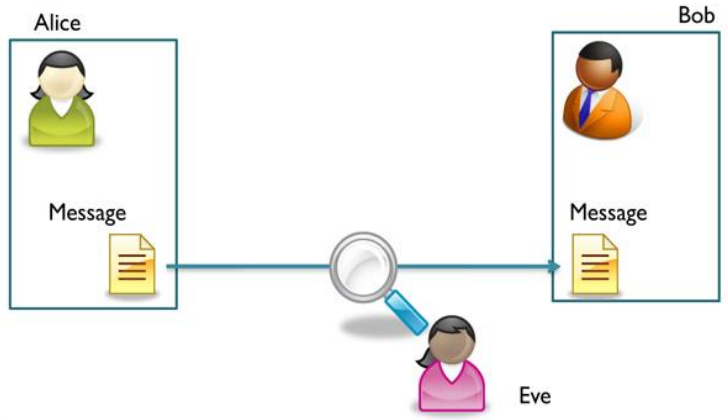
DETERMINISTIC VS PROBABILISTIC

- Textbook RSA (Deterministic):
  - Always same output for given input
  - Function(“Hello”) = 2345
  - RSA(2345) = 184
  - Eve can analyze RSA(“Hello”) = 184 by observation

- RSA with Padding (Probabilistic):-
  - Pad input with random hash
  - Input Function(“Hello”) + Hash(Random string) = 5677
  - RSA(5677) = 091
  - Since RSA(“Hello”) not always = 184, Eve cannot break the scheme



RSA ALGORITHM

- Rivest-Shamir-Adleman (RSA) Algorithm:
  - Choose 2 large prime numbers (p and q), and calculate ***n*** = p*q
  - Calculate Phi(***n***) = (p-1)(q-1)
  - Choose a number ***e***, such that GCD(e, Phi(***n***)) = 1
  - Choose d such that; ***e***.***d*** ≡ 1. mod Phi(***n***) i.e., d = e^{-1}
- Now, we can perform encryption and decryption as follows:
  - Encryption:- c = $(m)^e$ mod n
  - Decryption:- m = $(c)^d$ mod n

PRIME NUMBER THEOREM

- Probability of *odd number p* being prime = $2 / \log_e(p)$
- Probability for bit lengths generated:

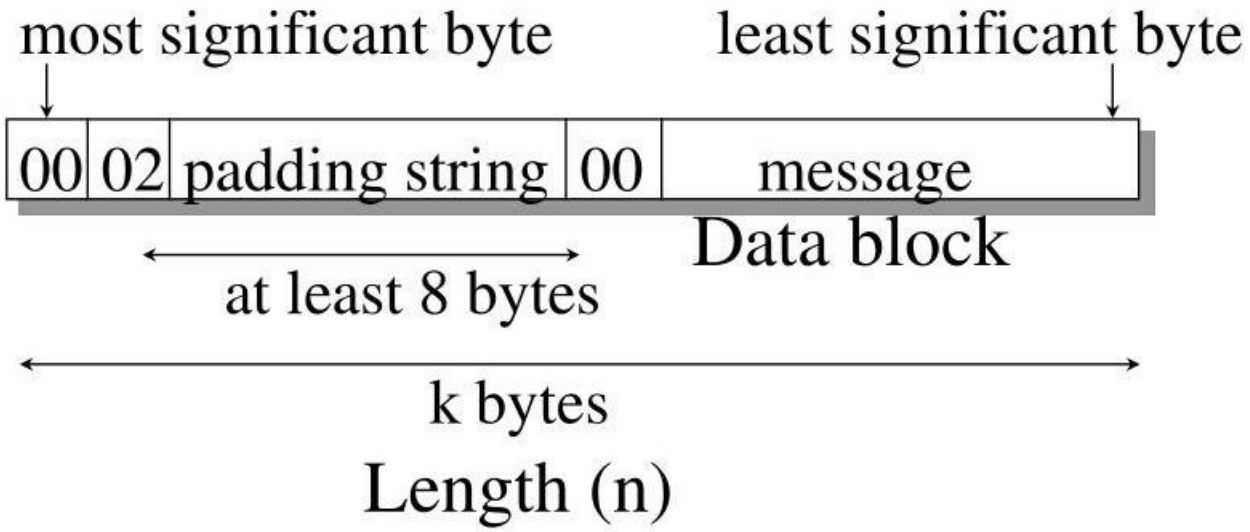
Bit length	Number of tries
512	177
647	224
768	266
813	292
1024	355
2048	710

Prime number 1024-bit long = 2^{1024}

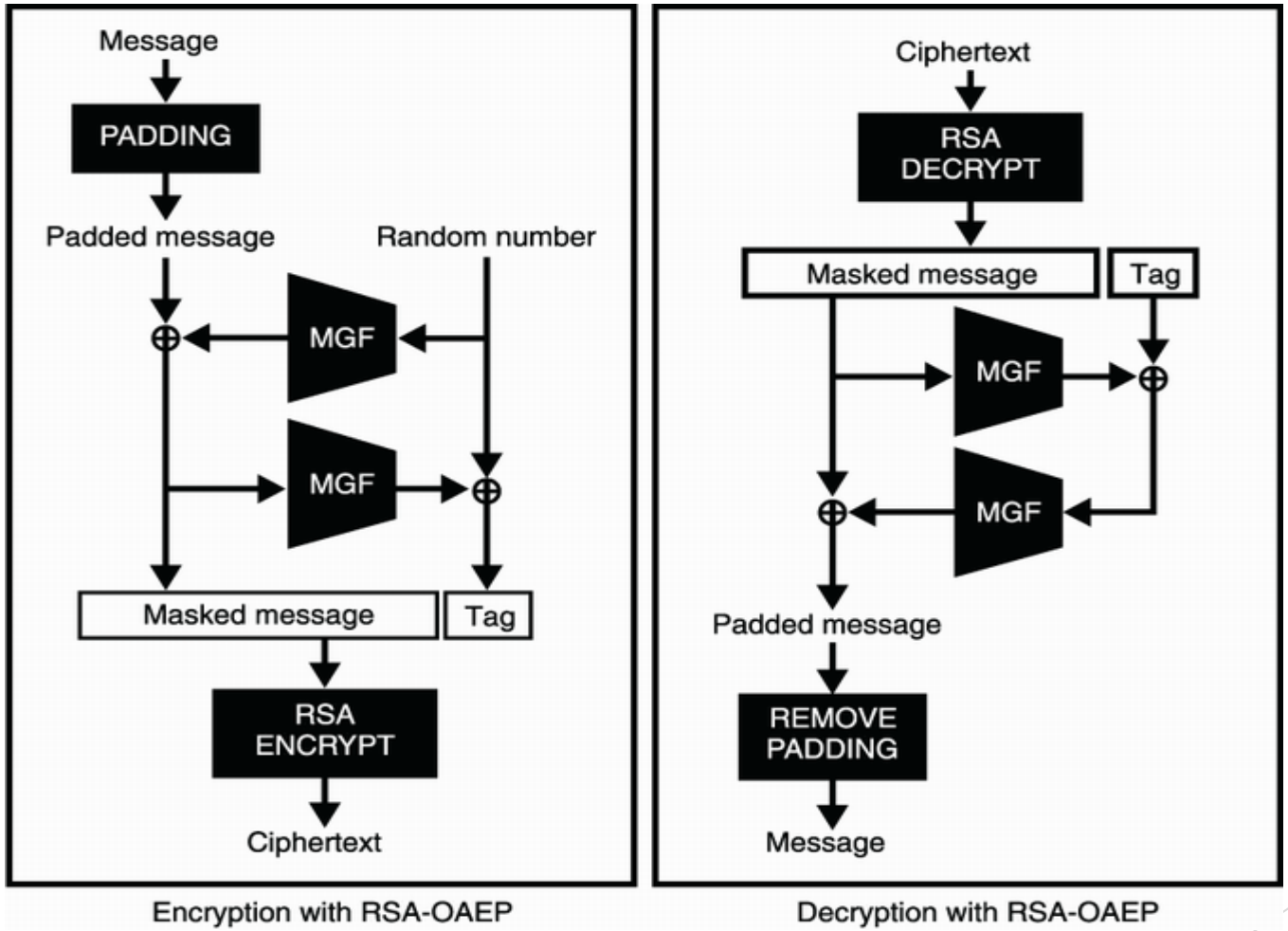
$P(2^{1024} \text{ is prime})$
$= 2 / \log_e(2^{1024})$
$= 2 / 1024 \log_e(2)$
$= 2 / 1024 * 0.693$
$= 1 / \text{approx}(355)$

Thus, about 355 1024-bit odd numbers need to be checked for obtaining a prime number.

Public key Cryptography Standards (PKCS) #1 v1.5



OPTIMAL ASYMMETRIC ENCRYPTION PADDING (OAEP)



REFERENCES

- https://external-content.duckduckgo.com/iu/?u=https%3A%2F%2Finterstices.info%2Fupload%2Fcodes-secrets%2Falice-et-bob1.jpg&f=1&nofb=1
- https://upload.wikimedia.org/wikipedia/commons/7/70/EME-OAEP.jpg
- https://image.slideserve.com/1301747/pkcs-1-v-1-5-padding-for-encryption-l.jpg