1. 100 bit msg. $\longrightarrow$ 2^{100} possible msgs.
16 bit MAC $\longrightarrow$ 2^{16} possible MACs
32 bit key. $\longrightarrow$ 2^{32} possible keys.

each MAC is generated by a total of
$2^{100}/2^{16} = 2^{84}$ different msgs. on avg.

Here key size $K = 32 > n$ = MAC size.
On average α rounds will be needed
where $K = \alpha \times n$
$\therefore \alpha = K/n = 32/16 = 2$

no. of rounds required.

On average, $2^{K-n} = 2^{32-16} = 2^{16}$ keys will produce a match.

2. $M = X_1 \| X_2 \| \cdots \| X_m$
concatenation of 64 bit blocks &
MAC uses encryption with $K = 56$ bit (key size)
$n = 64$ bit (tag size)

$$\Delta(M) = X_1 \oplus X_2 \oplus \cdots \oplus X_m$$
$$MAC(K, M) = E(K, \Delta(M))$$

If opponent observes $\{M \| MAC(K, M)\}$, a brute force attempt to determine K will require 2^{56} encryption (at least).
But, easier way is to replace X_1 to X_{m-1} with any desired values Y_1 to Y_{m-1} & replace X_m with
$$Y_m = Y_1 \oplus Y_2 \oplus \cdots \oplus Y_{m-1} \oplus \Delta(M).$$

The opponent can concatenate the new message $Y_1 || Y_2 || \dots || Y_m$ using the original tag to form a new message which will be accepted as authentic by the receiver.

By this process, any message of length $64(m-1)$ bits can be fraudulently inserted.

3. MD5 ⇒ 64 bit MAC.
   ↳ 128-bit hash fn.

- to attack MD5 attacker can choose any set of messages and work on these offline to find a collision.
- Attacker knows hash algo & default IV
  → can generate hash code for each of the messages generated by him.
- However, when attacking HMAC, the attackers cannot generate message/code pairs offline because attacker does not know K.
  → Attacker need to observe sequence of messages generated by HMAC under the same key. & perform attack on these known messages.

→ Hash code 128 bits ⇒ 2^{64} observed blocks
⇒ 2^{72} bits generated by same key.

→ On 1 Gbps line it will take ~ 150000 yrs. to succeed if observed a continuous stream of messages with no change in key.

4. RSA digital signature scheme:

$p = 823, \quad q = 953, \quad e = 313, \quad d = 160009$

$n = p \times q = 823 \times 953 = 784319$

$\phi(n) = (p-1)(q-1) = 822 \times 952 = 782544$

$e \times d = 1 \bmod \phi(n)$

$\{n, e\} \Rightarrow$ public key $\quad d \Rightarrow$ private

$M' = M \pmod n \qquad S = M^d \bmod n$

Signature:

$$\begin{cases} M_1 = 24019 \Rightarrow S_1 = (24019)^{160009} \bmod n \\ M_2 = 70190 \Rightarrow S_2 = (70190)^{160009} \bmod n \end{cases}$$

Verification:

$$\begin{cases} S_1^e \bmod n = \left((24019)^{160009}\right)^{313} \bmod n \\ S_2^e \bmod n = \left((70190)^{160009}\right)^{313} \bmod n \\ \qquad = (70190)^{160009 \times 313} \bmod n \\ \qquad = 70190 \bmod n \end{cases}$$

Known message attack:

$M = (M_1 \times M_2) \bmod n$

$= (24019 \times 70190) \bmod n$ ← New message

Forged signature ⇒

$S_1 = (S_1 \times S_2) \bmod n$

$= (24019 \times 70190)^{160009} \bmod n$

This is valid as per RSA

5. Zero knowledge Proof (Fiat-Shamir Protocol)

$p = 569$, $q = 683$, $s = 157$

public key $v = s^2 \bmod n$

$n = p \times q = 569 \times 683 = 388627$

$v = (157)^2 \bmod = 24649$

$r \Rightarrow$ a random no.

Sender	Receiver
1. $x = r^2 \bmod n$	
2. witness x ⟶	
	3. challenge e ⟵
4. $y = r \times (157)^e \bmod n$	
5. Response y ⟶	
	6. $y^2 \bmod n$
	check if $y^2 \bmod n = x v^e \bmod n$

Probability of fooling 15 times in a row

$= \left(\frac{1}{2}\right)^{15}$

6.

7. Diffie-Hellman Protocol

$q = 7,\ p = 23,\ x = 3,\ y = 5$

Alice: $R_1 = q^x \bmod p = 7^3 \bmod 23 = 343 \bmod 23 = 21$

Bob: $R_2 = q^y \bmod p = 7^5 \bmod 23$

Alice $\xrightarrow{R_1}$ Bob

Alice $\xleftarrow{R_2}$ Bob

$K = (R_2)^x \bmod p = (7^5)^3 \bmod 23$

$= (R_1)^y \bmod p = (7^3)^5 \bmod 23$