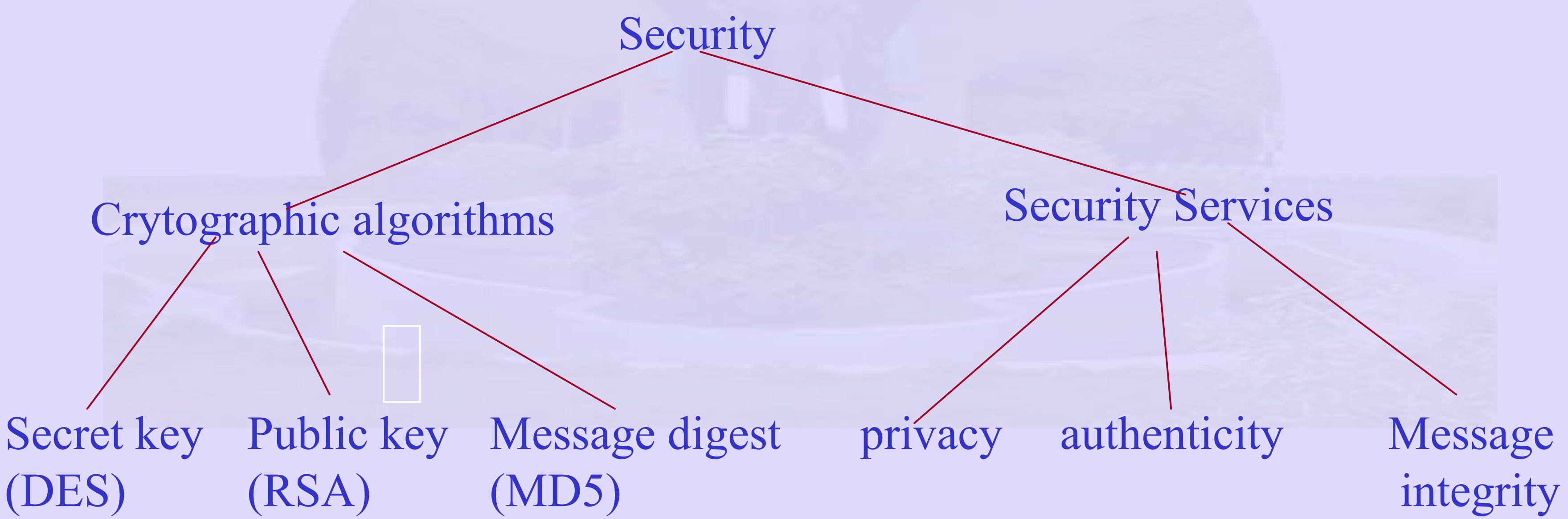
Network Security



Secret Key Encryption

Public Key Encryption

- Each participant has a secret key (private key)
- The key is not stored
  - Publish on the web (for instance)
- To send a message
  - Encrypt with public key
  - To decrypt, decrypt using a private key

Message Digest Encryption

- Map a potentially large message into a small fixed length number
- Compute checksum for message
- Given cryptographic checksum
  - Difficult to figure out the message

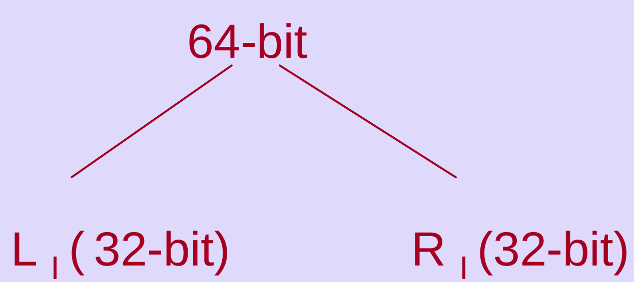
DES (secret key encryption)

- Block cipher (operates on a fixed block of bits)
- Encrypts a 64-bit of plain text using a 64-bit key
  - Only 56 bits used
  - Last bit of every byte is a parity bit
- Three phases in DES
  - 64-bits in each block are permuted

DES (secret key encryption)

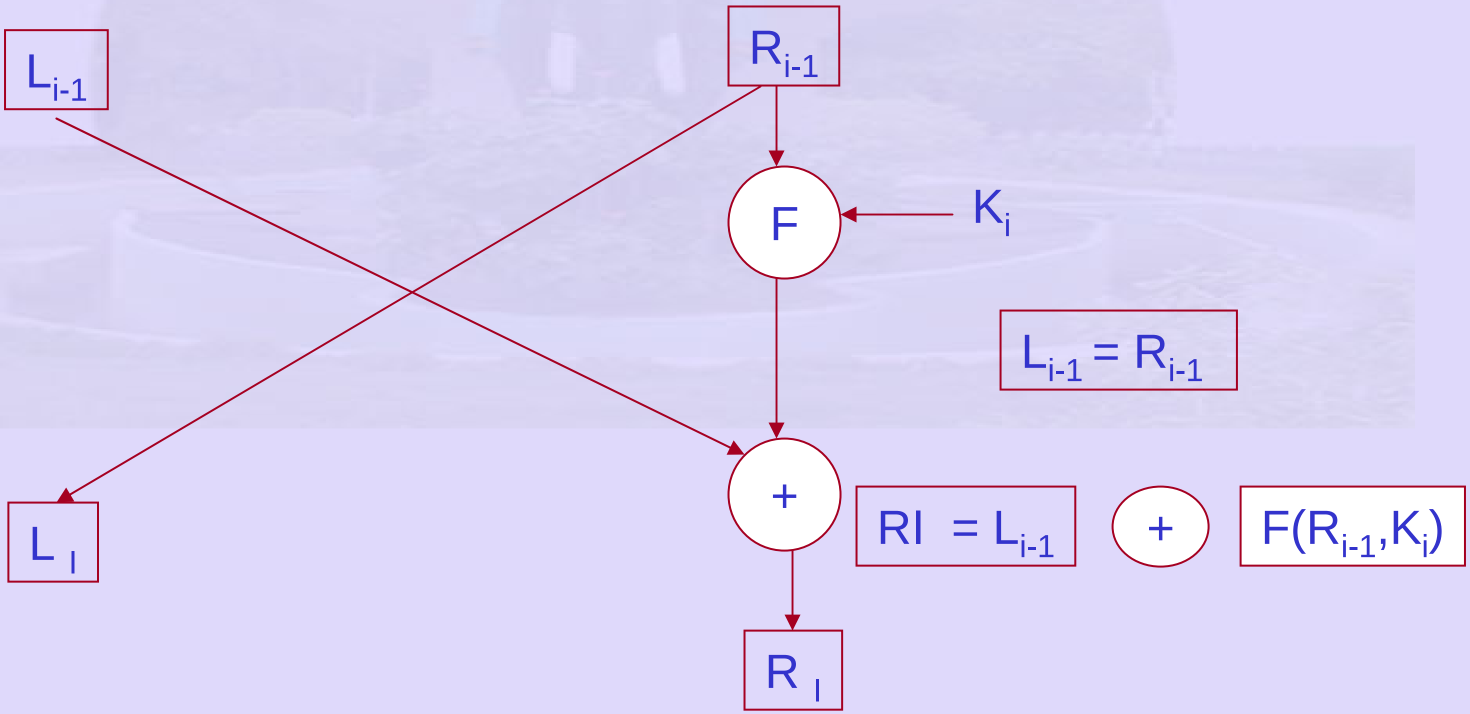
- – Sixteen rounds of an identical operation are applied to the resulting data and key
- – The inverse of the original operation is applied to the result

- During each round – split 64-bit into two 32-bit blocks



DES (secret key encryption)

– Choose 48-bit from 56-bit key



DES (secret key encryption)

- Define F, generate K_i
- Initially the permuted 56-bit key is divided into two blocks of 28-bit
  - Ignore every 8^{th} bit in original key
  - Each half is rotated 1/2 bits depending upon the round
  - A table is used to define the rotation of the 28-bit

DES (secret key encryption)

- DES compression permutation
  - 48-bit key is permuted and then used in the current round as key
- Function F combines 48-bit key (K_i) with the right half of data after round i-1 (R_{i-1})
- Expand R from 32-bit to 48-bit
  - Divide R into 4-bit chunks
  - Expand each chunk into 6-bit

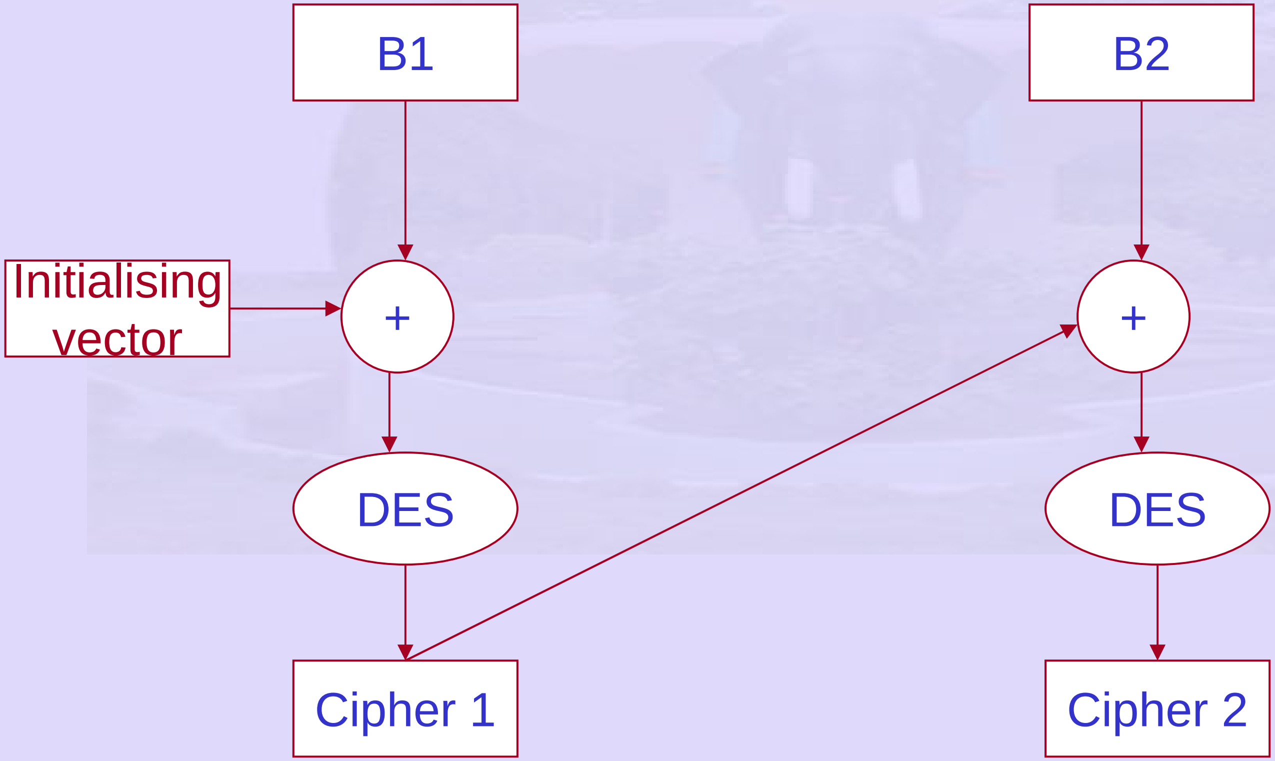
DES (secret key encryption)

  - 1-bit from left, 1-bit from right
  - 1st and last bit –use circular shift – they get from each other
- Divide 48-bit into 6-bit chunks
- XOR expanded R
- Finally pass 6-bit through substitution box to get 4-bit from 6-bit

DES (Decryption)

- Algorithm works exactly the same as that of encryption
- Apply keys in reverse
  - K_{16}, K_{15}, K_{14}, …, K_1
- Encryption of large messages
  - Cipher block chaining

Cipher Block Chaining



Public Key Encryption (RSA)

- Choose two large prime numbers p and q (typically greater than 10^{100}
- Choose
  - $n = p \times q$
  - $z = (p-1) \times (q-1)$
- Choose a number d relatively prime to z
  - z and d are coprimes – GCD (z,d) = 1
- Find e s.t. $e \times d = 1 \bmod z$

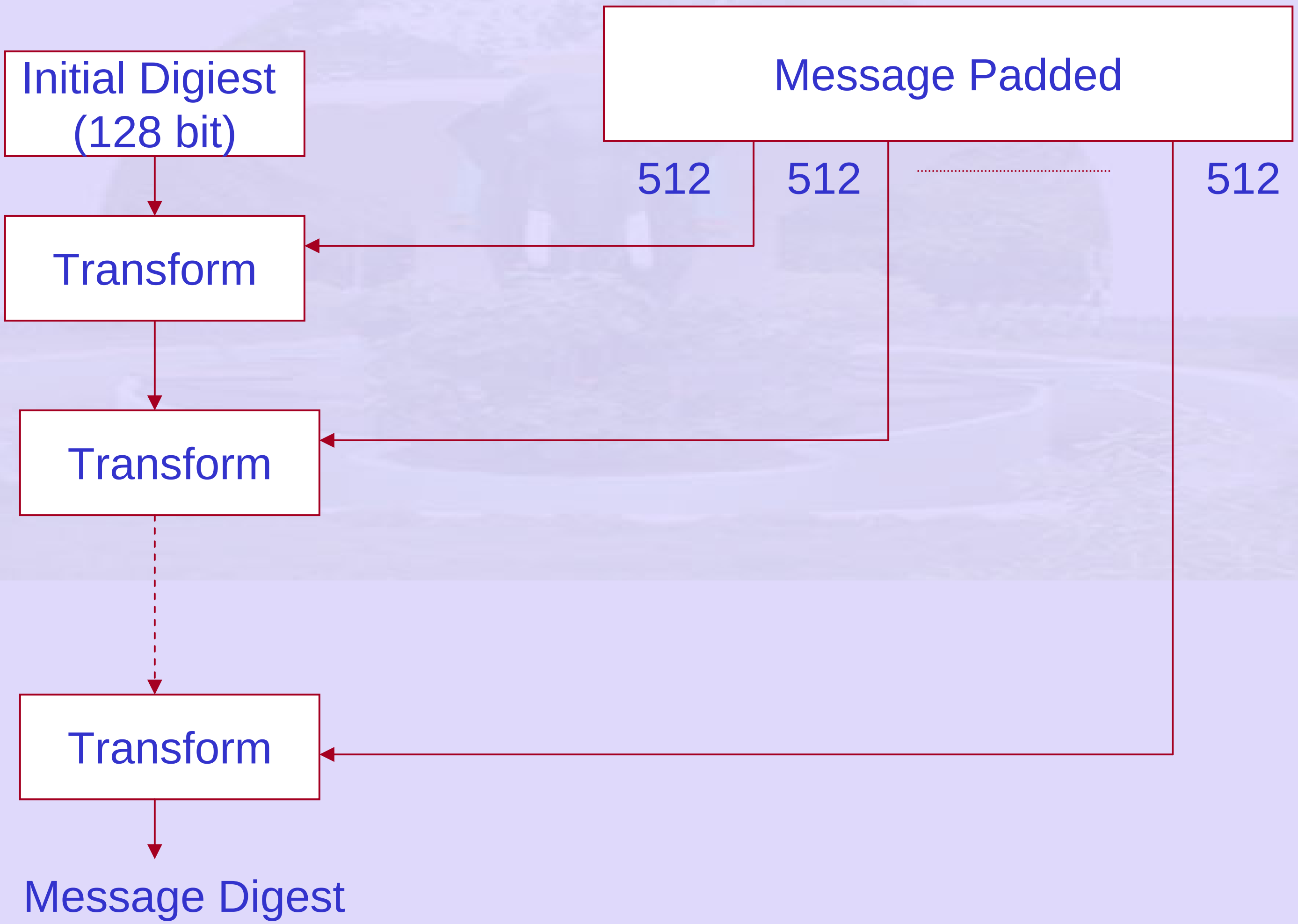
Public Key Encryption (RSA)

- Compute these parameters in advance
- Divide plaintext into blocks s.t. each plaintext is $0 \leq P < n$
  - i.e group bits such that (if k-bits) $2^k < n$
- To encrypt P, compute
  - $c = P^e \pmod{n}$
- To decrypt C, compute
  - $P = c^d \pmod{n}$

Public Key Encryption (RSA)

- To encrypt
  - e, n required (public key)
- To decrypt
  - c, n required (private key)
- Analogy
  - Suitcase with a press lock that is unlocked
    - Anybody can put stuff inside and lock the suitcase
    - But suitcase can ONLY be opened by the key

Message Digest



Message Digest

- Modern day: Operates on 32-bit quantities
- Current digest (d_0, d_1, d_2, d_3)
- Works on the hope that it is difficult to create the *transformations* and the *initial digest.*