12 Conclusion

Cryptography
Security objectives addressed by cryptography:

- Confidentiality
  - symmetric cryptography
  - asymmetric cryptography
- Integrity
  - hashing
- Authentication and non-repudiation
  - digital signatures

Cryptography
Types of attack:

Ciphertext only attack: ciphertext known to the adversary (eavesdropping)

Known plaintext attack: plaintext and ciphertext are known to the adversary

Chosen plaintext attack: the adversary can choose the plaintext and obtain its encryption (for example, has access to the encryption system)

Chosen ciphertext attack: the adversary can choose the ciphertext and obtain its decryption

Historical Ciphers
Monoalphabetic:

- Shift Cipher
- Substitution Cipher

Polyalphabetic:

- Vigenère Cipher
- Vernam Cipher (one-time pad)
- Rotor Machines

Security of Ciphers
Computational security: best known algorithm for breaking cipher requires an unreasonably large amount of computer time.

- Shift cipher, substitution cipher and Vigenère Cipher are not computationally secure.
- Block ciphers such as AES and public-key ciphers such as RSA are computationally secure.

Unconditional security (perfect secrecy, information-theoretical security, semantic security): no bound on the computational power of the adversary.

- Block ciphers such as AES and public-key ciphers such as RSA are not unconditionally secure.
- Vernam cipher (one-time pad) is unconditionally secure if used correctly.

Stream Ciphers
Encrypt an arbitrary stream of data by combining it with a keystream.
Examples: A5/1, RC4.
Keystream can be generated by a cryptographically secure pseudo-random generator (PRG) seeded with the key and should:

- Look random
- Be unpredictable
- Have large linear complexity
- Have low correlation between key bits and keystream bits.

The linear congrentual generator is not cryptographically secure, but the Blum Blum Shub generator is.
Most stream ciphers are based on a non-linear combination of LFSRs (Linear Feedback Shift Registers).

Block Ciphers
Plaintext is divided into blocks of fixed length and every block is encrypted one at a time.
DES:

- S-P Network, iterated cipher, Feistel structure
- 64-bit block size, 56-bit key size
- 8 different S-boxes
- non-invertible round
- design optimised for hardware implementations

AES:

- S-P Network, iterated cipher
- 128-bit block size, 128-bit (192, 256) key size
- one S-box
- invertible round
- design optimised for byte-orientated implementations

Block vs Stream Ciphers
Block ciphers:

- More versatile: can be used as stream cipher.
- Standardisation: DES and AES + modes of operation.
- Very well studied and accepted.

Stream ciphers:

- Easier to do the maths.
- Either makes them easier to break or easier to study.
- Supposedly faster than block ciphers (less flexible).

Modes of Operation
Block ciphers can be used in different modes of operation:

- Electronic Code Book (ECB mode)
- Cipher Block Chaining (CBC mode)
- Output Feed Back (OFB mode)
- Cipher Feed Back (CFB mode)
- Counter (CTR mode)

Hash Functions
A hash function is an efficient one-way function mapping binary strings of arbitrary length to binary strings of fixed length called the hash-value or digest.
A hash function should have the following properties:

- Pre-image resistant: given a digest, it should be computationally infeasible to find a piece of data that produces the digest.
- Weakly collision-free or second pre-image resistant: given message M it is computationally infeasible to find a different message M' such that $H(M) = H(M')$.
- Strongly collision-free: it is computationally infeasible to find different messages M and M' such that $H(M) = H(M')$.

Hash Functions
The birthday paradox: if there are n possibilities then on average $\sqrt{n}$ trials are required to find a collision.
Most practical hash functions make use of the Merkle-Damgård construction.
Applications of hash functions: message authentication, digital signatures, password storage, key generation, pseudorandom number generation, intrusion detection, virus detection.
Example hash functions: MD2, MD4, MD5, SHA-1, SHA-2, SHA-3, RIPEMD, RIPEMD-160.

MDCs and MACs
Used to ensure integrity of data.
Manipulation Detection Code (MDC): hash function without key.
Message Authentication Code (MAC): hash function with secret key.
Types of MAC:

- MACs based on block ciphers in CBC mode (CBC-MAC).
- MACs based on MDCs (e.g. HMAC).
- Customized MACs.

Public Key Cryptography
Each user has a key pair, which consists of a public key (made public, used for encryption) and a private key (kept secret, used for decryption).
Make use of number-theoretic problems to implement trapdoor one-way functions.
Example one-way functions:

- Multiplication
  - Inverse problem: factoring
- Modular exponentiation
  - Inverse problem: discrete logarithm

Public Key Cryptography
Hard Problems:

- Integer factorisation problem
- RSA problem
- Quadratic residuosity problem
- Square root problem
- Discrete logarithm problem
- Diffie-Hellman problem

- Decisional Diffie-Hellman problem

We can compare the relative difficulty of some of these problems using reductions.

Public Key Cryptography
Public key algorithms:

- Key exchange: Diffie-Hellman
- Encryption: encrypt using public key, decrypt using private key.
- Digital signature: encrypt using private key, decrypt using public key.
  - Schemes with message recovery.
  - Schemes with appendix.

Public Key Cryptography: Encryption
RSA
Encryption and decryption using modular exponentiation.
Encryption can be made more efficient using the square and multiply algorithm, and selecting an encryption exponent with very few bits set e.g. $2^{16}+1$.
Decryption can be made more efficient by making use of the knowledge of prime factors of the modulus and using the Chinese Remainder Theorem.
The security of RSA relies on the difficulty of the integer factorisation problem.
Raw RSA is not semantically secure, so a padding scheme should be used to add randomness and redundancy.

Public Key Cryptography: Encryption
Other public key algorithms:

- ElGamal
  - Security relies on the difficulty of the discrete logarithm problem.
  - Used as the basis of the Digital Signature Algorithm (DSA).
- Rabin
  - Security relies on the difficulty of the square root problem with a composite modulus.
  - Can be viewed as similar to RSA with an encryption exponent of 2.

Main drawback of public key cryptography is the inherently slow speed.
Therefore, public key schemes are not used directly for encryption.
Instead, they are used in conjunction with secret key schemes.

- Encryption is performed by secret key schemes (e.g. AES).
- Key agreement is performed by public key schemes (e.g. RSA or Diffie-Hellman).

Public Key Cryptography: Digital Signatures
Public key algorithms for digital signatures:

- RSA
  - Deterministic signatures: for each message, one valid signature exists
  - Faster verifying than signing
- ElGamal
- DSA: based on ElGamal
  - Non-deterministic signatures: for each message, many valid signatures exist
  - Faster signing than verifying
- Blind signatures
  - Signing of message without learning anything about it
  - Can be used in online elections

Key Exchange and Authentication Protocols
Secure communications protocols normally have a number of phases:

1. Authentication Phase
2. Key-Exchange Phase
3. Data Transfer Phase

Authentication techniques:

- Passwords
- Symmetric ciphers
- Asymmetric ciphers
- Zero knowledge Proof

Key exchange techniques:

- Key exchange
- Symmetric ciphers
- Asymmetric ciphers

Key Exchange and Authentication Protocols
Protocols for authentication:

- Needham-Shroeder Public-Key Protocol
- Fiat-Shamir Protocol
- X.509 Authentication Protocols

Protocols for key exchange:

- Diffie-Hellman Key-Agreement Protocol

Protocols for authentication and key exchange:

- Needham-Shroeder Secret-Key Protocol
- Otway-Rees Protocol
- Kerberos
- Diffie-Hellman Key-Agreement embedded in Needham-Shroeder Public-Key Protocol

Key Exchange and Authentication Protocols
Attacks on protocols:

- Replay attacks
- Reflection attacks
- Interleaving attacks
- Chosen plaintext attacks
- Man-in-the-middle attacks

Real World Protocols
Application Layer

- Secure e-mail (PGP, S/MIME)
- SSH

Transport Layer

- SSL
- TLS

Network Layer

- IPSec

Link Layer

- WEP
- WPA
- WPA2

Digital Cash
SET (Secure Electronic Transactions)

- Confidentiality: dual signatures
- Integrity: hashing
- Authenticity: digital signatures

Bitcoin

- Secure transactions
- Protection against double-spending
- Anybody can be a “merchant” or a “customer”
- Pseudo-anonymity
- Digital signatures initiate the transaction
- All transactions are in the blockchain
- Miners verify the transactions