## 

## [3413ICT Network Security](file:///C:\Documents%20and%20Settings\s995689\My%20Documents\Teaching\Courses_2013\Courses_2003\6216INT_03\6216inthome.html)

### **Workshop 3B**

**Part 1 – Reviewing the lecture notes, answer the following questions**

1. What requirements should a hash function meet for information security services?
2. Can be applied to any sized message **M**
3. Produces fixed-length output **h**
4. Is easy to compute **h=H(M)** for any message **M**
5. Given **h,**it is infeasible to find **x** such that **H(x)=h**
   * + **One-way property**
6. Given **x,** it is infeasible to find **y,**s.t. **H(y)=H(x)**
   * + **Weak collision resistance**
7. Infeasible to find any **x**,**y** such that **H(y)=H(x)**
   * + **Strong collision resistance**
8. What are the properties for a strong digital signature scheme?
   * Create and verify author identity, date & time (through the signature)
   * Authenticate message contents
   * Signature can be verified by a third party to resolve disputes
9. Reviewing slides in Lecture 2B, explain how a hash function is used in digital signature creation and verification.

Hash functions provided both authentication(digital signature) and confidentiality(verification). Confidentiality is achieved by checking the digest to ensure the message has not changed. Authentication is achieved by signing the hash function with a signature.

1. Reviewing the collision-resistance properties of hash functions, explain why it is computationally infeasible to forge with a new message for an existing digital signature.

A new message will provide a new digital signature. A hash function operates one way so it is computationally infeasible to forge a new message for an existing digital signature.

1. Given a piece of message M and a hash function H, briefly describe the processes of signature creation and verification with the ElGamal digital signature scheme.
   1. The hash ***m = H(M)*, *0 ≤ m ≤(q-1)***
   2. Choose random integer **K** with **1 ≤ K ≤(q-1)** and **gcd(K,q-1)=1**
   3. Compute the value: **S1 = ak mod q**
   4. Compute **K-1** the inverse of **K mod (q-1)**
   5. Compute the value: **S2 = K-1(m-xAS1) mod (q-1)**
   6. Signature is: **(S1,S2)**
   7. Any user B can verify the signature by computing
   8. **V1 = am mod q**
   9. **V2 = yAS1 S1S2 mod q**
   10. Signature is valid if **V1 = V2**

**Part 2 – Challenge Exercises**

1. Answer the following questions:
2. Suppose message M is represented by a sequence of bits, i.e., . The hash value of M is defined as , where is the XOR operation. Use the hash value *H(M)* as a checksum. Can this checksum detect ***any*** modification of the message? Why or why not?

The checksum can detect any modification to the message because there is a parity bit that is shifted during the XOR operations.

1. Suppose the message is a sequence of decimal numbers, that is, . The hash value of M is defined as . Do you think this hash function H meets the requirement of strong collision resistance? Why or why not?
2. Consider the ElGamal signature scheme. Suppose (S1,S2)is the digital signature for message M. In order to verify the signature, user B calculates the following values:

V1 = am mod q

V2 = (yAS1 ) ×(S1S2 ) mod q

Explain why the signature is valid when V1 = V2.

**Part 3 – Exercises via CrypTool**

Using the CrypTool, complete the following exercises and answer the questions.

1. Use the RSA keys that you have created in the previous workshop to sign the following document; and then to verify the signature. Is the signature valid? Now, create another pair of keys, and use the new key to verify the same signature. Is the signature valid now?
2. Review the different roles of a private key in digital signature and encryption/decryption, and answer the following question:

In the previous week when you decrypted the ciphertext, CrypTool required that you provide your PIN code (while when the document was encrypted, it did not require the PIN). Now, when you sign the document, the system requires that you enter the PIN (while it does not require the PIN when the signature is verified). Explain the above-mentioned differences, that is, why the PIN is required for decryption and signature generation, while the PIN is not required for encryption and signature verification.

A flurry of events in 1995 demonstrated that issues related to electronic commerce and information security are of deep concern to the public, businesses, government, researchers, and users of the rapidly expanding Internet. The sometimes heated and wide ranging debate concerning cryptographic policy, content controls, commerce, and interoperability on the Internet tends to divert attention away from the need for a reasoned assessment and understanding of the true dynamics of nurturing a diverse global marketplace on the Internet. Lost in the contentious debate is a principle we feel is particularly important given the nature of information technology: that a policy consistent with user requirements and market acceptance provides economic benefits. In some cases, the 'economic pie' can be expanded--or shrunk--by corporate or government actions, inevitably affecting all.

We argue that information security is compromised if federal policies and corporate initiatives ignore user requirements and the basic principles of Internet economics. This in turn may limit market acceptance of new research, services, applications, and technologies. For example, unintended results from ill-formed federal laws may severely limit economic benefits gained from billions of dollars of federally sponsored research that created and sustained the Internet. Furthermore, in the guise of enhanced security and advanced features, proprietary systems and partitioned markets lead to a lack of interoperability that further compromises prospects for society to realize the aforementioned benefits. Many of these problems can be avoided by an open policymaking process that is informed by collaborative research and development activities.

[Source: from a paper on technologies and policies for information security, by Lee McKnight, et al, March 1995]