WEB3CLUBS FOUNDATION LIMITED

Course Instructor: DR. Cyprian Omukhwaya Sakwa
PHONE: +254723584205 Email: cypriansakwa@gmail.com

Foundational Mathematics for Web3 Builders

Lecture 24

June 13, 2024

Hash Functions

- Basic RSA and ElGamal signature systems have a disadvantage of being close to the original message in terms of length.
- Having a signature that is the same length as the document is inefficient and so signature should be minimized.
- One method is to use a hash function, which takes a message as input and outputs a smaller hash value of a set length (e.g., 64 bits) in a deterministic but unexpected manner.
- This means that messages that are nearly identical will typically have different hashes.
- If Cyprian sends a message to Alex, he can sign the hashed value instead of the whole message.

- Alex can validate the message received from Cyprian by computing its hash and verifying its signature. If the message has been altered, the signature will not match the hash of the message Alex received, indicating that something went wrong.
- Hashing also has the use of error detection; Cyprian might send the message's original hash together with the message instead of signing it. Alex will be aware that he did not receive the message correctly if the hash of the message does not match the value Cyprian sent.
- Depending on the hash calculation method, Alex could even be able to identify or correct the errors on his own.
- While it is an interesting topic, we will not be discussing designing error-detecting/error-correcting procedures today. Rather, we are interested in applying hashes to cryptography.

- Hash functions are used in public key cryptography. For example, hash functions are aided by the use of: SSL/TLS certificates (i.e., website security certificates), Code signing certificates, Document signing certificates, and Email signing certificates.

Definition 1 (Hash function)

A hash function H is a function that takes in messages of unbounded length and produces a hash value of fixed length (e.g., 64 bits).

To utilize a hash function H for cryptography, it must meet certain extra requirements:

a) H should be simple to evaluate for any message m. Evaluation should typically take polynomial time based on the number of bits m.

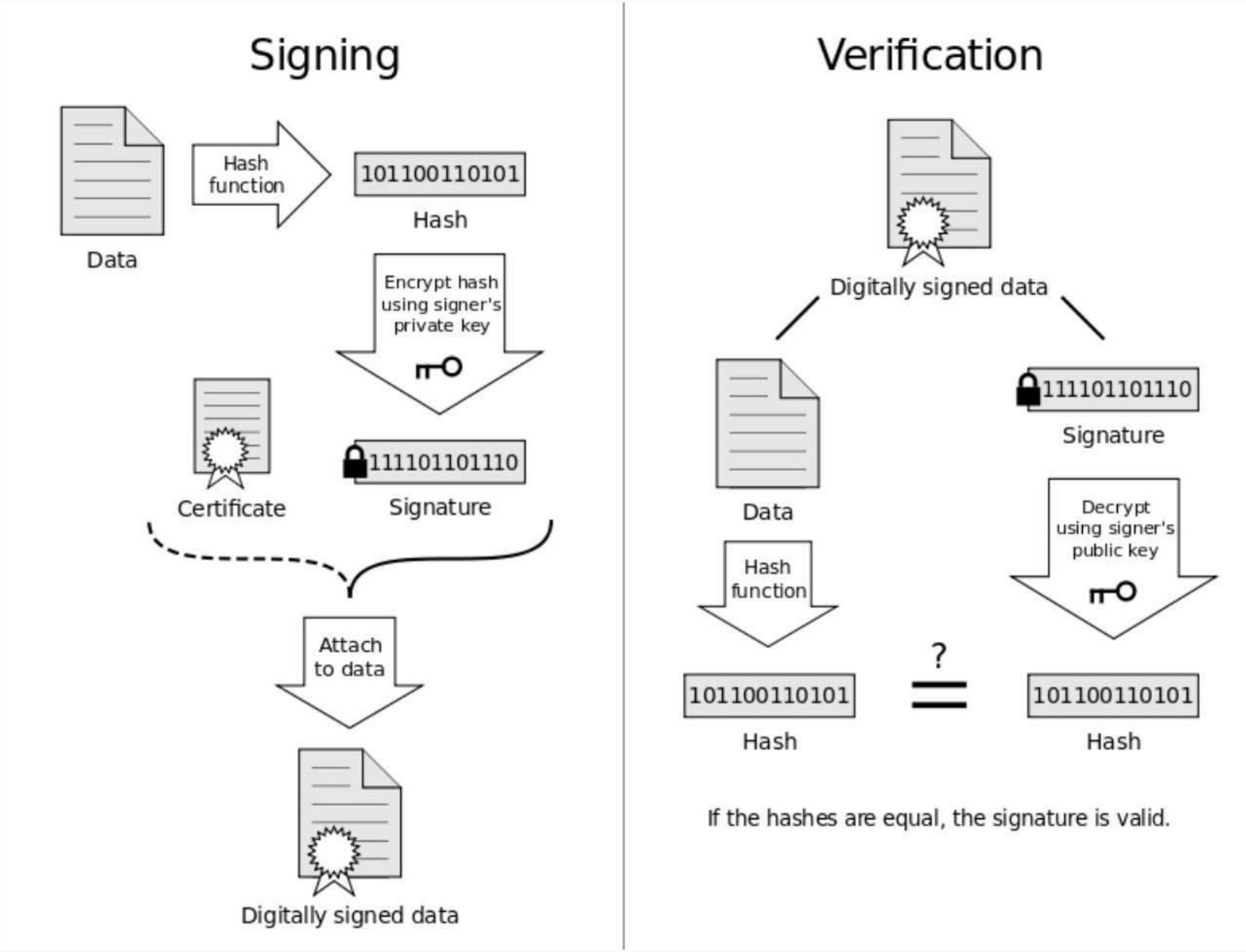
b) H should be difficult to invert. That is, for any y, finding a message x with $H(x) = y$ should be computationally infeasible.

c) H should be collision-resistant, meaning it should be difficult to identify messages m_1 and m_2 where $H(m_1) = H(m_2)$.

1.1 Types of Hash Functions

- One of the widely recognized series of hash functions is the Message-Digest (MD) hash functions which includes MD2, MD4, MD5, and MD6,
- Another widely recognized series is the NSA's Secure Hash Algorithm (SHA) family which includes SHA-0, SHA-1, SHA-2, and SHA-3. The original version was SHA-0, a 160-bit hash algorithm.
- RIPEMD-160
- HAVAL

1.2 The diagram below illustrates the process of digitally signing a message with RSA



Cryptographic hash functions are commonly employed in cryptocurrencies to transmit transaction data anonymously. For example, Bitcoin, the first and largest cryptocurrency, employs the SHA-256 cryptographic hash function in its algorithm.Ethereum, the second most popular blockchain, employs Keccak-256 to hash data.

1.3 Comparison of Secure Hash Functions

	SHA-1	MD5	RIPEMD-160
Digest length	160 bits	128 bits	160 bits
Basic unit of processing	512 bits	512 bits	512 bits
Number of steps	80 (4 rounds of 20)	64 (4 rounds of 16)	160 (5 paired rounds of 16)
Maximum message size	2^{64}-1 bits	∞	∞

RSA Digital Signature Algorithm

The RSA Digital Signature Algorithm is comparable to the RSA algorithm.

Algorithm 1

Assume that you intend to sign and send a message n to your friend. You do the following:

1. Select two prime numbers p and q such that $p \neq q$ and calculates $m = p \times q$.
   Note that m will be made public while p and q will be kept private.
2. Calculate $\phi(m) = (p-1)(q-1)$
3. Choose an integer e such that, $1 < e < \phi(m)$ and e is coprime to $\phi(m)$.
   Note that e will be made public.

Algorithm (conti...)

5. Compute the d such that, $1 < d < \phi(m)$ and that $ed \equiv 1 (\mathrm{mod}\, \phi(m))$. Note that d will be private.
6. Publishe the public key (m, e) and keeps p, q, d and $\phi(m)$ private.
7. Create your signature by raising n to the power d modulo m.

$$\text{Your Signature: } s \equiv n^d \bmod m$$

8. Send n and s to your friend. That is, send the signed ducument (n, s).
9. Your friend verifies that it is really you who signed the message by using your public key (m, e) then raises s to the power e modulo m.

$$s^e \equiv (n^d)^e \equiv n^{ed} \equiv n \bmod m$$

As a result, nobody should be able to forge your signature. If a document has your signature, you cannot successfully deny signing it.

Example 1

For each of the following, show how Ken would sign a message m using RSA digital signature algorithm with modulus $n = p \times q$ to obtain a signed document (m, s) if he chooses a public exponent e. Then show how Naomi would verify that the message was signed by Ken.

a) $p = 5,\ q = 13,\ e = 7,\ m = 6$

b) $p = 7,\ q = 19,\ e = 5,\ m = 7$

Solution

a) Here, $n = 5 \times 13 = 65$ and $\phi(n) = 4 \times 12 = 48$.

Now find d such that $7d \equiv 1 \mod 48$. By Euclid's algorithm we have

$$48 = 7(6) + 6$$

$$7 = 6(1) + 1$$

$$6 = 1(6) + 0$$

Now use extended Euclidean algorithm to get

$$1 = 7 - 6(1) = 7 - [48 - 7(6)](1)$$

$$= 7 - 48(1) + 7(6) = 7(7) + 48(-1)$$

Thus $d = 7$. The public key is $(65, 7)$ and private key is $(65, 7)$

To sign the private key is used.

Signature: $s = m^d = 6^7 \mod 65 = 46 \mod 65$.

Signed document is $(m, s) = (6, 46)$.

To verify we use the public key is used.

Solution (conti...)

$46^7 \bmod 65$. Let us use the fast powering algorithm.

$$\begin{aligned} 46^1 &= 46 \\ 46^2 &= 36 \\ 46^4 &= 61 \\ \therefore 46^7 &= 46^4 \times 46^2 \times 46^1 \\ &= 61 \times 36 \times 46 = 6 \end{aligned}$$

Hence verified.

b) Here, $n = 7 \times 19 = 133$ and $\phi(n) = 6 \times 18 = 108$.

By Euclid's algorithm we get:

Now find d such that $5d \equiv 1 \bmod 108$

$$\begin{aligned} 108 &= 5(21) + 3 \\ 5 &= 3(1) + 2 \\ 3 &= 2(1) + 1 \end{aligned}$$

Solution (conti...)

Now use extended Euclidean algorithm

$$\begin{aligned} 1 = 3 - 2(1) &= 3 - [5 - 3(1)](1) \\ &= 3 - 5(1) + 3(1) = 5(-1) + 3(2) \\ &= 5(-1) + [108 - 5(21)](2) = 108(2) + 5(-43) \end{aligned}$$

Thus, $d = -43 \bmod 108 \equiv 65$

Signature: $s = 7^{65} \bmod 133$.

Use fast powering algorithm

$$7^1 = 7 \qquad 7^2 = 49$$

$$7^4 = 7 \qquad 7^8 = 49$$

$$7^{16} = 7 \qquad 7^{32} = 49$$

$$7^{64} = 7 \qquad \therefore 7^{65} = 7 \times 7 = 49$$

Solution (conti...)

Thus signed document is $(7, 49)$

To verify we compute $49^5 \bmod 133 = 7$

Example 2

Lynette uses the RSA signature with public modulus 55 and public encryption exponent 7. She then sends three documents to Alex with her signature attached. Alex creates a fourth document and unsuccessfully tries to forge Alice's signature. Which of the four documents is the forgery?

$(3, 27)$, $(12, 23)$, $(9, 12)$, $(19, 39)$

Solution

Solution

Solution

Solution

Example 3

Cyprian chooses primes $p = 113$ and $q = 167$ and public exponent $e = 71$. He wants to sign the document $m = 11$. What is the signed document? Show the calculations that verify that the above signed document is valid.

Solution

$n = 113 \times 167 = 18871$ and $\phi(n) = 112 \times 166 = 18592$

Now find d if $71d \equiv 1 \bmod 18592$

Solution (conti...)

By Euclid's algorithm we have

$$18592 = 71(261) + 61$$

$$71 = 61(1) + 10$$

$$61 = 10(6) + 1$$

By extended Euclidean algorithm we have

$$1 = 61 - 10(6) = 61 - [71 - 61(1)](6)$$

$$= -71(6) + 61(7) = -71(6) + [18592 - 71(261)](7)$$

$$= 18592(7) + 71(-1833)$$

Thus $d = -1833 \bmod 18592 = 16759$

The signature $s = 11^{16759} \bmod 18871 = 2321$ by fast powering algorithm

The signed document is $(11, 2321)$

To verify we compute $2321^{71} \bmod 18871 = 11$ by fast powering algorithm

Exercise (conti...)

3. Alex chooses 7 and 11 as p and q and calculates $n = 7 \times 11 = 77$. The value of $\phi(n) = (7-1)(11-1) = 60$. If he chooses $e = 23$ to be his public key, calculate d, his private key such that $ed \equiv 1 \bmod \phi(n)$. Now show how you would sign the document $m = 15$ to Alex and show how it can be verified as valid.
4. Now assume that Ochieng' wants to send a message to Alex. Ochieng' can use the same public key announced by Alex (probably on his website), 23; Ochieng"s plaintext is 67. Show how Ochieng' would encrypt 67 and how Alex would decrypt the ciphertext.

Exercise (Conti...)

5. You have been sent the following message. You may use your computer.

   $79880,\ 113612,\ 97518,\ 82767,\ 80745,\ 102524,\ 1076,\ 102745,\ 91940$

   It has been encrypted using $p = 313, q = 373,$
   $m = pq = 116749$, and $e = 161$. Decrypt the message.

6. Encrypt "YOU SETTLE THE CASE" using $p = 5, q = 17,$ $e = 3$. How will your friend decrypt the ciphertext?

7. Write a program to implement the RSA cryptosystem. Make your program as user friendly as possible. In particular, the person encoding a message should be able to type in their message as words, including spaces and punctuation; similarly, the decoder should see the message appear as words with spaces and punctuation decrypt this?

Exercise (Conti...)

8 You have a public RSA modulus $n = 119 = 7 \times 17$ and public exponent $e = 5$. You want to sign the document $m = 7$.

a) What is the signed document?

b) Show that how calculation verifies that the document you produced in part (a) is valid.