(20 points) Determine gcd(12075, 4655)

|  |  |  |
| --- | --- | --- |
| Rewrite as Extended Euclidean | | |
|  |  | Extended Euclidean |
|  |  | substitution |
|  |  |  |
| Use Division Algorithm to Solve Euclidean equations | | |
|  |  | *solve , to find* |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  | *Repeat until , using as and as* |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  | *Repeat until , using as and as* |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  | *Repeat until , using as and as* |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  | *Repeat until , using as and as* |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  | *Repeat until , using as and as* |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  | *When ,* |

(20 points) Lab5 Use OpenSSL to Generate Random Number and Test Primality

1. (10 points) Based on the script primeTest, write another shell script using OpenSSL command line. The script is called isprime. It requires an integer as an input parameter. If the input integer is a prime, the script will show “\*\*\* is a prime”. If not, the script will show “\*\*\* is not a prime”.

|  |
| --- |
| #!/bin/bash  input**=$1**  # Use OpenSSL to check if prime  result**=$(openssl prime $input)**  # Verify OpenSSL Ouput and display message  **if** **[[** "$result" **==** **\***is\ prime **]];** **then**  **echo** "$input is a prime number"    **elif** **[[** "$result" **==** **\***not\ prime **]];** **then**  **echo** "$input is not a prime number"  **fi** |

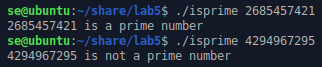
* 1. A screenshot of a computer program

     Description automatically generated

1. (5 points) Use the program to decide if the following numbers are prime numbers:

a. 2685457421

b. 4294967295

* 1. 

1. (5 points ) If an integer passes the Miller-Rabin primality test, does that guarantee the number is a prime number? Why?
   1. No, an integer passing the Miller-Rabin primality test does not guarantee that it is prime. Since the Miller-Rabin primality test is a probabilistic primality test, it can only indicate that the integer is a probable prime. In other words, since the test relies on randomness in its computations, there are scenarios where a composite number might pass the test as if it were prime.

Lab6 Use OpenSSL to Create RSA Public/Private Key (512bits) without Password Protection

1. (8 points) In Step 1, what is the public key (e, n) used? Why?
   1. The exponent, 65537 (0x10001), is shown during RSA private key generation in Step 1. It is a popular choice for RSA key generation as it is a prime number, which ensures that it is relatively prime to and , where and are the prime factors of the modulus . Also, it is small enough to not impede cryptographic operations while still being large enough that, when combined with large modulus, (i.e 2048-bit or 4096-bit), it provides enough complexity that it is practically impossible to break the encryption by factoring the modulus and obtaining the private key.
   2. Alternatively, is not shown during RSA private key generation in Step 1 and is randomized during each generation since it is derived from the product of two randomly selected large prime numbers ( and ). This is to ensure that each key pair is unique and not predictable. However, this value can be shown using the openssl command:

openssl rsa -in private512.key -modulus -noout

on the private key. Or using the command:

openssl rsa -pubin -in public512.key -text -noout

on the public key.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

1. (12 points) A certificate, Amazon.cer, can be found in the lab 6 folder, answer the following questions:
   1. What is the signature hash algorithm used to create the certificate?
      1. 1.2.840.113549.1.1.11 (sha256WithRSAEncryption)
      2. A screenshot of a computer

         Description automatically generated
      3. A screen shot of a computer

         Description automatically generated
   2. What is the public key cryptography used to create the signature?
      1. RSA
      2. A screenshot of a computer

         Description automatically generated
      3. A screenshot of a computer

         Description automatically generated
   3. What is the size of the public key?
      1. 2048
      2. A screenshot of a computer

         Description automatically generated
      3. A screenshot of a computer

         Description automatically generated
   4. What is the *e* used in the public key in the certificate?
      1. 65537 (0x10001)
      2. A screenshot of a computer

         Description automatically generated
   5. Who issued the certificate?
      1. Starfield Services Root Certificate Authority - G2 of Starfield Technologies, Inc.
      2. A screenshot of a computer

         Description automatically generated
      3. A computer screen shot of white text

         Description automatically generated
   6. Why can you trust the certificate?
      1. This certificate is trustworthy because it has been issued by a trusted root certificate authority (Starfield Services Root Certificate Authority - G2), it has been signed using RSA encryption, and it is still within its validity timeframe (May 25, 2015 - December 31, 2037)

Lab7 Use OpenSSL to Create RSA Public/Private Key (4096bits) with Password Protection

1. (8 points) Use symmetric cipher and message authentication code to provide confidentiality, integrity, and authentication. (use the giving cryptographic operations only).

A diagram of a rectangular object with black text

Description automatically generated

1. (8 points) Use symmetric cipher and digital signature to provide confidentiality, integrity, and authentication. (use the giving cryptographic operations only).
2. A diagram of a data flow

   Description automatically generated
3. (4 points) Compare the two scenarios as designed in a) and b). What are their advantages and limitations?
4. Scenario A, the symmetric cipher and MAC, offers efficient data encryption, ensuring confidentiality while maintaining data integrity and authentication. The addition of a MAC provides an extra layer of security on top of the cipher as it verifies data integrity and source authentication. However, securely managing and distributing the shared secret keys for the symmetric cipher can be challenging. Scenario B, the symmetric cipher with a digital signature, improves security by providing a means to authenticate the sender's identity and prevent data tampering. However, these are susceptible to various attacks, including replay attacks, man-in-the-middle attacks, and key compromise attacks

Lab8 A Combination of RSA and AES to Encrypt a File

1. (6 points) What is the throughput when you sign and verify a message using a 1024-bit key in RSA? Which one is faster? Why?
   1. Sign throughput: Approximately 0.000063 seconds per operation (RSA 1024 bits)
   2. Verify throughput: Approximately 0.000004 seconds per operation (RSA 1024 bits)
   3. The verify throughput is faster than the sign throughput because verifying a signature involves simpler computations with a small public exponent (e.g., 3, 5, 17, 257, or 65537), thus making it quick and inexpensive, whereas generating signatures requires more computationally intensive operations with a private exponent (d).
   4. A computer screen shot of a computer screen

      Description automatically generated
2. (6 points) What is the throughput when you encrypt a 1024-bit message block using DES CBC mode? Compared with a 1024-bit RSA verify throughput, which one is faster? Compared with a 1024-bit RSA sign throughput, which one is faster?
   1. 113255.42k bytes per second (1024 bytes)
   2. RSA 1024-bit verify throughput: 253774.1 operations per second
   3. RSA 1024-bit sign throughput: 15905.2 operations per second
   4. Comparatively, DES CBC encryption is faster than RSA verification but slower than RSA signing for 1024-bit operations.
   5. A screenshot of a computer program

      Description automatically generated
3. (8 points) What is the plaintext you restored from the cipher text?
   1. After decrypting the cipher text with the password obtained from RSA decryption (hello world), the restored plaintext is:
   2. Belief creates the actual fact. William James
   3. A computer screen shot of a computer code

      Description automatically generated