Compute The Two Public Keys And The Common Key For The Dhke Scheme With The Parameters P = 467, = 2,

Compute The Two Public Keys And The Common Key For The Dhke Scheme With The Parameters P = 467, = 2,

Introduction

The Diffie-Hellman Key Exchange (DHKE) is a fundamental cryptographic protocol that enables two parties to securely establish a shared secret over an insecure communication channel. This shared secret can then be used to encrypt subsequent communications, ensuring confidentiality. The security of the DHKE relies on the difficulty of solving the discrete logarithm problem in a finite cyclic group, typically defined using a large prime modulus and a primitive root.

In this article, we will take a detailed, step-by-step approach to compute the two public keys and the common secret key for the DHKE scheme, given specific parameters: a prime number $P = 467$ and a primitive root (also called a generator) $g = 2$. We will assume two users, traditionally named Alice and Bob, each selecting their private keys, and then proceed to calculate their corresponding public keys and the shared secret.

Understanding the Parameters and Notation

Before diving into calculations, it’s important to clarify the variables involved:

- Prime modulus P: The prime number defining the finite cyclic group, here $P = 467$.
- Primitive root g: A generator of the multiplicative group modulo P, here $g = 2$.
- Private keys: Secret integers chosen independently by Alice and Bob, denoted as a and b.
- Public keys: Computed values shared publicly, denoted as A and B.
- Shared secret key: The secret value both Alice and Bob can derive, denoted as K.

Step 1: Selecting Private Keys

The first step in the Diffie-Hellman scheme involves each participant selecting a private key:

Choosing Private Keys

- Alice selects a private key a, a randomly chosen integer $1 < a <$

P-1 \).
- Bob selects a private key \(b \), similarly a random integer \(1 < b < P-1 \).

For simplicity, and to illustrate the process explicitly, let’s choose small private keys:

- Alice’s private key: \(a = 5 \)
- Bob’s private key: \(b = 7 \)

These values are kept secret and are only known to Alice and Bob, respectively.

Step 2: Computing Public Keys

Each participant computes their public key using the primitive root \(g \), their private key, and the prime \(P \):

$$
A = g^a \mod P
$$
$$
B = g^b \mod P
$$

Calculating Alice's Public Key \(A \)

$$
A = 2^a \mod 467 = 2^5 \mod 467
$$
$$
2^5 = 32
$$
$$
A = 32 \mod 467 = 32
$$

So, Alice’s public key is:

$$
A = 32
$$

Calculating Bob's Public Key \(B \)

$$
B = 2^b \mod 467 = 2^7 \mod 467
$$
$$
2^7 = 128
$$

$$
B = 128 \mod 467 = 128
$$

Thus, Bob’s public key is:

$$
B = 128
$$

Summary of Public Keys

- Alice’s public key: $A = 32$
- Bob’s public key: $B = 128$

These public keys are exchanged over an insecure channel.

Step 3: Computing the Common Secret Key

After exchanging public keys, each party computes the shared secret key K using their private key and the other party’s public key.

Alice’s Computation of K

$$
K = B^a \mod P = 128^5 \mod 467
$$

Calculating $128^5 \mod 467$:

- First, compute $128^2 \mod 467$:

$$
128^2 = 16,384
$$
$$
16,384 \mod 467
$$

To compute $16,384 \mod 467$:

Divide 16,384 by 467:

$$
467 \times 35 = 16,345
$$
$$
16,384 - 16,345 = 39
$$

Thus:

$$
128^2 \equiv 39 \mod 467
$$

- Next, compute $128^4 = (128^2)^2$:

$$
39^2 = 1,521
$$

Calculate $1,521 \mod 467$:

$$
467 \times 3 = 1,401
$$
$$
1,521 - 1,401 = 120
$$

So:

$$
128^4 \equiv 120 \mod 467
$$

- Now, compute $128^5 = 128^4 \times 128$:

$$
128^5 \equiv 120 \times 128 \mod 467
$$

Calculate 120×128:

$$
120 \times 128 = (120 \times 100) + (120 \times 28) = 12,000 + 3,360 = 15,360
$$

Now, compute $15,360 \mod 467$:

Divide 15,360 by 467:

$$
467 \times 32 = 14,944
$$
$$
15,360 - 14,944 = 416
$$

Thus:

$$
\[

$$
K = 416
$$

Bob’s Computation of K

Similarly, Bob computes:

$$
K = A^b \mod 467 = 32^7 \mod 467
$$

We will compute $32^7 \mod 467$ efficiently using successive squaring:

- Compute 32^2:

$$
32^2 = 1024
$$
$$
1024 \mod 467
$$

Divide 1024 by 467:

$$
467 \times 2 = 934
$$
$$
1024 - 934 = 90
$$

So,

$$
32^2 \equiv 90 \mod 467
$$

- Compute $32^4 = (32^2)^2$:

$$
90^2 = 8,100
$$

Calculate $8,100 \mod 467$:

$$
467 \times 17 = 7,939
$$
$$
8,100 - 7,939 = 161
$$

Thus,

$$
32^4 \equiv 161 \mod 467
$$

- Compute $32^7 = 32^{4} \times 32^{2} \times 32^{1}$:

$$
32^7 \equiv 161 \times 90 \times 32 \mod 467
$$

First, compute 161×90:

$$
161 \times 90 = 14,490
$$

Then, compute $14,490 \times 32$:

$$
14,490 \times 32 = (14,490 \times 30) + (14,490 \times 2) = 434,700 + 28,980 = 463,680
$$

Now, compute $463,680 \mod 467$:

Divide 463,680 by 467:

Calculate:

- 467×990:

$$
467 \times 990 = 462,330
$$

Difference:

$$
463,680 - 462,330 = 1,350
$$

Next, $467 \times 2 = 934$:

$$
1,350 - 934 = 416
$$

Since 416 < 467, the remainder is 416.

Therefore,

$$
K = 416
$$

Confirming the Shared Secret

Both calculations yield:

$$
K = 416
$$

This shared secret key can now be used for encrypting subsequent communications.

Summary of Results

- Alice’s public key: $A = 32$
- Bob’s public key: $B = 128$
- Shared secret key: $K = 416$

Additional Considerations

Security Aspects

- The private keys a and b should be randomly chosen and kept secret.
- The prime $P = 467$ and generator $g = 2$ are relatively small for real-world cryptography; in practice, larger primes are used.
- The security relies on the difficulty of the discrete logarithm problem in the finite group modulo P.

Practical Implementations

- Implementation of DHKE involves selecting large primes (2048 bits or more).
- Additional steps like authentication are necessary to prevent man-in-the-middle attacks.

Conclusion

The Diffie-Hellman Key Exchange provides a simple yet powerful method for two parties to agree on a shared secret over an insecure channel. By choosing private keys, computing public keys via exponentiation modulo a prime, and then deriving the secret key through exponentiation of the other’s public key, both parties arrive at the same secret without transmitting it directly.

In this detailed example with parameters $(

Frequently Asked Questions

What are the steps to compute the public keys in the Diffie-Hellman key exchange when P = 467 and g = 2?

To compute the public keys, each party chooses a private key (say, a and b), then computes their public keys as A = g^a mod P and B = g^b mod P. The specific values depend on the private keys chosen by each party.

How do you compute the common key once the public keys are known in the Diffie-Hellman scheme with P = 467 and g = 2?

The common key is computed by each party raising the other’s public key to their own private key modulo P. For example, Alice computes K = B^a mod P, and Bob computes K = A^b mod P. Both should arrive at the same value.

Given P = 467 and g = 2, what are example private keys, and how are the corresponding public keys calculated?

Private keys are typically chosen randomly; for illustration, if Alice's private key is a=5, then her public key A = 2^5 mod 467 = 32. If Bob's private key is b=7, then his public key B = 2^7 mod 467 = 128.

How is the shared secret key calculated in the Diffie-Hellman scheme with the provided parameters?

After exchanging public keys, each party computes the shared secret as the other’s public key raised to their own private key modulo P. For example, if Alice's private key is a and Bob's public key is B, Alice computes K = B^a mod 467.

Can you demonstrate the calculation of public and shared keys with specific private keys for P=467 and g=2?

Certainly. Suppose Alice's private key is a=3, then her public key A = 2^3 mod 467 = 8. Suppose Bob's private key is b=4, then his public key B = 2^4 mod 467 = 16. The shared key is then computed as B^a mod 467 = 16^3 mod 467 = 4096 mod 467 = 4096 - 8467 = 4096 - 3736 = 360.

What are common pitfalls or errors to avoid when

computing Diffie-Hellman keys with these parameters?

Common errors include using incorrect private keys, miscalculating modular exponentiation, or forgetting that both parties need to perform the exponentiation with their private keys to arrive at the same shared secret. Ensuring accurate calculations and proper modular arithmetic is essential.

Additional Resources

Compute The Two Public Keys And The Common Key For The DHKE Scheme With The Parameters P = 467, g = 2

Introduction

Diffie-Hellman Key Exchange (DHKE) is a pioneering cryptographic protocol that enables two parties to establish a shared secret over an insecure communication channel. Its significance lies in its ability to facilitate secure symmetric key establishment without prior shared secrets, laying the groundwork for modern secure communications such as SSL/TLS protocols.

In this detailed investigation, we analyze a specific instance of the DHKE scheme with the parameters $P = 467$ and $g = 2$. Our goal is to compute the two public keys generated by the communicating parties — traditionally Alice and Bob — and to derive their shared secret key. This exercise not only demonstrates the practical process of DHKE but also provides insight into the underlying mathematics, modular arithmetic, and cryptographic principles that make the scheme robust.

Background on Diffie-Hellman Key Exchange

The Mathematical Foundation

The DHKE relies on properties of modular exponentiation within a finite cyclic group. Given a large prime P and a primitive root g modulo P, each party selects a private key and computes a corresponding public key:

- Private key: A randomly chosen secret integer a (for Alice) and b (for Bob).
- Public key: Computed as $A = g^a \mod P$ for Alice, and $B = g^b \mod P$ for Bob.

The shared secret key is then derived as:

$$
\text{Shared Key} = g^{ab} \mod P
$$

\]

which can also be computed independently by each party as:

- Alice computes \(K = B^a \mod P \),
- Bob computes \(K = A^b \mod P \).

The security stems from the difficulty of deriving \(a \) or \(b \) given \(g \), \(P \), and the public keys \(A \) and \(B \).

Parameter Specification

In our scenario, the parameters are:

- Prime modulus, \(P = 467 \)
- Generator, \(g = 2 \)

We will assume specific private keys for Alice and Bob to demonstrate the process. Without loss of generality, let's select:

- Alice's private key, \(a = 123 \)
- Bob's private key, \(b = 45 \)

These values are chosen randomly but within the range \(1 \leq a, b < P \).

Step 1: Computing Public Keys

Alice's Public Key \(A \)

Alice computes her public key as:

\[
A = g^a \mod P = 2^{123} \mod 467
\]

Calculating \(2^{123} \mod 467 \) directly is computationally intensive; instead, we employ fast modular exponentiation (also known as binary exponentiation or square-and-multiply algorithm) for efficiency.

Fast Modular Exponentiation Process:

- Express exponent \(123 \) in binary: \(1111011_2 \)
- Iteratively square and multiply, reducing modulo 467 at each step.

Step-by-step calculation:

1. Initialize \(result = 1 \), \(base = 2 \), \(exponent = 123 \), \(

modulus = 467 \).

2. Loop through bits of the exponent:

- For each bit:

- If the bit is 1, multiply result by base modulo 467.
- Square the base modulo 467.

3. Calculations:

Step	Exponent bit	Action	Result	Base	Explanation
1	1	result = (1 2) % 467 = 2	2	Initial
2	1	result = (2 2) % 467 = 4	4	Square base
3	1	result = (4 4) % 467 = 16	16	Square base
4	1	result = (16 16) % 467 = 256	256	Square base
5	0	skip multiplication	256	Square base
6	1	result = (256 256) % 467	256	Square base
7	1	result = (256 256) % 467	256	Square base

But to avoid ambiguity, let's perform the calculation stepwise:

- \(2^{1} \equiv 2 \)
- \(2^{2} \equiv 4 \)
- \(2^{4} \equiv 16 \)
- \(2^{8} \equiv 16^{2} \equiv 256 \)
- \(2^{16} \equiv (2^{8})^{2} \equiv 256^{2} \equiv ? \)

Calculate \(256^{2} \mod 467 \):

\[
256^{2} = 65536
\]

Divide 65536 by 467:

\[
467 \times 140 = 65380
\]
\[
65536 - 65380 = 156
\]

So,

\[
256^{2} \equiv 156 \pmod{467}
\]

Similarly, \(2^{32} = (2^{16})^{2} \equiv 156^{2} \):

$$
156^{2} = 24336
$$

Divide 24336 by 467:

$$
467 \times 52 = 24284
$$
$$
24336 - 24284 = 52
$$

Thus,

$$
2^{32} \equiv 52 \pmod{467}
$$

Next, $2^{64} = (2^{32})^{2} \equiv 52^{2}$:

$$
52^{2} = 2704
$$

Divide 2704 by 467:

$$
467 \times 5 = 2335
$$
$$
2704 - 2335 = 369
$$

Thus,

$$
2^{64} \equiv 369 \pmod{467}
$$

Finally, $2^{123} = 2^{64} \times 2^{32} \times 2^{16} \times 2^{8} \times 2^{3}$.

Express 123 in binary: 1111011_2, corresponding to:

$$
2^{64} + 2^{32} + 2^{16} + 2^{8} + 2^{2} + 2^{1} + 2^{0}
$$

From previous calculations:

- $2^{64} \equiv 369$

- $2^{32} \equiv 52$

- $2^{16} \equiv 156$

- $2^{8} \equiv 256$

- $2^{2} = 4$

- $2^{1} = 2$

- $2^{0} = 1$

Now, compute A:

$$
A \equiv (369 \times 52 \times 156 \times 256 \times 4 \times 2 \times 1) \mod 467
$$

Perform stepwise:

1. 369×52:

$$
369 \times 52 = (369 \times 50) + (369 \times 2) = 18450 + 738 = 19288
$$

Modulo 467:

$$
467 \times 41 = 19147
$$
$$
19288 - 19147 = 141
$$

So,

$$
(369 \times 52) \equiv 141 \pmod{467}
$$

2. Multiply by 156:

$$
141 \times 156 = (141 \times 100) + (141 \times 50) + (141 \times 6) = 14,100 + 7,050 + 846 = 22, -66
$$

Sum:

$$
14,100 + 7,050 + 846 = 22, -66
$$

Actually, better to do directly:

$$
141 \times 156
$$

Calculate:

$$
141 \times 156 = (141 \times 150) + (141 \times 6) = 21,150 + 846 = 22, -66
$$

Wait, double-check:

- $141 \times 150 = 21,150$
- $141 \times 6 = 846$

Sum: $21,150 + 846 = 22, -66$

But negative indicates a mistake; perhaps an error in the calculation. Let's do exact:

$$
141 \times 156 = ?

$$

Break down:

$$
141 \times 156 = 141 \times (100 + 50 + 6) = 141 \times 100 + 141 \times 50 + 141 \times 6
$$

Calculate:

Compute The Two Public Keys And The Common Key For The Dhke Scheme With The Parameters P 467 2

Compute The Two Public Keys And The Common Key For The Dhke Scheme With The Parameters P 467 2

Related Articles

- At A Diving Competition, Allison Scored 78.5 On Her First Dive. Hannah's Score On Her First Dive Was
- Charles Dickens, The Great 19th Century British Author, Did Not Think Too Favorably Of Accountants -
- At A Sale This Week, A Desk Is Being Sold For $285.60. This Is 34% Of The Original Price.What Is The

Back to Home