Example 144. Find x such that $4 \equiv 3^x \pmod 7$.

Solution. We have seen in Example 131 that 3 is a primitive root modulo 7. Hence, there must be such an x.
Going through the possibilities ($3^2 \equiv 2$, $3^3 \equiv 6$, $3^4 \equiv 4$), we find $x = 4$, because $3^4 \equiv 4 \pmod 7$.

Example 145. Find x such that $3 \equiv 2^x \pmod{101}$.

Solution. Let us check that the solution is $x = 69$. Indeed, a quick binary exponentiation confirms that $2^{69} \equiv 3 \pmod{101}$. (Do it!)
The point is that it is actually (believed to be) very difficult to compute these **discrete logarithms**. On the other hand, just like with factorization, it is super easy to verify the answer if somebody tells us the answer.
Comment. We can check that 2 is a primitive root modulo 101. That is, $2 \pmod{101}$ has (multiplicative) order 100.

Diffie–Hellman key exchange

The key idea that makes ElGamal encryption work is that Alice (her private secret is y) and Bob (his private secret is x) actually share a secret: g^{xy}

Since g^x is publicly known, Alice can compute $g^{xy} = (g^x)^y$ using her secret y.
Similarly, since g^y is known from the ciphertext, Bob can compute $g^{xy} = (g^y)^x$ using his secret x.

(Diffie–Hellman key exchange)

- Alice or Bob choose a prime p and a primitive root $g \pmod p$.
- Bob randomly selects a secret integer x and reveals $g^x \pmod p$ to everyone.
  Alice randomly selects a secret integer y and reveals $g^y \pmod p$ to everyone.
- As above, Alice and Bob now share the secret $g^{xy} \pmod p$.

Why is this secure? We need to see why eavesdropping Eve cannot (simply) obtain the secret $g^{xy} \pmod p$.
She knows $g, g^x, g^y \pmod p$ and needs to find $g^{xy} \pmod p$.
This is precisely the CDH problem, which is believed to be hard.

Example 146. You are Eve. Alice and Bob select $p = 53$ and $g = 5$ for a Diffie–Hellman key exchange. Alice sends 43 to Bob, and Bob sends 20 to Alice. What is their shared secret?

Solution. Let's crack Alice's secret y (you can also attack Bob).
For that, we need to find y such that $5^y = 43 \pmod{53}$. That means we have to compute a discrete logarithm.
Since we haven't learned a better method, we just try $y = 2, 3, \ldots$ until we find the right one: $5^2 = 25$, $5^3 \equiv 19$, $5^4 \equiv 19 \cdot 5 \equiv -11$, $5^5 \equiv -11 \cdot 5 \equiv -2$, $5^6 \equiv -2 \cdot 5 \equiv -10 \equiv 43 \pmod{53}$.
Hence, Alice's secret is $y = 6$. The shared secret is $20^6 \equiv 9 \pmod{53}$.

Armin Straub
straub@southalabama.edu

(ElGamal encryption)

- Bob chooses a prime p and a primitive root $g \pmod p$.
  Bob also randomly selects a secret integer x and computes $h = g^x \pmod p$.
- Bob makes (p, g, h) public. His (secret) private key is x.
- To encrypt, Alice first randomly selects an integer y.
  Then, $c = (c_1, c_2)$ with $c_1 = g^y \pmod p$ and $c_2 = h^y m \pmod p$.
- Bob decrypts $m = c_2 c_1^{-x} \pmod p$.

Why does that work? $c_2 c_1^{-x} = (h^y m)(g^y)^{-x} = ((g^x)^y m)(g^y)^{-x} = m \pmod p$

Comment. For ElGamal, the message space actually is $\{1, 2, ..., p-1\}$. $m = 0$ is not permitted.
That's, of course, no practical issue. For instance, we could simply identify $\{1, 2, ..., p-1\}$ with $\{0, 1, ..., p-2\}$ by adding/subtracting 1.

Comment. p and g don't have to be chosen randomly. They can be reused. In fact, it is common to choose p to be a "safe prime" (see next comment), with specific pre-selected choices listed, for instance, in RFC 3526.

Advanced comment. Note that in order to check whether g is a primitive root modulo p, we need to be able to factor $p-1$, which in general is hard (2 is an obvious factor, but other factors are typically large and, in fact, we need them to be large in order for the discrete logarithm problem to be difficult). It is therefore common to start with a prime n and then see if $2n+1$ is prime as well, in which case we select $p = 2n+1$. Such primes p [primes such that $(p-1)/2$ is prime, too] are called **safe primes**.
On the other hand, g doesn't necessarily have to be a primitive root. However, we need the group generated by g (the elements $1, g, g^2, g^3, ...$) to be large. For more fancy cryptosystems, we can even replace these groups with other groups such as those generated by elliptic curves.

- Like RSA, ElGamal is terribly slow compared with symmetric ciphers like AES.

  Encryption under ElGamal requires two exponentiations (slower than RSA); however, these exponentiations are independent of the message and can be computed ahead of time if need be (in that case, encryption is just a multiplication, which is much faster than RSA). Decryption only requires one exponentiation (like RSA).

- In contrast to RSA, ElGamal is randomized. That is, a single plaintext m can be encrypted to many different ciphertexts.

  A drawback is that the ciphertext is twice as large as the plaintext.
  On the positive side, an attacker who might be able to guess potential plaintexts cannot (as in the case of vanilla RSA) encrypt these herself and compare with the intercepted ciphertext.

Example 147. Bob chooses the prime $p=31$, $g=11$, and $x=5$. What is his public key?

Solution. Since $h=g^x \pmod p$ is $h\equiv 11^5\equiv 6 \pmod{31}$, the public key is $(p,g,h)=(31,11,6)$.
Comment. Bob's secret key is $x=5$. In principle, an attacker can compute x from $11^x\equiv 6 \pmod{31}$. However, this requires computing a discrete logarithm, which is believed to be difficult if p is large.

Example 148. Bob's public ElGamal key is $(p,g,h)=(31,11,6)$.

(a) Encrypt the message $m=3$ ("randomly" choose $y=4$) and send it to Bob.

(b) Recall that Bob's secret private key is $x=5$. Use it to decrypt $c=(9,13)$.

Solution.

(a) The ciphertext is $c=(c_1,c_2)$ with $c_1=g^y \pmod p$ and $c_2=h^y m \pmod p$.
Here, $c_1=11^4\equiv 9 \pmod{31}$ and $c_2=6^4\cdot 3\equiv 13 \pmod{31}$. Hence, the ciphertext is $c=(9,13)$.

(b) We decrypt $m=c_2c_1^{-x} \pmod p$.
Here, $m=13\cdot 9^{-5}\equiv 3 \pmod{31}$.
Comment. One option is to compute $9^{-1}\equiv 7 \pmod{31}$, followed by $9^{-5}\equiv 7^5\equiv 5 \pmod{31}$ and, finally, $13\cdot 9^{-5}\equiv 13\cdot 5\equiv 3 \pmod{31}$. Another option is to begin with $9^{-5}\equiv 9^{25} \pmod{31}$ (by Fermat's little theorem).

Example 149. Bob's public ElGamal key is $(p,g,h)=(23,10,11)$.

(a) Encrypt the message $m=5$ ("randomly" choose $y=2$) and send it to Bob.

(b) Encrypt the message $m=5$ ("randomly" choose $y=4$) and send it to Bob.

(c) Break the cryptosystem and determine Bob's secret key.

(d) Use the secret key to decrypt $c=(8,7)$.

(e) Likewise, decrypt $c=(18,19)$.

Solution.

(a) The ciphertext is $c=(c_1,c_2)$ with $c_1=g^y \pmod p$ and $c_2=h^y m \pmod p$.
Here, $c_1=10^2\equiv 8 \pmod{23}$ and $c_2=11^2\cdot 5\equiv 6\cdot 5\equiv 7 \pmod{23}$. Hence, the ciphertext is $c=(8,7)$.

(b) Now, $c_1=10^4\equiv 18 \pmod{23}$ and $c_2=11^4\cdot 5\equiv 13\cdot 5\equiv 19 \pmod{23}$ so that $c=(18,19)$.

(c) We need to solve $10^x\equiv 11 \pmod{23}$. This yields $x=3$.
(Since we haven't learned a better method, we just try $x=1,2,3,\ldots$ until we find the right one.)

(d) We decrypt $m=c_2c_1^{-x} \pmod p$.
Here, $m=7\cdot 8^{-3}\equiv 7\cdot 4\equiv 5 \pmod{23}$.
[$8^{-1}\equiv 3 \pmod{23}$, so that $8^{-3}\equiv 3^3\equiv 4 \pmod{23}$. Or, use Fermat: $8^{-3}\equiv 8^{19}\equiv 4 \pmod{23}$.]

(e) In this case, $m=19\cdot 18^{-3}\equiv 19\cdot 16\equiv 5 \pmod{23}$.