ElGamal Cryptosystem

DSC 4110/6111: Modern Cryptography and AI/ML Security

Dr. Laltu Sardar

School of Data Science,
Indian Institute of Science Education and Research Thiruvananthapuram (IISER TVM)

August 12, 2026

Table of Content

Diffie–Hellman Key Exchange

A reminder

Discrete Logarithm Problem and Diffie–Hellman Key Exchange

Discrete Logarithm Problem (DLP): Given a cyclic group G, generator g, and $y = g^x$, it is easy to compute y from x, but computationally hard to recover x from (g, y).

Diffie–Hellman (DH) Hardness Assumption: Given a cyclic group G, generator g, and g^a, g^b, it is computationally hard to compute g^{ab} from (g, g^a, g^b).

Diffie–Hellman Key Exchange

Public parameters: (G, q, g), where G is a cyclic group of order q and g is a generator.

Alice chooses $a \xleftarrow{\$} \mathbb{Z}_q, \qquad A = g^a.$

Bob chooses $b \xleftarrow{\$} \mathbb{Z}_q, \qquad B = g^b.$

They exchange A and B and compute

$$K_A = B^a = g^{ab}, \qquad K_B = A^b = g^{ab}.$$

Thus,

$$\boxed{K_A = K_B = g^{ab}}.$$

Security: An eavesdropper sees g^a and g^b but should not be able to compute g^{ab} under an appropriate DH hardness assumption.

ElGamal Encryption Scheme

ElGamal Encryption Scheme

Let (G, q, g) be public parameters, where G is a cyclic group of prime order q generated by g.

KeyGen

Choose

$$x \xleftarrow{\$} \mathbb{Z}_q.$$

Compute

$$y = g^x.$$

$$\boxed{pk = y, \quad sk = x}$$

Enc

For $m \in G$, choose fresh

$$r \xleftarrow{\$} \mathbb{Z}_q.$$

Compute

$$c_1 = g^r,$$

$$c_2 = m\, y^r.$$

$$\boxed{c = (c_1, c_2)}$$

Dec

Using $sk = x$, compute

$$m = c_2 (c_1^x)^{-1}.$$

Since

$$c_1^x = g^{rx} = y^r,$$

$$\boxed{m = c_2 (c_1^x)^{-1}}$$

Security

ElGamal security is based on the hardness of discrete-logarithm-related problems in the chosen group. Fresh r makes encryption probabilistic.

ElGamal Signature Scheme

ElGamal-Style Digital Signature

Let $G = \langle g \rangle$ be a cyclic group of prime order q, and let H be a cryptographic hash function.

KeyGen

Choose

$$x \xleftarrow{\$} \mathbb{Z}_q^*.$$

Compute

$$y = g^x.$$

$$\boxed{pk = y, \quad sk = x}$$

Sign

Compute $h = H(m)$. Choose fresh

$$k \xleftarrow{\$} \mathbb{Z}_q^*.$$

Compute

$$r = g^k,$$

$$s = k^{-1}(h - xr) \pmod{q}.$$

$$\boxed{\sigma = (r, s)}$$

Verify

Compute

$$h = H(m).$$

Accept iff

$$\boxed{g^h \equiv y^r r^s \pmod{q}}$$

Security

Security is based on the discrete-logarithm-related hardness of the chosen group. The ephemeral k must be fresh and secret for every signature.

ElGamal-Style Signature: Correctness

During signing,

$$s = k^{-1}(H(m) - xr) \pmod{q}.$$

Therefore,

$$ks \equiv H(m) - xr \pmod{q},$$

and hence

$$xr + ks \equiv H(m) \pmod{q}.$$

During verification,

$$y^r r^s = (g^x)^r (g^k)^s = g^{xr+ks}.$$

Using

$$xr + ks \equiv H(m) \pmod{q},$$

we obtain

$$y^r r^s \equiv g^{H(m)} \pmod{q}.$$

Thus the verification equation

$$\boxed{g^{H(m)} \equiv y^r r^s \pmod{q}}$$

holds for every correctly generated signature.

References

Textbook:

1. Douglas R. Stinson and Maura Paterson. *Cryptography: Theory and Practice*. 4th Edition. Chapman & Hall/CRC Press, 2019. ISBN: 9780367148591.

Acknowledgment:

- The figures in this work were generated with the assistance of ChatGPT (OpenAI).

Dr. Laltu Sardar, Assistant Professor, IISER Thiruvananthapuram
laltu.sardar@iisertvm.ac.in, laltu.sardar.crypto@gmail.com
Course webpage: https://laltu-sardar.github.io/courses/26_crypto.html.