Trong trao đổi khóa EC Diffie-Hellman, ta chọn một điểm G có hạng n lớn, và giao thức trao đổi khóa giữa Alice và Bob tiến hành như sau:

1) Alice chọn một số $n_A < n$ và giữ bí mật số n_A này. Sau đó trong $E_q(a,b)$ Alice tính $P_A = n_A G$ và gửi P_A cho Bob.
2) Tương tự Bob chọn một số bí mật n_B, tính P_B và gửi P_B cho Alice.
3) Alice tạo khóa phiên bí mật là $K = n_A P_B = n_A n_B G$
4) Bob tạo khóa phiên bí mật là $K = n_B P_A = n_B n_A G = n_A n_B G$ (nhóm Abel có tính giao hoán) giống với khóa của Alice.

Trudy có thể chặn được P_A và P_B, tuy nhiên chỉ có thể tính được:

$$P_A + P_B = n_A G + n_B G = (n_A + n_B)G$$

Để tính được $K = n_A n_B G$, Trudy phải tìm được n_A, n_B từ P_A, P_B và G . Tuy nhiên điều này là bất khả thi như ta đã thấy ở phần trên.

Chú ý: khóa phiên *K* là một điểm trong đường cong Elliptic, để sử dụng khóa này cho mã hóa đối xứng như DES hay AES, ta cần chuyển *K* về dạng số thường.

10.4.2 Mã hóa và giải mã EC

Tương tự như vấn đề trao đổi khóa, trong vấn đề mã hóa/giải mã, ta cũng chọn các tham số để tạo một nhóm Abel $E_q(a,b)$ và chọn một điểm cơ sở G có hạng n lớn.

Các thành phần khóa khóa riêng và công khai trong mã hóa EC được định nghĩa như sau:

$$K_R = (d, G, q, a, b)$$
$$K_U = (E, G, q, a, b)$$

Trong đó $d < n$ và $E = dG$ với d là một số bí mật do người sinh khóa chọn. Do tính chất của hàm một chiều từ E và G không thể suy ra được d.

Từ đó chúng ta có hai cách thức thực hiện mã hóa/ giải mã như sau:

1) Phương pháp Elgamal:

Giả sử Alice muốn gửi một thông điệp M cho Bob, trước tiên Alice chuyển M từ dạng dãy bít sang dạng điểm $P_M = (x, y)$. Bản mã C_M (dùng khóa công khai của Bob) được tính là một cặp điểm như sau:

$$C_M = \{kG, P_M + kE)$$ với k là một số ngẫu nhiên do Alice chọn

Để giải mã dùng khóa riêng, Bob sẽ nhân điểm thứ nhất trong C_M với d, sau đó lấy điểm thứ hai trừ cho kết quả:

$$P_M + kE - dkG = P_M + kdG - kdG = P_M$$

Trong phương thức mã hóa, Alice đã che giấu P_M bằng cách cộng P_M với kE. Để giải mã, Bob cần trừ ra lại kE. Thay vì gửi trực tiếp k cho Bob để Bob tính kE (Trudy có thể chặn được), Alice gửi một *dấu hiệu* là kG . Dựa vào kG và d, Bob có thể tính kE. Còn Trudy, dù biết G và kG, tuy nhiên vẫn không thể tính được k do tính chất của hàm một chiều.

Ví dụ: chọn p = 751, a = 1, b = 188 ta có đường cong Elliptic trên Z_{751} như sau

$$y^2 \bmod 751 = (x^3 + x + 188) \bmod 751 \quad a, b, x, y \in Z_{751}$$

Chọn điểm cơ sở là G =(0, 376).

Giả sử Alice cần mã hóa bản rõ là điểm P_M = (562, 201) dùng khóa công khai E = (201, 5). Alice chọn k = 386. Ta có:

386(0, 376) = (676, 558)

(562,201) + 386(201, 5) = (385, 328)

Vậy bản mã là cặp điểm { (676, 558), (385, 328) }

2) *Phương pháp Menezes - Vanstone:*

Thông điệp M của Alice được tách thành hai phần M=(m_1, m_2) sao cho m_1, $m_2 \in Z_p$. Alice chọn một số ngẫu nhiên k, kết hợp với khóa công khai của Bob, Alice tính điểm P như sau:

$$P(x_P, y_P) = kE$$

Bản mã C_M gồm ba thành phần:

$$C_M = \{c_0, c_1, c_2\} = \{kG\ ,\ \ x_P m_1 \ \ mod\ p,\ \ y_P m_2 \ \ mod\ p\ \}$$

Để giải mã dùng khóa riêng, từ dấu hiệu kG, Bob tính:

$$P(x_P, y_P) = dkG$$

và từ đó tính nghịch đảo của ${x_P}^{-1}$ và ${y_P}^{-1}$ trong phép $modulo\ p$. Cuối cùng, bản giải mã là:

$$M = \{m_1, m_2\} = \{\ {x_P}^{-1} c_1\ mod\ p\ \ ,\ \ {y_P}^{-1} c_2\ mod\ p\ \}$$

Tương tự như phương pháp Elgamal, dù biết G và kG, Trudy cũng không thể tính được k để tính P.

10.4.3 Độ an toàn của ECC so với RSA

Hiện nay, phương pháp nhanh nhất để tính logarit đường cong Elliptic (tính k biết G và kG) là phương pháp Pollar rho. Bảng sau đây liệt kê kích thước khóa của phương pháp ECC và phương pháp RSA dựa trên sự tương đương về chi phí phá mã.

Mã hóa đối xứng (số bít của khóa)	*Mã hóa ECC (số bít của n)*	*Mã hóa RSA (số bít của N)*
56	112	512
80	160	1024
112	224	2048
128	256	3072
192	384	7680
256	512	15360

Như vậy với cùng một độ an toàn thì mã hóa ECC chỉ dùng các phép tính có số bít nhỏ hơn nhiều lần so với mã hóa RSA.

10.5 Chuẩn chữ ký điện tử (Digital Signature Standard – DSS)

Trong chương 5, chúng ta đã tìm hiểu về cách sử dụng hàm hash cùng với mã hóa RSA để tạo chữ ký điện tử. Hình bên dưới trình bày lại mô hình này: