Việc chuyển đổi từ PBC (Pairing-Based Cryptography Library) sang RELIC trong libbswabe nhằm cải thiện hiệu suất, bảo mật, và tính linh hoạt. Dưới đây là danh sách các thay đổi và ý nghĩa của chúng.

1. **Thay đổi về cài đặt và biên dịch**

|  |  |  |  |
| --- | --- | --- | --- |
| **Yếu tố** | **Trước (PBC)** | **Sau (RELIC)** | **Ý Nghĩa** |
| Thư viện sử dụng | -lpbc | -lrelic | RELIC nhẹ hơn, tối ưu hiệu suất |
| Khởi tạo | pairing\_t pairing; | KHÔNG cần pairing\_t | RELIC tự động thiết lập nhóm |

1. **Thay đổi về kiểu dữ liệu**

|  |  |  |  |
| --- | --- | --- | --- |
| **Mô tả** | **Trước (PBC)** | **Sau (RELIC)** | **Ý Nghĩa** |
| G1 | element\_t g; element\_init\_G1(g, pairing); | g1\_t g; g1\_new(g); | Phân biệt nhóm ngay từ khi khai báo |
| G2 | element\_t gp; element\_init\_G2(gp, pairing); | g2\_t gp; g2\_new(gp); | Giảm nguy cơ sai nhóm |
| GT | element\_t g\_hat\_alpha; element\_init\_GT(g\_hat\_alpha, pairing); | gt\_t g\_hat\_alpha; gt\_new(g\_hat\_alpha); | Tối ưu hiệu suất |
| Zr | element\_t beta; element\_init\_Zr(beta, pairing); | bn\_t beta; bn\_new(beta); | Quản lý bố số hiệu quả |

1. **Sinh số ngẫu nhiên**

|  |  |  |
| --- | --- | --- |
| **Trước (PBC)** | **Sau (RELIC)** | **Ý Nghĩa** |
| element\_random(beta); | bn\_rand\_mod(beta, order); | Kiểm soát giá trị tốt hơn |

1. **Thực hiện phép nhân trong các nhóm**

|  |  |  |  |
| --- | --- | --- | --- |
| **Phép toán** | **Trước (PBC)** | **Sau (RELIC)** | **Ý Nghĩa** |
| Nhân G1 | element\_mul(h, g, beta); | g1\_mul(h, g, beta); | Tăng hiệu suất |
| Nhân G2 | element\_mul(gp, g\_alpha, beta); | g2\_mul(gp, g\_alpha, beta); | An toàn hơn |
| Pairing | element\_pairing(g\_hat\_alpha, g, g\_alpha); | pc\_map(g\_hat\_alpha, g, g\_alpha); | Hỗ trợ nhiều đường cong |

1. **Thay đổi cách ánh xạ chuỗi thành phần tử nhóm**

|  |  |  |
| --- | --- | --- |
| **Trước (PBC)** | **Sau (RELIC)** | **Ý Nghĩa** |
| element\_from\_hash(h, str, strlen(str)); | g1\_map(h, (uint8\_t\*)str, strlen(str)); | Chính xác hơn |

1. **Serialization (Lưu và đọc dữ liệu)**

|  |  |  |
| --- | --- | --- |
| **Trước (PBC)** | **Sau (RELIC)** | **Ý Nghĩa** |
| element\_to\_bytes(buf, g); | g1\_write\_bin(buf, size, g, 1); | Dễ đọc và ghi hơn |
| element\_from\_bytes(g, buf); | g1\_read\_bin(g, buf, size); | Chuẩn hóa |

1. **Giải phóng bộ nhớ**

|  |  |  |
| --- | --- | --- |
| **Trước (PBC)** | **Sau (RELIC)** | **Ý Nghĩa** |
| element\_clear(g); | g1\_free(g); | Tránh rò rỉ bộ nhớ |

1. **Kết luận**

|  |  |
| --- | --- |
| **Điểm cải tiến** | **Lý do** |
| **Hiệu suất cao hơn** | RELIC có thuật toán nhanh hơn |
| **Bảo mật tốt hơn** | RELIC chống tấn công side-channel |
| **Quản lý bộ nhớ tốt hơn** | Giảm rò rỉ RAM |
| **API rõ ràng hơn** | Dễ dùng, dễ debug |