**🎯 Mục tiêu**

**Tìm flag bằng cách thu hồi lại giá trị a từ phương trình:**

**hint = g^a mod p**

**Sau đó dùng a làm key AES để giải mã ciphertext.**

**📄 Thông tin thu được**

**Từ các file main.py và cycles.txt, ta có:**

* **g = 3**
* **p là một số nguyên tố lớn**
* **hint = 1**
* **ciphertext: 48 bytes, mã hóa bằng AES ECB**
* **Mã hóa AES:**

**key = long\_to\_bytes(a)[:16]**

**cipher = AES.new(key, AES.MODE\_ECB)**

**ct = cipher.encrypt(pad(flag, AES.block\_size))**

**👉 AES sử dụng 16 byte đầu tiên của a làm key.**

**🔍 Phân tích kỹ thuật**

**✅ Dòng quan trọng:**

**hint = pow(g, a, p)**

**Biết rằng:**

**hint = 1**

**→ Tức là:**

**g^a ≡ 1 mod p**

**🧠 Phân tích toán học**

**Mục tiêu:**

**Giải phương trình rời rạc:**

**3^a ≡ 1 mod p**

**Khi nào g^a ≡ 1 mod p?**

**Điều này xảy ra khi:**

**a ≡ 0 mod ord\_p(g)**

**Tức là:**

**a = k × ord\_p(g)**

**Vì g = 3 và p là số nguyên tố lớn → theo Định lý Fermat nhỏ:**

**g^(p−1) ≡ 1 mod p**

**Nếu g là nguyên thủy modulo p thì:**

**ord\_p(g) = p − 1**

**=> Khi hint = 1 → suy ra:**

**a = k × (p - 1)**

**→ Giá trị a là bội của (p - 1)**

**🛠️ Quy trình giải**

**✅ Bước 1: Nhận ra cấu trúc**

**hint = g^a mod p = 1**

**=> a = k × (p - 1) với k ∈ [1, N]**

**→ Ta brute-force giá trị k nhỏ.**

**✅ Bước 2: Tạo AES key từ a**

* **Convert a → bytes**
* **Lấy key = first 16 bytes of a**
* **Dùng AES ECB để giải mã:**

**cipher = AES.new(key, AES.MODE\_ECB)**

**pt = unpad(cipher.decrypt(ciphertext), AES.block\_size)**

**✅ Bước 3: Kiểm tra kết quả**

* **Nếu plaintext:**
  + **Có thể decode UTF-8**
  + **Có dạng flag hợp lệ → thành công!**

**⚠️ Lỗ hổng chính**

* **Việc chọn a sao cho g^a ≡ 1 mod p khiến bài toán mất tính một chiều**
* **Không cần dùng các thuật toán Discrete Log mạnh như:**
  + **Baby-Step Giant-Step**
  + **Pollard’s Rho**
* **Thay vào đó, chỉ cần brute-force với a = k × (p - 1)!**

**✅ Kết luận**

**Đây là một bài khai thác sai sót trong thiết kế số mũ a.  
Việc chọn a thuộc bội số của (p − 1) khiến phép g^a mod p trở nên dễ bị dự đoán — từ đó phá vỡ hoàn toàn hệ mã hóa AES, và khôi phục được flag.**