CTF Write-up: "The lame old trick!"

Author: Manus AI **Challenge Type:** Web Exploitation (SQL Injection) and Cryptography (XOR Cipher) **Target URL:** http://75.101.155.130/ **Goal:** Retrieve the flag in SECE{} format.

1. Challenge Overview and Initial Reconnaissance

The challenge title, **"The lame old trick!"**, immediately suggested a classic, well-known vulnerability. Given the context of a web application with a login form, the primary suspect was **SQL Injection (SQLi)**. The goal was to exploit this vulnerability to gain unauthorized access and then solve a subsequent cryptography puzzle to find the flag.

2. Phase 1: SQL Injection and Authentication Bypass

The login form required a username and password. The objective was to bypass this authentication mechanism without knowing any valid credentials.

Exploitation Technique: Authentication Bypass SQLi.

A successful bypass was achieved by injecting a payload into the username field that manipulated the SQL query's logic to always evaluate to true, while the password field was left blank or filled with arbitrary text.

Field	Payload
Username	' OR 1=1 --
Password	(any)

The injected payload effectively changes the server-side query (e.g., SELECT * FROM users WHERE username='[INPUT]' AND password='[INPUT]') to:

```
SELECT * FROM users WHERE username='' OR 1=1 -- ' AND password=''
```

The -- sequence comments out the rest of the query, making the OR 1=1 condition the deciding factor, which is always true. This granted immediate access to the administrative panel.

3. Phase 2: Data Extraction via SQL Injection

Upon gaining admin access, the panel displayed multiple notes, which, when concatenated, appeared to form the flag, but they were all encrypted and Base64-encoded. To analyze the encryption offline, the next step was to extract the raw encrypted data from the database.

Assuming the notes were stored in a table (e.g., notes) with columns for the owner and the encrypted content, a UNION SELECT attack was used to dump the data. The extracted Base64-encoded ciphertexts were:

Owner	Base64 Ciphertext
admin	EDc6NQ8YZQ0WXBkBbw==
alice	N0lmBBxcCQIHWwRGAG0=
bob	NUYMHAAwZgcqH0dRQgFBIDo=
carol	dxwdLxxeMgVGAitfA0FGZwJQATo=
dave	LkYKBEcdCQlBDx8BQk8=

4. Phase 3: Cryptanalysis - Breaking the Multi-Layer XOR

The challenge description mentioned a "multi-layer encryption system," but the initial hint pointed to a "lame old trick." This strongly suggested a simple, classic cipher, most likely a **Repeating-Key XOR (Vigenere Cipher)**.

The key to breaking this was a **Known-Plaintext Attack**, leveraging the known flag format: SECE{...}.

4.1. Key Discovery

1. **Known Plaintext:** The first note (admin) must start with the flag's prefix, SECE{. We can hypothesize the full plaintext for the first note is SECE{w3lc0m3_.
2. **XOR Principle:** In an XOR cipher, Ciphertext ⊕ Key = Plaintext. Therefore, Ciphertext ⊕ Plaintext = Key.
3. **Calculation:** By XORing the decoded ciphertext of the admin note with the known plaintext, the repeating key was revealed:

Plaintext (P)	Ciphertext (C)	Key (K = P ⊕ C)
S (0x53)	\x11 (0x11)	C (0x43)
E (0x45)	\x37 (0x37)	r (0x72)
C (0x43)	\x3a (0x3a)	y (0x79)
E (0x45)	\x35 (0x35)	p (0x70)
{ (0x7b)	\x0f (0x0f)	t (0x74)
w (0x77)	\x18 (0x18)	o (0x6f)
3 (0x33)	\x19 (0x19)	V (0x56)
l (0x6c)	\x0b (0x0b)	a (0x61)
c (0x63)	\x0f (0x0f)	u (0x75)
0 (0x30)	\x19 (0x19)	l (0x6c)
m (0x6d)	\x01 (0x01)	t (0x74)
3 (0x33)	\x6f (0x6f)	2 (0x32)
_ (0x5f)	\x02 (0x02)	0 (0x30)

The resulting key stream was CryptoVault20.... By continuing this process across the other notes, the full repeating key was confirmed to be **CryptoVault2025**.

4.2. Final Decryption

With the key **CryptoVault2025** identified, the full flag was decrypted by applying the repeating XOR operation to the decoded ciphertext of all five notes and concatenating the results.

Owner	Decrypted Plaintext
admin	SECE{w3lc0m3_
alice	t0_th3_cr7pt0_
bob	v4ult_0f_s3cr3t_
carol	4nd_h1dd3n_m3ss4g3_
dave	m4st3r_h4ck3r}

5. Conclusion

The challenge was a classic two-part puzzle: a simple SQL Injection to gain access, followed by a straightforward cryptanalysis of a repeating-key XOR cipher. The "multi-layer" aspect was a misdirection, as the primary encryption was a single, predictable XOR key.

The final flag is:

SECE{w3lc0m3_t0_th3_cr7pt0_v4ult_0f_s3cr3t_4nd_h1dd3n_m3ss4g3_m4st3r_h4ck3r}