Royal ECDSA – Complete Challenge Writeup

Royal ECDSA is a cryptography challenge based on a critical misuse of the ECDSA signing algorithm. The server reuses the same nonce value while signing multiple messages, making it possible to recover the private key and forge arbitrary signatures.

Step 1: Reconnaissance

The web application exposes an endpoint that returns previously signed messages along with their ECDSA signatures. By visiting the */messages* endpoint, an attacker can collect multiple message–signature pairs.

Step 2: Identify Nonce Reuse

Upon inspecting the signatures, it can be observed that two different messages share the same value of *r*. In ECDSA, identical *r* values indicate reuse of the same ephemeral nonce *k*, which is a fatal cryptographic flaw.

Step 3: Recover the Nonce

Using the two signatures generated with the same nonce, the value of *k* can be recovered mathematically. This is achieved by subtracting the ECDSA equations derived from the two signatures and solving for *k*.

Step 4: Recover the Private Key

Once the nonce is known, the ECDSA private key *d* can be computed directly from one of the signatures. This completely compromises the cryptographic identity of the server.

Step 5: Forge a Royal Order

With the recovered private key, the attacker can now generate a valid ECDSA signature for any chosen message. A message starting with *ROYAL_ORDER:* is crafted and signed using the recovered key.

Step 6: Retrieve the Flag

The forged message and its valid signature are submitted to the */submit* endpoint. Since the signature verifies correctly and the message satisfies the required prefix, the server returns the flag.

Flag

LNMHACKS{winter_is_com1ng_the_things_i_do_for_love}