

Міністерство освіти і науки України
НТУУ «Київський політехнічний інститут»
Фізико-технічний інститут

Системи та засоби інтерактивної аналітики
Лабораторна робота №4
Запити розширеного SQL (advanced SQL). RegExp
Варіант №6

Виконав:
Студент 4-го курсу
групи ФІ-21
Климентьєв Максим
Перевірив:

Зміст

1	Мета роботи	3
2	Завдання	4
3	Код реалізації	6
4	Висновки	11
5	Контрольні питання	12

1 Мета роботи

Навчитися створювати розширені SQL запити і опанувати роботу з RegExpr.

2 Завдання

```
13.66.139.0 - - [19/Dec/2020:13:57:26 +0100] "GET /index.php?option=com_phocagallery&view=category&id=1:almhuetter-raith&Itemid=53 HTTP/1.1" 200 32653 "-" "Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.htm)" "-"

157.48.153.185 - - [19/Dec/2020:14:08:06 +0100] "GET /apache-log/access.log HTTP/1.1" 200 233 "-" "Mozilla/5.0 (Windows NT 6.3; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36" "-"

157.48.153.185 - - [19/Dec/2020:14:08:08 +0100] "GET /favicon.ico HTTP/1.1" 404 217 "http://www.almhuetter-raith.at/apache-log/access.log" "Mozilla/5.0 (Windows NT 6.3; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36" "-"

216.244.66.230 - - [19/Dec/2020:14:14:26 +0100] "GET /robots.txt HTTP/1.1" 200 304 "-" "Mozilla/5.0 (compatible; DotBot/1.1; http://www.opensiteexplorer.org/dotbot, help@moz.com)" "-"

54.36.148.92 - - [19/Dec/2020:14:16:44 +0100] "GET /index.php?option=com_phocagallery&view=category&id=2%3Awinterfotos&Itemid=53 HTTP/1.1" 200 30662 "-" "Mozilla/5.0 (compatible; AhrefsBot/7.0; +http://ahrefs.com/robot/)" "-"

92.101.35.224 - - [19/Dec/2020:14:29:21 +0100] "GET /administrator/index.php HTTP/1.1" 200 4263 "" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)" "-"
```

Рисунок 4.1 - Приклад файлу access.log

Формат файлу наступний.

```
%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"
```

- %h - хост/IP-адреса, з якої зроблено запит до сервера;
- %t - час запиту до сервера та часовий пояс сервера;
- %g - тип запиту, його вміст та версія;
- %s - код стану HTTP;
- %b - кількість відданих сервером байт;
- %{Referer} - URL-джерело запиту;
- %{User-Agent} - HTTP-заголовок, що містить інформацію про запит (клієнтський додаток, мову і т. д.);
- %{Host} - ім'я Virtual Host, до якого йде звернення.

Цей файл приєднаний до завдання (tblaccesslog.sql), треба його завантажити у БД зроблену у попередніх роботах. (PhpMyAdmin → import). Вікно імпорту представлено на Рис. 4.2

Гакери намагаються зламати ваш сайт і зіпсувати базу даних. Треба знайти записи, з нестандартними діями, ідентифікувати IP адреси комп'ютерів з яких відбувається атака і передати ці записи до кіберполіції. Можна створювати додаткові таблиці.

Запити повинні фільтрувати інформацію точно, а не близько до завдання.

1. До БД створеної у попередніх роботах імпортувати таблицю з даними.
2. Зробити запити наведені у Табл. 4.1.

Обов'язково використати regexr.

Зверніть увагу, кодів завершення багато.

3. Створити звіт. Приєднати до класу.

- У звіті навести все необхідне для повторення і перевірки ваших дій (діаграму БД, SQL запити для створення БД і таблиць, структуру БД, і т.д). Навести знімки екрана, які підтверджують виконані дії.
- У протоколі SQL запити наводити у текстовому вигляді щоб їх можна було редагувати і модифікувати під час захисту.
- Зробити висновки по роботі і занести їх у звіт.

4. Підготувати відповіді на контрольні питання (для офлайн захисту навести їх у протоколі, розкрити сутність, навести приклади).

5. Захистити роботу.

№ варіанта	Завдання
6	<u>Використати існуючу БД (лаб. роб. 2,3)</u> Запити: 1. Вивести повні адреси JS скриптів (розширення js) і їх розмір, які запитали з комп'ютера з IP адресою 83.227.29.211, для яких запит завершився вдало. 2. Придумайте будь який запит з використанням UNION, в дослідити різні форми цієї команди. 3. Знайдіть розмір всього скачаного комп'ютером з IP адресою 83.227.29.211

3 Код реалізації

Імпортуння до бази даних "plant_store"

Файл для імпорту:

Файл може бути стиснений (gzip, bzip2) або нестиснений.
Ім'я стисненого файлу повинно закінчуватися на `.[format].[compression]`. Приклад: `.sql.zip`

Переглянути Ваш комп'ютер: (Макимум: 40MB)

Ви також можете перетягнути файл на будь-якій сторінці.

Кодування файлу:

Частковий імпорт:

☒ Дозволити переривання імпорту у випадку, коли скрипт виявить наближення до вичерпання часу очікування PHP.
Це може бути хорошим способом для імпорту файлів великого розміру, проте це може призвести до переривання транзакції.

Пропустити вказане число запитів (для SQL), починаючи з першого:

Інші параметри

☒ Відключити перевірку зовнішніх ключів

Формат

Параметри форматування:

Режим сумісності SQL:

☒ Не використовувати AUTO_INCREMENT для нульових значень

✓ Імпорт завершився вдало, 31 запит виконано. (tblaccesslog.sql)

1. Вивести повні адреси JS скриптів (розширення js) і їх розмір, які запросили з комп'ютера з IP адресою 83.227.29.211, для яких запит завершився вдало.

```
SELECT
    Script,
    Size
FROM (SELECT
    SUBSTRING_INDEX(Line, ' ', 1) as IP,
    SUBSTRING_INDEX(SUBSTRING_INDEX(SUBSTRING_INDEX(SUBSTRING_INDEX
(Line, '"', 2), '"', -1), ' ', 2), ' ', -1) as Script,
    SUBSTRING_INDEX(SUBSTRING_INDEX(Line, '"', 2), ' ', -1) as
Size,
    SUBSTRING_INDEX(SUBSTRING_INDEX(SUBSTRING_INDEX(Line, '"', 2),
' ', -2), ' ', 1) as Status
FROM `tblaccesslog`) as first_task
WHERE first_task.IP = '83.227.29.211'
AND first_task.Status REGEXP '^2'
AND first_task.Script REGEXP '\\.js(\\?.*)?$';
```

Script	Size
/media/system/js/modal.js	10588
/components/com_phocagallery/assets/js/shadowbox/shadowbox.js	27272
/components/com_phocagallery/assets/js/shadowbox/src/lang/shadowbox-en.js	2337
/components/com_phocagallery/assets/js/shadowbox/src/skin/classic/skin.js	3495
/components/com_phocagallery/assets/js/shadowbox/src/player/shadowbox-img.js	8324

2. Придумайте будь який запит з використанням UNION, в дослідити різні форми цієї команди.

```
(SELECT
  IP,
  Size
FROM (SELECT
  SUBSTRING_INDEX(Line, ' ', 1) as IP,
  SUBSTRING_INDEX(SUBSTRING_INDEX(Line, ' "', 2), ' ', -1) as
  Size,
  SUBSTRING_INDEX(SUBSTRING_INDEX(SUBSTRING_INDEX(Line, ' "', 2),
  ' ', -2), ' ', 1) as Status
FROM `tblaccesslog`) as second_task
WHERE second_task.IP REGEXP '173.255.176.5|83.169.39.166'
AND second_task.Size = 0
AND second_task.Status REGEXP '^2')
UNION
(SELECT
  IP,
  Size
FROM (SELECT
  SUBSTRING_INDEX(Line, ' ', 1) as IP,
  SUBSTRING_INDEX(SUBSTRING_INDEX(Line, ' "', 2), ' ', -1) as
  Size,
  SUBSTRING_INDEX(SUBSTRING_INDEX(SUBSTRING_INDEX(Line, ' "', 2),
  ' ', -2), ' ', 1) as Status
FROM `tblaccesslog`) as second_task
WHERE second_task.IP REGEXP '83.227.29.211'
AND second_task.Size < 200
AND second_task.Status REGEXP '^2');
```

IP	Size
173.255.176.5	0
83.169.39.166	0
83.227.29.211	174
83.227.29.211	155


```

(SELECT
  IP,
  Size
FROM (SELECT
  SUBSTRING_INDEX(Line, ' ', 1) as IP,
  SUBSTRING_INDEX(SUBSTRING_INDEX(Line, ' "', 2), ' ', -1) as
  Size,
  SUBSTRING_INDEX(SUBSTRING_INDEX(SUBSTRING_INDEX(Line, ' "', 2),
  ' ', -2), ' ', 1) as Status
FROM `tblaccesslog`) as second_task
WHERE second_task.IP REGEXP '173.255.176.5|83.169.39.166'
AND second_task.Size = 0
AND second_task.Status REGEXP '^2')
UNION ALL
(SELECT
  IP,
  Size
FROM (SELECT
  SUBSTRING_INDEX(Line, ' ', 1) as IP,
  SUBSTRING_INDEX(SUBSTRING_INDEX(Line, ' "', 2), ' ', -1) as
  Size,
  SUBSTRING_INDEX(SUBSTRING_INDEX(SUBSTRING_INDEX(Line, ' "', 2),
  ' ', -2), ' ', 1) as Status
FROM `tblaccesslog`) as second_task
WHERE second_task.IP REGEXP '83.227.29.211'
AND second_task.Size < 200
AND second_task.Status REGEXP '^2');

```

IP	Size
173.255.176.5	0
173.255.176.5	0
173.255.176.5	0
173.255.176.5	0
173.255.176.5	0
173.255.176.5	0
173.255.176.5	0
173.255.176.5	0
173.255.176.5	0
83.169.39.166	0
83.227.29.211	174
83.227.29.211	155

3. Знайдіть розмір всього скачаного комп'ютером з IP адресою 83.227.29.211

```
SELECT
    IP,
    sum(Size) as Total
FROM (SELECT
        SUBSTRING_INDEX(Line, ' ', 1) as IP,
        SUBSTRING_INDEX(SUBSTRING_INDEX(Line, '"', 2), ' ', -1) as
    Size
    FROM `tblaccesslog`) as third_task
WHERE third_task.IP = '83.227.29.211'
GROUP BY third_task.IP;
```

IP	Total
83.227.29.211	462376

4 Висновки

Набуто навички роботи з RegExр у SQL-запитах.

5 Контрольні питання

1. Що таке запити розширеного SQL (advanced SQL)?

Розширені SQL-запити — це запити, які використовують підзапити, об'єднання, агрегації та умови.

2. Що таке регулярні вирази?

Регулярні вирази — це спеціальна мова шаблонів для пошуку або перевірки текстових даних.

Приклад:

```
SELECT * FROM logs WHERE Line REGEXP '\\.js$';
```

Знайде всі рядки, що закінчуються на .js.

3. Для чого використовуються запити з командою UNION?

Команда UNION об'єднання результатів двох селектив виключаючи повторні рядки, тоді як UNION ALL залишає всі записи.

```
SELECT name FROM students
UNION
SELECT name FROM teachers;
```

Поверне список усіх імен без повторів.

4. Що таке агрегатні функції?

Агрегатні функції — це функція, які повертають одинарне значення з колекції вхідних значень такої як множина. До них належать:

- COUNT () — кількість рядків;
- SUM () — сума значень;
- AVG () — середнє значення;
- MIN (), MAX () — мінімальне та максимальне значення.

5. Для чого використовуються запити з командою ORDER?

Команда ORDER BY використовується для впорядкування (сортування) набору результатів у порядку зростання або спадання.

Наприклад, якщо потрібно продивитись юзерів за спаданням їх віку.

```
SELECT name, age FROM users ORDER BY age DESC;
```

6. Для чого використовуються запити з командою HAVING?

Команда HAVING використовується якщо треба накласти умову на результат агрегатної функції.

```
SELECT department, COUNT(*) AS workers  
FROM employees  
GROUP BY department  
HAVING workers > 10;
```

Повертає лише ті відділи, де кількість працівників більша за 10.