# 数据库结构

## 公用结构

### 表cve

- 功能：保存CVE信息

- 列：

- cve\_num：CVE编号（CVE-2019-xxxxx） （主键 varchar 50 ）

- level：严重级别（致命、高危、中危、低危）

- description：描述

- platform：适用平台（linux、windows、android、ios、all...）

- statment：所需条件（这是一个逻辑表达式，比如：openssl.version >= 1.0.9 && windows.version >= 6.3，按字符串存储即可，格式无需使用数据库技术约束）

- payload：PoC文件的路径（/xxx/xxx/xxx.py），可为空



### 表platform\_risk

- 功能：保存平台和代码安全风险定义

- 列：

- id

- name：风险名称

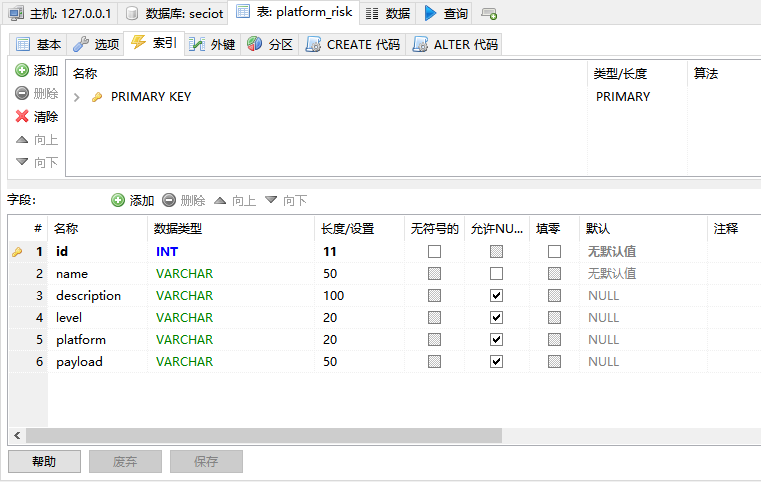
- description：风险描述

- level：严重级别（致命、高危、中危、低危）

- platform：适用平台（linux、windows、android、ios、all...）

- statment：所需条件（这是一个逻辑表达式，比如：openssl.version >= 1.0.9 && windows.version >= 6.3，按字符串存储即可，格式无需使用数据库技术约束）

- payload：PoC文件的路径（/xxx/xxx/xxx.py），可为空



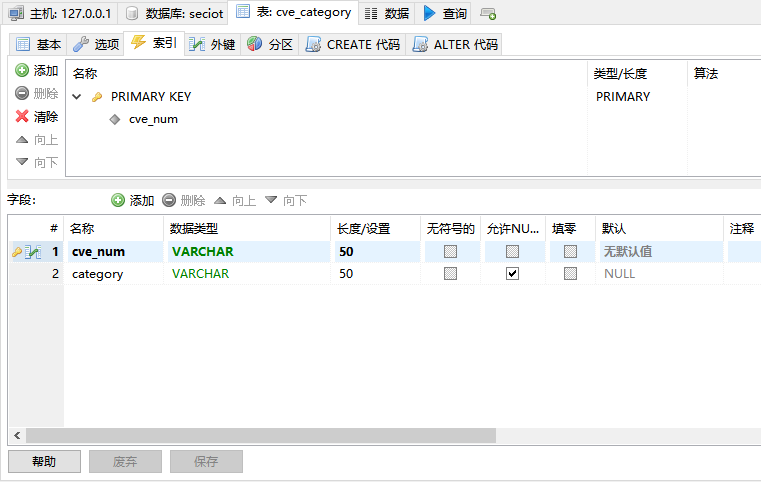
### 表cve\_category

- 功能：关联cve和分析类型

- 列：

- cve\_num：cve表的外键（允许重复，表示多类型的CVE） （）

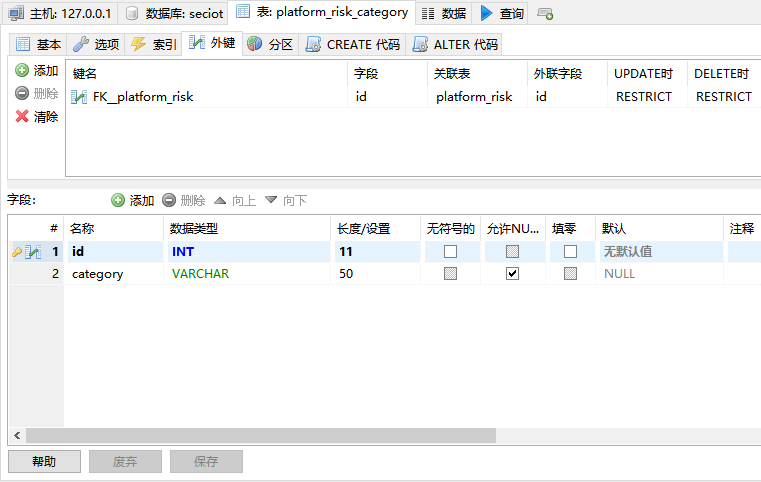
- category：分类（系统固件、安卓应用、云端服务）



### 表platform\_risk\_category

- 功能：关联表platform\_risk和分析类型

- 列：类似cve\_category



### 表fw\_third\_part\_library

- 功能：保存第三方库的种类

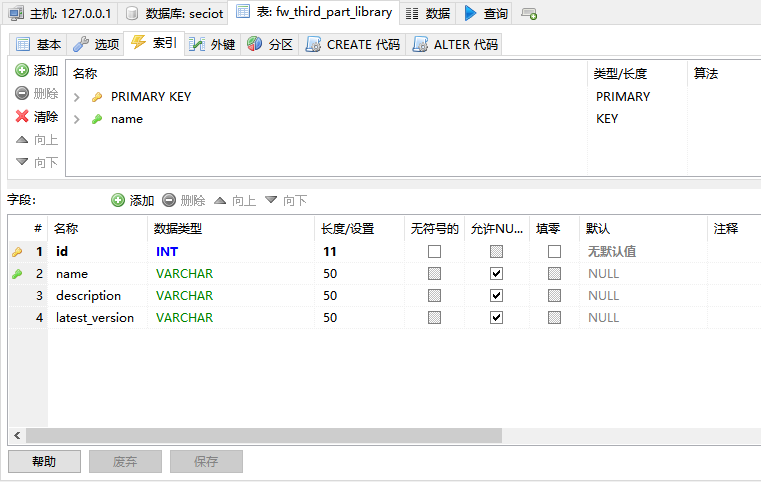
- 列：

- id

- name：库名称

- description：库的描述

- latest\_version：库最新版本



### 表fw\_library\_risk

- 功能：保存某个库按版本号的漏洞情况

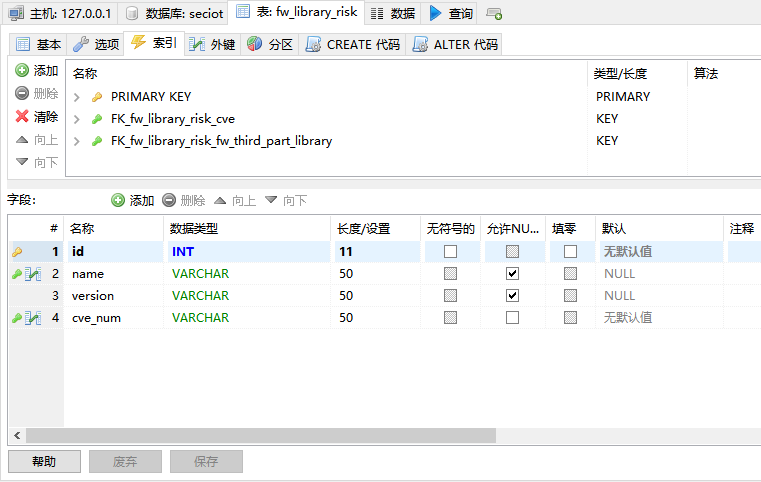
- 列：

- id

- name：fw\_third\_part\_library表的外键 （要和上表关联成外键 上表中name必须得是主键）

- version：版本号（例如1.0.9a）

- cve\_num：cve表的外键



### 表protocol

- 功能：保存各种类型的协议

- 列：

- id

- name：协议名称（例如SSL）

- layer：层（计算机网络中的层：2~5，int类型）

- description：描述



### 表protocol\_risk

- 功能：保存协议安全风险定义

- 列：

- id

- name：风险名称

- description：风险描述

- level：严重级别（致命、高危、中危、低危）

- platform：适用平台（linux、windows、android、ios、all...）

- protocol：协议类型（表protocol外键，例如SSL）

- version：协议版本（例如2.0）

- payload：PoC文件的路径（/xxx/xxx/xxx.py），可为空

