**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Факультет прикладной математики-процессов управления**

**Кафедра “фундаментальная информатика и информационные технологии”**

**отчет**

**по домашнему заданию “Разработка Сетевого Сканера”**

**по дисциплине «Системное программирование в Линукс»**

|  |  |  |
| --- | --- | --- |
| Студент гр. 22Б16 |  | Олизько С.С. |
| Преподаватель |  | Киямов Ж.У. |

**Санкт-Петербург**

**2023 г.**

**Оглавление**

1. Цель работы
2. Задача
3. Теоритическая часть
4. Алгоритм метода
5. Описание программы
6. Рекомендации по использованию
7. Вывод

**Цель:**

Создать инструмент для сканирования и анализа сетевой активности.

**Задача:**

Реализовать функционал сканирования ip адресов и port-ов. Создать графический интерфейс. Реализовать анализ полученных результатов.

**Теоретическая часть:**

IP (Internet Protocol) - это протокол, который обеспечивает уникальную идентификацию и маршрутизацию устройств в сети Интернет. Каждому устройству присваивается уникальный IP-адрес, который состоит из четырех чисел (от 0 до 255), разделенных точками. IP-адрес используется для определения местонахождения устройства в сети и обеспечивает возможность передачи данных между компьютерами.

Порт - это логический конечный пункт в сети, представляющий собой числовой адрес в диапазоне от 0 до 65535. Порты используются для идентификации конкретных процессов или служб на устройстве. Например, веб-сервер может слушать запросы на порту 80, а шифрованный веб-трафик (HTTPS) обычно использует порт 443.

Сеть - это система взаимосвязанных устройств, которые могут обмениваться данными. Устройства в сети могут быть соединены проводными или беспроводными каналами связи, и они используют стандартные протоколы для взаимодействия друг с другом.

Сетевые сканеры - это инструменты, используемые для анализа сети с целью выявления устройств, активных портов и служб, работающих на этих портах. Они могут использоваться для обеспечения безопасности сети, выявления уязвимостей и мониторинга активности в сети. Сетевые сканеры могут выполнять различные виды сканирования, такие как сканирование открытых портов, определение характеристик устройств и сбор информации о сетевой топологии.

**Алгоритм программы:**

1. **Ввод параметров сканирования:**
   1. Пользователь вводит диапазон IP-адресов и портов для сканирования через графический интерфейс.
   2. Можно выбрать опцию сканирования всех портов или указать конкретные порты.
2. **Подготовка к сканированию:**
   1. Загружаются информация о портах из файла "port\_info.txt".
   2. Создаются функции для сканирования HTTP, HTTPS, определения информации о хосте и сканирования портов.
3. **Генерация диапазона IP-адресов:**
   1. Если введен IP-диапазон с дефисами (например, "192.168.0.1-5"), генерируется список IP-адресов в заданном диапазоне.
4. **Сканирование IP-адресов:**
   1. Для каждого IP-адреса в диапазоне выполняется проверка доступности через протоколы HTTP и HTTPS.
   2. Если IP-адрес доступен, производится сканирование указанных портов.
5. **Вывод результатов сканирования:**
   1. Отображаются открытые порты для каждого IP-адреса.
   2. Для открытых портов выводится информация о сервисах, используя данные из файла "port\_info.txt".
6. **Оценка времени сканирования:**
   1. Пользователю выводится информация о предполагаемом времени сканирования, учитывая количество IP-адресов и портов.
7. **Отображение информации о хостах:**
   1. Пользователь может запросить дополнительную информацию о хостах, такую как Host Name, географическое расположение, провайдер и т.д.
8. **Анализ результатов:**
   1. Пользователь может анализировать результаты сканирования с использованием фильтров и сортировки, заданных через графический интерфейс.
9. **Вывод статистики:**
   1. Показывается общая статистика, такая как общее количество хостов, общее количество открытых портов, среднее и максимальное количество открытых портов.

**Описание программы:**

|  |  |  |
| --- | --- | --- |
| **Имя функции** | **Тип возвращаемого значения** | **Описание функции** |
| scan\_ip\_http | bool | Проверяет доступность IP-адреса по протоколу HTTP и возвращает True в случае успешного соединения. |
| scan\_ip\_https | bool | Проверяет доступность IP-адреса по протоколу HTTPS и возвращает True в случае успешного соединения. |
| get\_host\_info | dict | Получает информацию о хосте, используя API ipinfo.io, и возвращает словарь с данными о хосте. |
| scan\_port | bool | Сканирует указанный порт на указанном IP-адресе и возвращает True для открытого порта. |
| load\_port\_info | dict | Загружает информацию о портах из файла "port\_info.txt" и возвращает словарь с данными о портах. |
| determine\_services | dict | Определяет сервисы, работающие на открытых портах, используя информацию из файла "port\_info.txt". |
| scan\_ip\_range | None | Сканирует диапазон IP-адресов и отображает открытые порты и соответствующие сервисы. |
| estimated\_time | float | Рассчитывает предполагаемое время сканирования, учитывая количество IP-адресов и портов. |
| generate\_ip\_range | list | Генерирует список IP-адресов в заданном диапазоне. |
| start\_scan | None | Запускает процесс сканирования, отображает результаты и предполагаемое время сканирования. |
| show\_host\_info | None | Отображает дополнительную информацию о хостах, такую как Host Name, географическое расположение и др. |
| analyse\_results | None | Анализирует результаты сканирования с применением фильтров и сортировки, выводит статистику. |

**Рекомендации пользователю:**

1. **Запуск программы:**
   1. Для запуска программы убедитесь, что у вас установлен Python. Если нет, установите его с официального сайта Python.
   2. Запустите программу, выполнив скрипт с помощью команды python script\_name.py в командной строке или терминале.
2. **Интерфейс вкладок:**
   1. Программа имеет три вкладки: "Scan" (Сканирование), "Host Info" (Информация о хосте) и "Analyse" (Анализ). Переключайтесь между ними для выполнения различных задач.
3. **Сканирование IP-адресов и портов:**
   1. В разделе "Scan" введите диапазоны IP-адресов и портов, затем нажмите кнопку "Start Scan". Результаты будут отображены в текстовом поле.
4. **Просмотр информации о хосте:**
   1. В разделе "Host Info" введите диапазоны IP-адресов и нажмите кнопку "Show Host Info". Информация о хосте будет отображена в текстовом поле.
5. **Анализ результатов:**
   1. В разделе "Analyse" выберите параметры фильтрации и сортировки, затем нажмите кнопку "Analyse". Статистика и дополнительные детали будут отображены в текстовом поле.

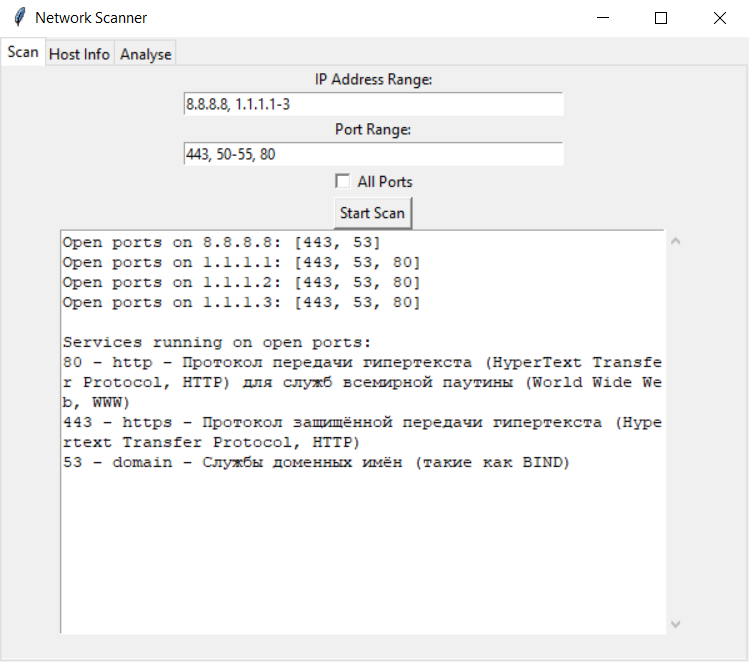
**Рекомендации для разработчика:**

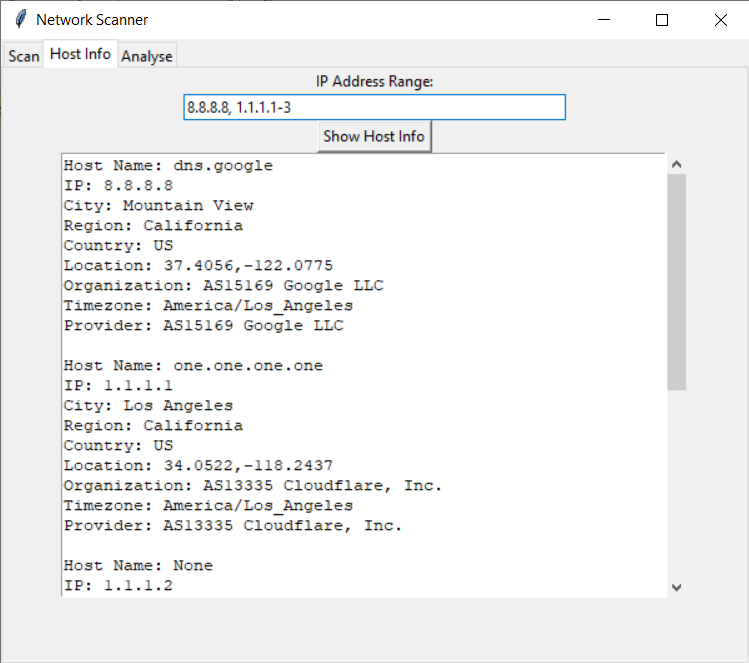
1. **Модификация портов и информации:**
   1. Изменяйте файл "port\_info.txt" для обновления информации о портах. Убедитесь, что формат остается тем же: <порт> <протокол> <описание>.

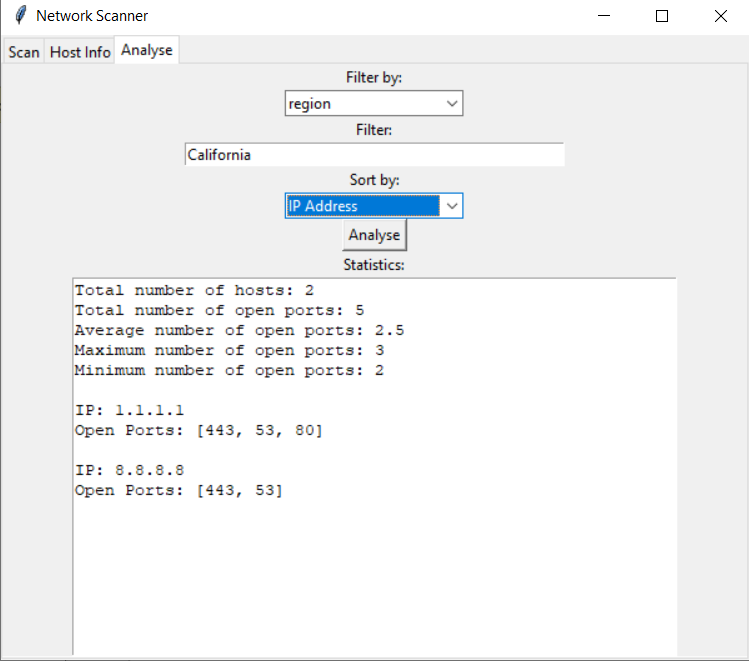
**Запуск программы:**

* Для запуска программы вам потребуется интерпретатор Python версии 3.x.
* Для запуска необходимо иметь библиотеки tkinter, requests, socket

**Контрольный пример:**







**Вывод:**

Реализован инструмент для сканирования и анализа сетевой активности.

**Список литературы:**

1. <https://ipinfo.io/>
2. [Документация по библиотеке socket](https://docs.python.org/3/library/socket.html)
3. [Документация по библиотеке requests](https://docs.python-requests.org/en/latest/)
4. [Документация по библиотеке tkinter](https://docs.python.org/3/library/tkinter.html)

**GitHub:**

https://github.com/StephanOlizko/Network-scanner