## Лекция 2.7. Автоматизация поиска чувствительных данных на Python

В ходе лекции будут рассмотрены следующие вопросы:  
1. Понятие конфиденциальности и её значение для бизнеса.  
2. Способы защиты информации: технические и организационные.  
3. Оценка рисков и управление ими.  
4. Как производится поиск чувствительных данных.

### Глава 1. Автоматизация поиска чувствительных данных на Python

Конфиденциальность — это защита информации от несанкционированного доступа, использования и распространения. В современной бизнес-среде, информация является одним из самых важных ресурсов, и её конфиденциальность играет критически важную роль в обеспечении целостности бизнеса.  
Для того чтобы защитить информацию, приходится использовать множество способов:  
1. Технический способ заключается в использовании различных средств связи, таких как пароли, шифрование, сложные протоколы связи и т. д.  
2. Организационный подход является более важным для правильной защиты информации, он включает в себя управление доступом к информации, внутреннюю политику безопасности, документационное обеспечение, обучение сотрудников и т. д.  
Процесс поиска чувствительных данных должен включать в себя оценку рисков и управление ими. Это подразумевает анализ возможных потенциальных угроз, определение уязвимых мест и разработку мер защиты.  
Процесс поиска чувствительных данных включает в себя три основных этапа:  
1. Определение чувствительных данных. Этот этап предполагает определение того, что именно считается чувствительной информацией в рамках вашей компании. Это может быть персональная информация, финансовые данные, бизнес-планы, интеллектуальная собственность и другие категории информации.  
2. Определение уязвимых мест. На этом этапе аудитор должен определить, какие уязвимые места и возможности злоумышленник может использовать для доступа к чувствительным данным. Это может быть несколько категорий угроз, например: угроза персонала, угроза через сеть инфраструктуры, угрозы со стороны поставщиков услуг и т. д.  
3. Выбор мер защиты. Этот этап подразумевает разработку соответствующих мер защиты, которые помогут предотвратить доступ злоумышленников к чувствительным данным, если меры недостаточны, то могут произойти утечки данных и возникнуть прочие проблемы.

### Глава 2. Автоматизация поиска чувствительных данных на Python

Автоматизация поиска чувствительных данных является важным этапом в обеспечении безопасности данных. Для автоматизации этого процесса можно использовать различные инструменты, в том числе и на Python.  
Одним из таких инструментов является библиотека Python под названием «regex». Она предоставляет функциональность для работы с регулярными выражениями, которые могут быть использованы для поиска различных типов чувствительных данных, таких как номера кредитных карт, номера социального страхования, номера паспортов и другие.  
Существует множество других инструментов для автоматизации поиска чувствительных данных, которые могут быть использованы в целях обеспечения безопасности данных.  
Некоторые из таких инструментов:  
1. Инструменты обработки сетевых пакетов, такие как Scapy.  
2. Инструменты анализа логов, такие как ELK stack.  
3. Инструменты сканирования веб-приложений, такие как OWASP ZAP.  
4. Инструменты сканирования файловых систем, такие как TruffleHog.  
Это лишь несколько примеров инструментов для автоматизации поиска чувствительных данных, которые могут быть использованы в различных сферах. В каждом случае выбор инструмента зависит от конкретных потребностей и требований безопасности данных.

Для примера, рассмотрим следующий код на Python, который ищет номера кредитных карт в текстовых файлах в заданной директории:

import re

import os

# задаем путь к директории, в которой будем искать файлы

dir\_path = "path/to/directory"

# задаем регулярное выражение для поиска номеров кредитных карт

regex\_pattern = r"\b\d{4}[- ]?\d{4}[- ]?\d{4}[- ]?\d{4}\b"

# проходим по всем файлам в заданной директории

for filename in os.listdir(dir\_path):

# проверяем, что файл является текстовым

if filename.endswith(".txt"):

# открываем файл на чтение

with open(os.path.join(dir\_path, filename), "r") as f:

# читаем содержимое файла в переменную

content = f.read()

# ищем все номера кредитных карт в содержимом файла

matches = re.findall(regex\_pattern, content)

# выводим найденные номера кредитных карт

print(f"Credit card numbers found in {filename}: {', '.join(matches)}")

1. Какие данные не относятся к чувствительным данным?
   1. Фактологические данные
   2. Государственные и классифицированные данные
   3. Корпоративные данные
   4. Финансовые данные
   5. Медицинская информация
   6. Интеллектуальная собственность
2. Выберите все способы, которые относятся к техническим мерам защиты:
   1. Установка механизмов аутентификации
   2. Разработку политики безопасности
   3. Установку брандмауэров
   4. Авторизация
   5. Резервное копирование
   6. Шифрование данных
3. Выберите все способы, которые относятся к организационным мерам защиты:
   1. Разработка политики безопасности
   2. Резервное копирование
   3. Обучение сотрудников
   4. Аудит и мониторинг
   5. Установление правил доступа и контроля
4. Какой процесс включает идентификацию уязвимостей, определение вероятности возникновения инцидентов и их потенциальные последствия?
   1. Оба ответа верны
   2. Управление рисками
   3. Оценка рисков
5. Какой процесс включает принятие мер для снижения рисков и реагирования на возможные инциденты?
   1. Оба ответа верны
   2. Управление рисками
   3. Оценка рисков
6. Выберите все уязвимости, которые включает в себя оценка рисков:
   1. Недостаточная защита паролей
   2. Политика сложных паролей
   3. Несекьюрные соединения
   4. Несанкционированный физический доступ
7. В рамках какого метода поиска чувствительных данных происходит поиск по определённому формату номеров телефонов или адресов электронной почты?
   1. Сканирование файлов и систем
   2. Применение регулярных выражений
   3. Анализ и мониторинг сетевого трафика
   4. Физический инспект