УДК 004.16

С. О. Маценко
О. П. Войтович

ЗАХИСТ ВЕБ-РЕСУРСІВ ВІД НЕСАНКЦІАНОВАНОГО КОПІЮВАНЯ

Вінницький національний технічний університет

Анотація
Безумовно, хто володіє інформацією - володіє світом. Але що робити, якщо обсяг інформації настільки великий, що потребує занадто багато зусиль для збору та аналізу? З цією проблемою допомагає впоратися парсинг - процес збору інформації з сайтів. Хоча сам парсинг існує вже досить давно і активно застосовується, його правомірність все ще не визначена (у всякому разі, на території України і країн ближнього зарубіжжя). Впоратися з не правомірним використанням парсингу допоможе технологія захисту веб-ресурсів основана на використанні чорних списків і підміні інформації
Ключові слова: парсинг, чорні списки, веб-ресурс, несанкціоване копіювання.

Abstract
Of course, who owns the information - owns the world. But what if the amount of information is so large that it takes too much effort to collect and analyze? Parsing, the process of gathering information from sites, helps to deal with this problem. Although parsing itself has existed for a long time and is actively used, its legitimacy has not yet been determined (at least in Ukraine and CIS countries). Web resource protection technology based on the use of blacklists and substitution of information will help to cope with the misuse of parsing.
Keywords: parsing, blacklists, web resource, unauthorized copying.

Вступ

Якщо максимально спростити, парсинг - це процес збору даних зі сторінок сайтів. Він проходить в три етапи:

1. Програмі-парсеру задається, що (які дані) і звідки (посилання на сайт) потрібно обробити.
2. Пошук даних
3. Збереження знайдених даних

Існує безліч сервісів, які збирають бази даних користувачів. Наприклад, збір передплатників певної спільноти в соцмережі. Почнемо з того, що майже вся інформація про фізичну особу є конфіденційною і може бути використана тільки за її згодою. Користувач надає таку згоду, коли реєструється на сайті. Сайт стає розпорядником персональної інформації. Такою інформацією є відомості чи сукупність відомостей, за якими особа може бути ідентифікована. Частковий виняток становить інформація про держслужбовців та інших публічних осіб (музикантів, акторів, спортсменів).

Розпорядник персональної інформації також повинен надати згоду на використання персональних даних. Якщо коротко, щоб парсинг був законним, парсити потрібно або деперсоніфіковані дані, або отримувати згоду розпорядника інформації.

Результати досліджень

В результаті використанні парсерів веб-розробник отримує сайт, вся інформація якого вже існує в мережі. Такий метод отримання контенту не схвалюється пошуковими системами, так як отримані тексти не є унікальними. Ранжування пошуковими системами сайтів, вся інформація яких отримана з допомогою парсерів, завжди буде дуже низькою. За плагіат інформації такий сайт навіть може бути вилученим з результатів пошуку.

Звичайно ж, парсери не читають тексту, вони всього лише порівнюють запропонований набір слів з тим, що виявили в інтернеті і діють за заданою програмою. Те, як пошуковий робот повинен працювати зі знайденим контентом, написано в командному рядку, що містить набір букв, слів, виразів і знаків програмного синтаксису. Такий командний рядок називається «регулярний вираз». Програмісти використовують жаргонні слова «маска» і «шаблон». Щоб парсер розумів регулярні вирази, він повинен бути написаний на мові, що підтримує їх в роботі з рядками. Така можливість є в PHP, Perl. Регулярні

вирази описуються синтаксисом Unix, який хоча і вважається застарілим, але широко застосовується завдяки властивості зворотної сумісності. Синтаксис Unix дозволяє регулювати активність парсинга, роблячи його «ледачим», «жадібним» і навіть «надтожадібним». Від цього параметру залежить довжина рядка, яку парсер копіює з веб-ресурсу. Надтожадібний парсинг отримує весь контент сторінки, її HTML-код і зовнішню таблицю CSS.

Зазвичай парсери написані на високорівневиї мовах програмування, однією із них є PHP. Її основні переваги:

- У нього є вбудована бібліотека libcurl, за допомогою якої скрипт підключається до будь-яких типів серверів, в тому числі які працюють по протоколах https (зашифроване з'єднання), ftp, telnet.
- PHP підтримує регулярні вирази, за допомогою яких парсер обробляє дані.
- У нього є бібліотека DOM для роботи з XML - розширюваним мовою розмітки тексту, на якому зазвичай подаються результати роботи парсера.
- Він відмінно ладнає з HTML, оскільки створювався для його автоматичної генерації.

Окрім крадіжки інформації парсери створюють велике навантаження на сервери, що може призвести до уповільнення роботи сайту, або взагалі його відключення. Одним із прикладів є американська компанія QVC. QVC (телевізійний рітейлер) подали в суд на Resultly (додаток-магазин) через те, що пошукові боти Resultly перевантажили сервери QVC з відключенням електроенергії, що призвело до збитків у 2 мільйони доларів. Суд виправдав Resultly, на тій підставі, що вони не мали наміру завдати шкоди. Проте, в схожих ситуаціях ризик понести відповідальність за порушення роботи сайту все ж залишається.

З метою обмеження доступу до сайту можна використовувати капчу. Головним недоліком даного способу вважалися незручності, які створювалися для реальних користувачів. Наприклад, була необхідність введення тексту з картинки або розгадування графіки. Google придумав як вирішити проблему з такими незручностями. Рішення полягає в аналізі даних про активність користувачів. Якщо володіти даними про трафік, то перевірку можна проводити у фоновому режимі, без участі користувача. На практиці через Google reCAPTCHA API можна отримати дані про те, чи є клієнт роботом чи ні. По кожному з клієнтів Google передає числове значення за шкалою від 0.1 до 0.9. Приклади значень:

- 0.9. Означає, що клієнт з високою ймовірністю є користувачем. При таких значеннях ніяких додаткових перевірок і обмежень на використання сайту створювати не слід;
- 0.3. Означає, що клієнт швидше користувач, ніж робот. Для клієнтів з такими значеннями має сенс іноді проводити додаткову перевірку. Наприклад, перевірку можна проводити під час підозрілої активності або при піковому навантаженні на сервер сайту;
- 0.1 Означає, що клієнт швидше робот, ніж користувач.

Ще один спосіб захисту від парсингу сайту - Пастка для ботів. Як превентивний захід щодо захисту від парсинга сайту слід використовувати пастки для ботів, так звані honeypot. Суть даного методу полягає в створенні приманки для ботів, що згодом дозволяє зібрати список роботів, вивчити стратегію зловмисників і визначити перелік засобів, за допомогою яких можуть бути нанесені удари по серверам сайту. На практиці спосіб полягає в тому, що на сайті розміщується посилання, по якій не будуть переходити користувачі, але будуть переходити боти. Наприклад, в якості такого посилання може бути прозора картинка розміром 1 на 1 піксель.

Як альтернативу попереднім методам, також можна аналізувати властивості IP-адреси

Одним із методів боротьби є створення чорних списків і автоматизація процесу додаавання підозрілих користувачі у даний список. Як було вищеописано, парсер – це програмний застосунок який виконується з певним проміжком часу. Відслідкувавши інтервали між запитами які відбуваються з одного і того самого IP адреса, можна легко відслідкувати зловмисника і добавити його в чорний список.

Метод з використання чорних спискв дієвий, проте не надто результативний. Визначивши що зловмисника заблокували, він може з легкістю змінити свій IP адрес, або використати PROXY-сервер, продовживши викрадати інформацію. Даний метод можна вдосконалити, не блокуючи користувачів які находяться в чорному списку, а замінюючи їм інформацію на не правдиву. Цим самим зловмисник не зрозуміє що його обвели навколо пальця, а компанія збереже дорогоцінні дані.

Для реалізації вищеописаного методу слід використовувати високорівневу мову програмування яка широко використовується в сучасному світі. Такою мовою є PHP, сімдесят відсотків сайтів сучасного інтернету написані саме на даній мові. Оскільки в чистому вигляді дану мову майже ніхто не використовує для актуальності даного модулю на довгий період часу, розумно буде використати сучасний

фреймворк – Laravel. Із основних його переваг: суворе використання єдиного стилю написання коду, що дасть змогу вдосконалювати модуль не лише його розробникам, наявність вбудованого модулю Eloquent ORM для роботи з базами даних, що дасть змогу легко вести облік, зберігати дані користувача для подальшого аналізу, а також дотримання принципів ООР, що не мало важливо для написання якісного коду.

Дотримуючись принципів MVC (model, view, controller) реалізувати дану систему захисту буде не досить важко. Model – відповідає за зв'язок сайту з його не видимою стороною – базою даних, це дасть нам змогу, зберігати всю важливу для нас інформацію. View – відповідає за відображення інформації на сайті, це так звані шаблони. Controller – мозок даної конструкції, саме в контролері будуть відбуватися всі необхідні процедури. Контролер отримає інформацію про IP користувача, передасть її в Model для збереження, перевірить наявність користувача в чорному списку, і при виявленні бота, в момент передачі даних у View зробить їх підміну, в результаті чого зловмисник отримає фальшиву інформацію.

Основні етапи створення модуля для захисту веб-інформації:

1. Інтеграція на Веб-ресурс реєстру відвідування, з детальною інформацією про користувача і періодичність відвідування ним сайту
2. Створення рейтингу користувачів і чорного списку.
3. Аналіз отриманих даних і підміна, інформації для виводу, на основі отриманих результатів

Висновки

Згідно з проаналізованими дослідженнями встановлено, що захист веб-ресурсів від копіювання може зекономити компанію велику суму грошей, запобігти надмірному навантаженню на сайт і зберегти унікальність контенту.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Що таке парсер − [Електронний ресурс]. – Режим доступу: https://www.ua5.org/web/397-shho-take-parser.html. – Назва з екрану.
2. Парсинг: від теорії до судової практики − [Електронний ресурс]. – Режим доступу: https://jurliga.ligazakon.net/ua/analitycs/194822_parsing-vd-teor-do-sudovo-praktiki. – Назва з екрану.
3. Парсинг — а это вообще легально и законно?− [Електронний ресурс]. – Режим доступу: https://xmldatafeed.com/parsing-a-jeto-voobshhe-legalno-i-zakonno/. – Назва з екрану.
4. Парсинг. Что это и где используется [Електронний ресурс]. – Режим доступу: https://ipipe.ru/info/parsing/. – Назва з екрану.
5. What is Parsing? [Електронний ресурс]. – Режим доступу: https://medium.com/the-mighty-programmer/what-is-parsing-4012f997d265 – Назва з екрану.
6.

Маценко Сергій Олегович – студент групи 2БС-17б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, email: matsenko2502@gmail.com

Войтович Олеся Петрівна — к.т.н, доцент, доцент кафедри комп'ютерних наук, Вінницький національний технічний університет, м. Вінниця