УДК 004.021
DOI: 10.31673/2409-7292.2022.013843

Цуранов М.О.

ЗАХИСТ WEB-ДОДАТКУ ВІД SQL-INJECTION ЗА ДОПОМОГОЮ ОБЛАДНАННЯ FORTINET

У цій статті проведено аналіз щодо захисту web-додатку за допомогою програмно-технічного засобу Fortigate 60E, а також необхідності використання систем захисту web-додатків. Проаналізовано види sql-injection, їх вплив на роботу web-додатку. Визначені основні функції та можливості обладнання Fortigate 60E. Сформовані рекомендації щодо використання обладнання подібного Fortigate.

Ключові слова: Fortigate, web-додаток, захист, sql-injection.

Вступ

За останні двадцать років цифровий світ змінився дуже сильно. Сьогодні можно отримати майже будь-яку послугу онлайн не виходячи з дому, чи оформлення документів, чи придбання квитків тощо. Усі ці послуги побудовані та надаються за допомогою web-технологій. Надання послуг приносить доволі багато грошей, а де гроші там і зловмисники які бажають їх отримати. Технології розвиваючись ускладнюються і все складніше стає контролювати подібні системи. На сьогодні вразливості web-додатків перевершують по можливому впливу усілякі інші проблеми інформаційної безпеки. Нажаль, більшість зовнішніх атак саме на корпоративні інформаційні системи націлені на web-додатки. Йдучи за трендами сучасного світу все більше компаній переходять в онлайн. А поширення онлайн-платежів тільки підсилює цей тренд. Як показує досвід, там де є онлайн продажі, використання спеціалізованого обладнання для захисту є по-справжньому критичним та необхідним.

Основна частина

WEB-додаток та найпоширеніші вектори атак

Web-додаток - це різновид комп'ютерної програми. Він використовує онлайн-технології (включаючи браузери) для виконання величезного кола різноманітних завдань. Багато додатків використовуються для цілей онлайн-роздрібної торгівлі. Однак вони можуть служити для різних цілей, від замовлення їжі на винос до бронювання свят. Крім того, web-додаток може бути таким простим, як контактні форми web-сайту або онлайн-калькулятори. Web-програми отримують і зберігають інформацію за допомогою сценаріїв на стороні сервера (на таких мовах, як PHP і ASP), тоді як сценарії на стороні клієнта (в JavaScript і HTML5) представляють відповідну інформацію в інтерфейсі користувача. Ця інформація може мати будь-яку кількість форм. Поширені типи веб-додатків включають кошики для покупок, системи керування вмістом та онлайн-форми. Оскільки вони настільки універсальні, web-програми дозволяють людям виконувати різноманітні функції. Для споживачів це включає розміщення замовлень, створення списків бажань та запити про продукти чи послуги через веб-сторінки.

Додатки також дозволяють співробітникам обмінюватися документами, спілкуватися один з одним, редагувати файли та спільно працювати над спільними проектами. У нову епоху дистанційної роботи це надзвичайно важливо. Ми часто маємо тенденцію поєднувати веб-додатки з мобільними програмами для електронної комерції, але в першому є набагато більше, ніж в другому. Текстові процесори, програми для роботи з електронними таблицями та інше таке програмне забезпечення можна вважати web-додатками. Вони також можуть включати системи управління взаємовідносинами з клієнтами (CRM) і управління контентом [1].

Відповідно до звіту Itbrief [2], лише за 2021 рік кількість атак на web-додатки збільшилась на 88% порівняно до 2020 року. Спеціалісти прогнозують, що до 2025 року втрати від кібершихраств може сягнути 10.5 трильонів долларів щорічно [3]. Навіть невелики ризики можуть коштувати доволі багато грошей бізнесу і ще добре якщо тільки грошими.

Репутація та гарне ім'я сьогодні також є цінними активами для бізнесу у сфері надання послуг. Тому важливість захисту виходить на першу роль і стає першочерговим питанням.

Основні види атак на web-додатки та їх короткий опис

1. Cross-Site Scripting (XSS)

Атака XSS є найпоширенішою кібератакою, що становить приблизно 40% усіх атак. Незважаючи на те, що це найчастіша атака, більшість з цих атак не дуже складні і виконуються кіберзлочинцями-любителями за допомогою скриптів, створених іншими.

Міжсайтові сценарії націлені на користувачів сайту, а не на саму веб-програму. Зловмисник вставляє фрагмент коду у вразливий веб-сайт, який потім виконує відвідувач веб-сайту. Код може скомпрометувати облікові записи користувачів, активувати троянських коней або змінити вміст веб-сайту, щоб обманом змусити користувача надати особисту інформацію.

2. Injection Attacks

Метод SQL-ін'єкції є найпопулярнішою практикою, яку використовують кіберзлочинці в цій категорії. Методи ін'єкційної атаки спрямовані безпосередньо на веб-сайт і базу даних сервера. Використовуючи SQL Injection, зловмисник може обійти автентифікацію, отримати доступ, змінити та видалити дані в базі даних. Під час виконання зловмисник вставляє фрагмент коду, який розкриває приховані дані та введені користувачем, дозволяє змінювати дані та загалом ставить під загрозу web-додаток.

3. Fuzzing (або нечітке тестування)

Розробники використовують fuzz-тестування, щоб знайти помилки кодування та лазівки в безпеці програмного забезпечення, операційних систем або мереж. Однак зловмисники можуть використовувати ту саму техніку, щоб знайти вразливі місця на вашому сайті або сервері.

Він працює шляхом початкового введення великої кількості випадкових даних (fuzz) у програму, щоб змусити її зламатися. Наступним кроком є використання програмного інструменту fuzzer для визначення слабких місць. Якщо є якісь лазівки в безпеці, зловмисник може скористатися цим.

4. Path Traversal

Атаки обходу шляху спрямовані на кореневу веб-теку для доступу до неавторизованих файлів або каталогів за межами цільової папки. Зловмисник намагається ввести шаблони переміщення в каталог сервера, щоб рухатися вгору в ієрархії. Успішний обхід шляху може поставити під загрозу доступ до сайту, файли конфігурації, бази даних та інші веб-сайти та файли на тому ж фізичному сервері.

5. Distributed Denial-of-Service (DDoS)

Одна лише DDoS-атака не дозволяє зловмисникові порушити безпеку, але тимчасово або назавжди переведе сайт в автономний режим. DDoS-атака має на меті перевантажити веб-сервер цілі запитами, зробивши сайт недоступним для інших відвідувачів. Ботнет зазвичай створює величезну кількість запитів, які розподіляються між раніше зараженими комп'ютерами. Також DDoS-атаки часто використовуються разом з іншими методами. Мета першого – відвернути увагу систем безпеки, а наступного - використовуючи вразливість зламати web-додаток.

Різновид SQL-injection

SQL-injection можно поділити на три основні категорії: In-band SQLi, Inferential SQLi and Out-of-band SQLi:

1. In-band SQLi (класичний SQLi)

In-band ін'єкція SQL є найпоширенішою та легкою у використанні атак із застосуванням SQL. In-band ін'єкція SQL відбувається, коли зловмисник може використовувати один і той самий канал зв'язку для запуску атаки та збору результатів. Двома найпоширенішими типами In-band ін'єкції SQL є SQLi на основі помилок і SQLi на основі об'єднання:

SQLi на основі помилок — це метод In-band впровадження SQL, який покладається на повідомлення про помилки, що надсилаються сервером бази даних, щоб отримати інформацію про структуру бази даних. У деяких випадках для того, щоб зловмисник перерахував всю базу даних, достатньо лише ін'єкції SQL на основі помилок. Хоча помилки дуже корисні на етапі розробки веб-програми, їх слід вимкнути на реальному сайті або зареєструвати у файлі з обмеженим доступом;

SQLi на основі Union — це техніка In-band впровадження SQL, яка використовує оператор UNION SQL для об'єднання результатів двох або більше операторів SELECT в один результат, який потім повертається як частина відповіді HTTP.

2. Inferential SQLi (сліпа SQLi)

Inferential SQLi на відміну від внутрішньосмугового SQLi, може зайняти більше часу для зловмисника, однак він настільки ж небезпечний, як і будь-яка інша форма SQL Injection. Під час атаки SQLi з висновком фактично ніякі дані не передаються через веб-додаток, і зловмисник не зможе побачити результат атаки в діапазоні (тому такі атаки зазвичай називають «сліпими атаками SQL Injection»). . Натомість зловмисник може відновити структуру бази даних, надсилаючи корисні дані, спостерігаючи за реакцією веб-додатка та результатом поведінки сервера бази даних.

Двома типами ін'єкції SQL є SQLi на Blind-boolean-based SQLi та Blind-time-based SQLi:

Boolean-based SQL Injection — це метод ін'єкції SQL, який спирається на відправку SQL-запиту до бази даних, що змушує програму повертати інший результат залежно від того, чи повертає запит результат TRUE чи FALSE. Залежно від результату вміст відповіді HTTP зміниться або залишиться незмінним. Це дозволяє зловмиснику зробити висновок, чи повернуло використане корисне навантаження true чи false, навіть якщо дані з бази даних не повертаються. Ця атака зазвичай повільна (особливо на великих базах даних), оскільки зловмиснику потрібно буде перерахувати базу даних, символ за символом;

Time-based SQL — це методика ін'єкції SQL, яка ґрунтується на відправленні запиту SQL до бази даних, що змушує базу даних чекати певну кількість часу (у секундах), перш ніж відповісти. Час відповіді вкаже зловмиснику, чи є результат запиту TRUE чи FALSE. Залежно від результату відповідь HTTP буде повернуто із затримкою або негайно. Це дозволяє зловмиснику зробити висновок, чи повернуло використане корисне навантаження true чи false, навіть якщо дані з бази даних не повертаються.

3. Out-of-band SQLi

Out-of-band SQL Injection не дуже поширений, головним чином тому, що він залежить від функцій, увімкнених на сервері бази даних, який використовується веб-додатком. Out-of-band ін'єкція SQL відбувається, коли зловмисник не може використовувати той самий канал для запуску атаки та збору результатів. Out-of-band методи пропонують зловмиснику альтернативу методам висновку, заснованому на часі, особливо якщо відповіді сервера не дуже стабільні (що робить атаку на основі часу висновку ненадійною). Методи Out-of-band SQLi будуть покладатися на здатність сервера бази даних робити запити DNS або HTTP для доставки даних зловмиснику [4].

SQL ін'єкції є доволі небезпечними. Неправильна конфігурація серверу бази даних або неправильна фільтрація запитів може призвести до отримання паролю адміністратора або витоку даних користувачів. Саме тому важливість захисту web-додатку від SQL ін'єкцій неможна перебільшити.

Функції та можливості обладнання Fortigate

FortiGate, брандмауер нового покоління від лідера IT-лідера кібербезпеки Fortinet, забезпечує найвищий захист від загроз для бізнесу будь-якого розміру. Використовуючи спеціально створені процесори безпеки та аналіз загроз від FortiGuard, брандмауер FortiGate забезпечує неперевершену продуктивність і захист мережі. Брандмауер FortiGate працює, досліджуючи дані, які надходять у мережу, і перевіряючи, чи безпечно їх передавати до пункту призначення. Fortigate надає можливість відображати потрібні профілі безпеки для

зручності адміністрування та швидкого знаходження необхідних налаштувань. Перелік профілів якими Fortigate дає змогу керувати: Antivirus; Web Filter; Application Control; Intrusion Prevention; File Filter; VoIP; SSH\SSL Inspection.

Профіль Intrusion Prevention – це профіль запобігненю вторгнень. Саме для забезпечення захисту мережі від проникнення створений профіль Intrusion Prevention. Профіль виявляє аномалії та експлоїти. Різниця між двома термінами полягає в тому, що експлоїт це підтверджена атака яка може бути задетектована антивірусом або фаерволом веб-додатків. Аномалія ж в свою чергу – це незвична поведінка в мережі яка не співпадає з сигнатурами звичайного трафіку робочих додатків працівників. Intrusion Prevention System(IPS) також використовує сигнатури для виявлення відомих атак. Також в своєму арсеналі цей профіль має декодер протоколів. Декодер розбирає протоколи відповідно до специфікацій та визначає чи правильно сформований пакет і т.д.

Для симуляції атаки була розгорнута віртуальна машина OWASP Broken Web Applications за програмно-апаратним комплексом Fortigate. Схема підключення наступна (рис.1):

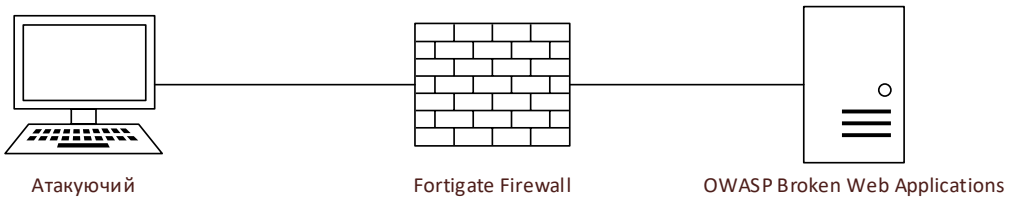

Рис. 1 Схема підключення атакуючого до OWASP Web Applications

Налаштування профілю IPS на Fortigate (рис. 2–3):

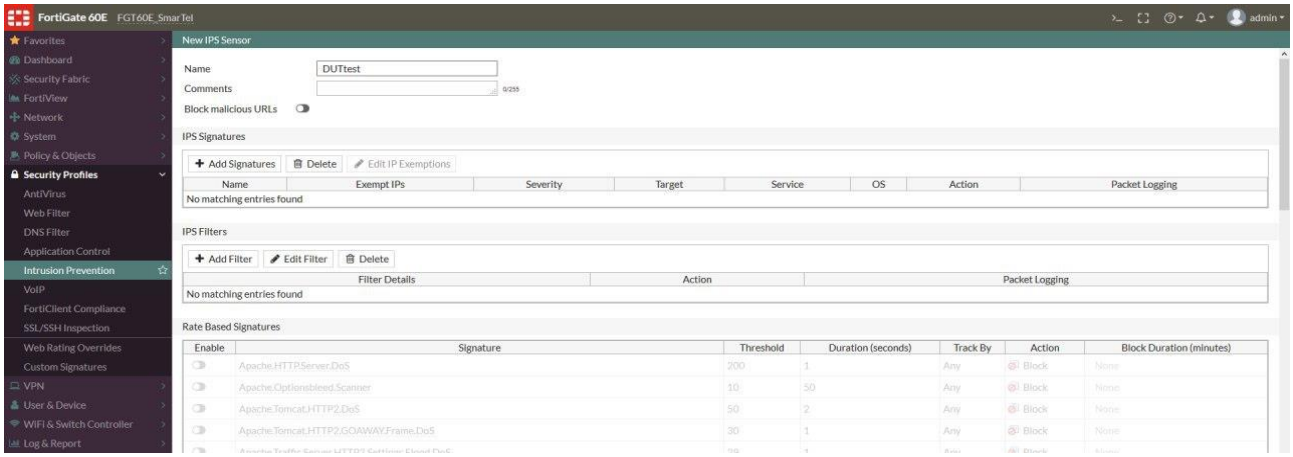

Рис. 2. Налаштування профілю IPS

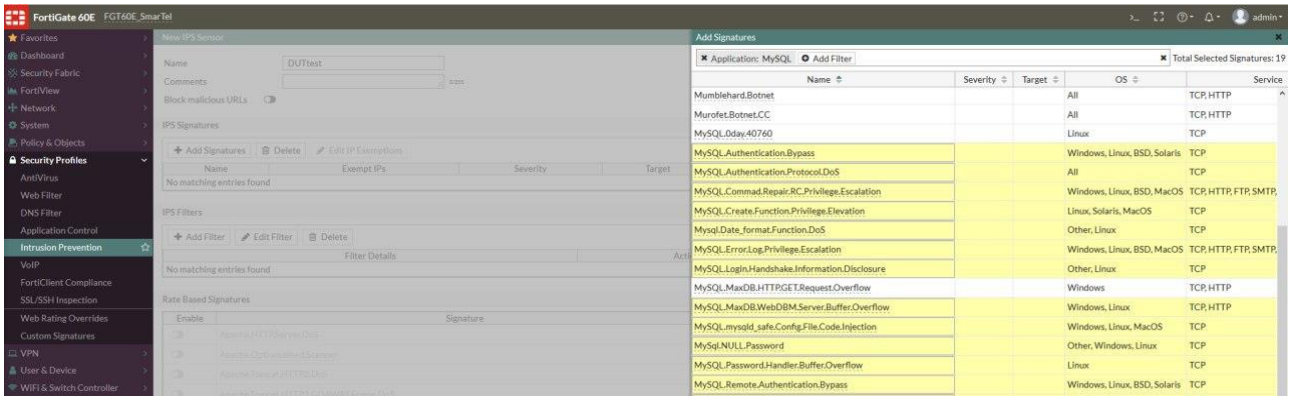

Рис. 3 Додавання сигнатур SQL ін'єкцій до профілю IPS

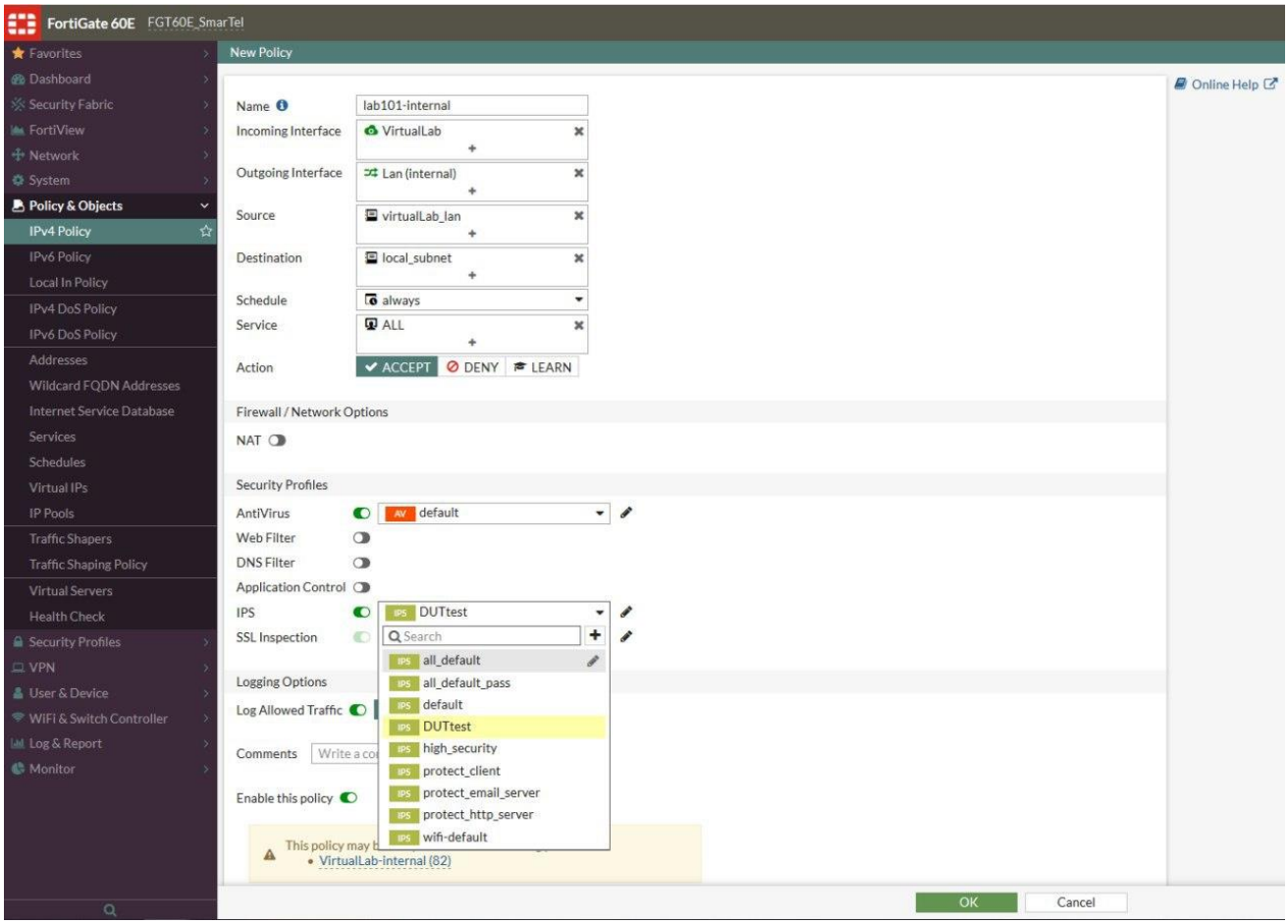

Рис. 4. Додавання тестового профілю IPS до політики доступу до серверу Web-додатку

Далі потрібно запустити SQLmap для тестування (рис.5).

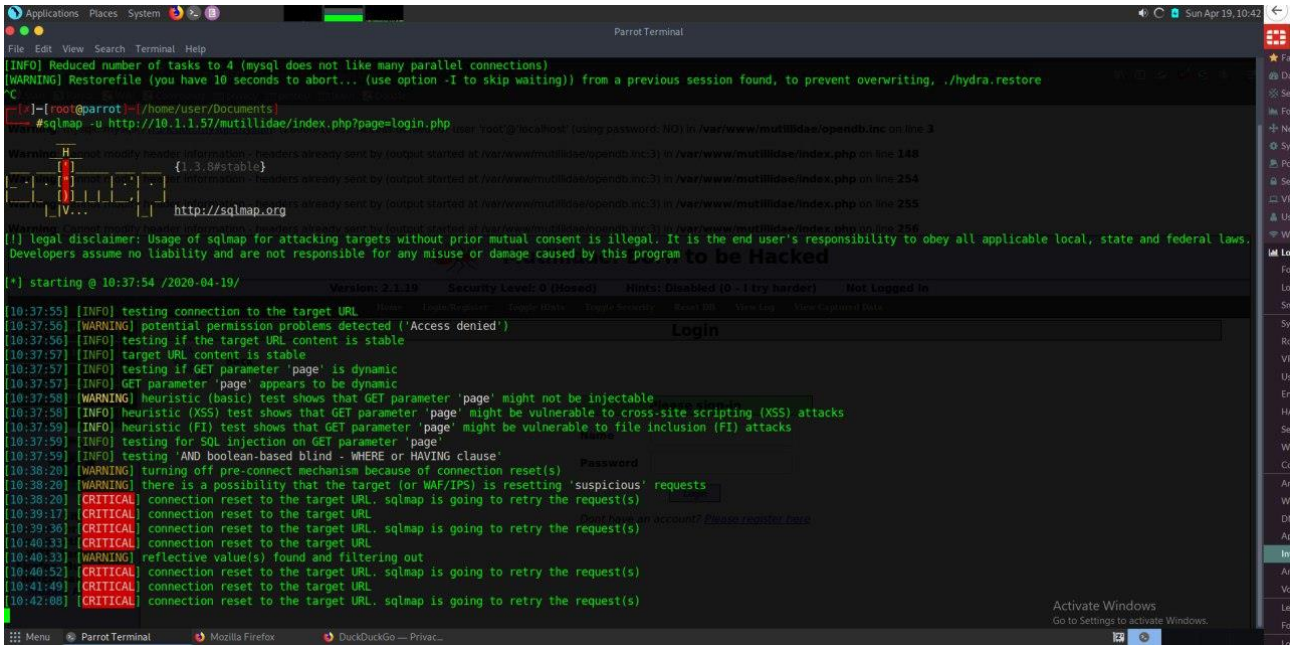

Рис. 5. Приклад виводу інструменту SQLmap для web-додатку

Результати роботи програмно-апаратного комплексу Fortigate (рис.6):

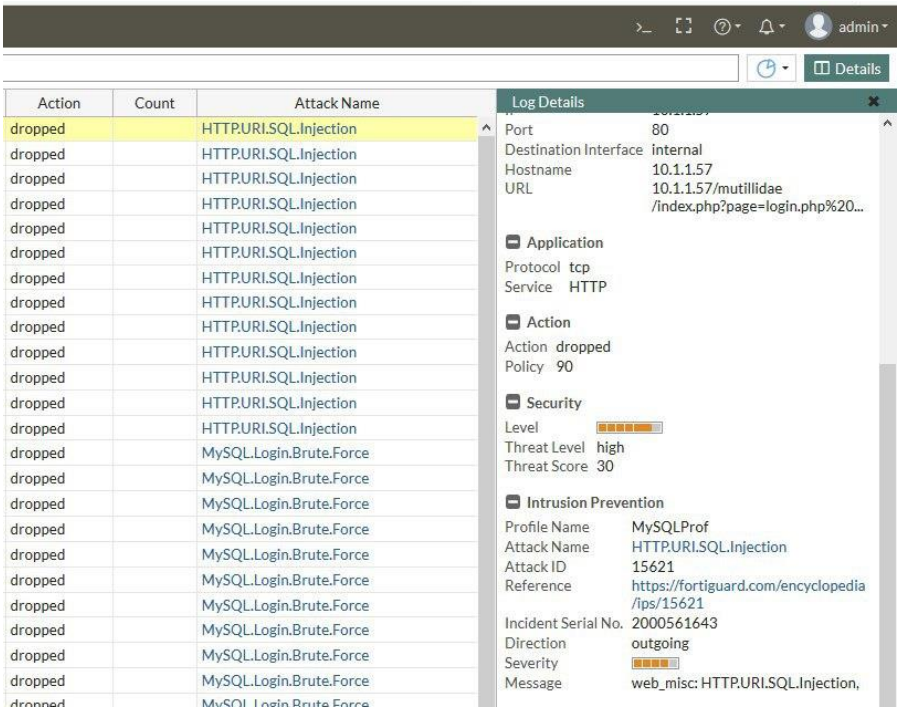

Рис. 6. Журнал записів Fortigate для профілю IPS

Як можно побачити на малюнку (рис.6), профіль IPS виконав своє завдання та не пропустив SQL ін'єкції. Це можно зрозуміти з колонки Action, що означає дії які були виконані над пакетами які потрапили на Fortigate. Тип дії dropped означає, що пакети були відкинути брендмауером и не потрапили на пункт призначення, тобто до OWASP Broken Web Application.

Висновки

Отже, SQL ін'єкція є досить небезпечною вразливістю web-додатків. Якщо не захищати такий вразливий web-додаток то можна отримати великі фінасові збитки через втрату конфеденційної інформації. Так як тенденція до атак на web-додатки тільки зростає, то необхідність захисту подібних вразливостей також збільшується.

В статті було розглянута важливість захисту web-додатків, основні атаки на web-додатки, а також найпоширініши SQL ін'єкції та їх типи. Проаналізовані функції та можливості апаратного-програмного комплексу Fortigate, а також його здатність захищати web-додатки за допомогою профілю IPS.

Перелік посилань

1. What is a web application? [Електронний ресурс] – Режим доступу до ресурсу: https://www.ringcentral.co.uk/gb/en/blog/definitions/web-application/
2. web application attacks climb 88% - report [Електронний ресурс] – Режим доступу до ресурсу: https://itbrief.com.au/story/malicious-web-application-attacks-climb-88-report
3. Cybercrime To Cost The World $10.5 Trillion Annually By 2025 [Електронний ресурс] – Режим доступу до ресурсу: https://cybersecurityventures.com/hackerpocalypse-cybercrime-report-2016/
4. Types of SQL Injection (SQLi) [Електронний ресурс] – Режим доступу до ресурсу: https://www.acunetix.com/websitesecurity/sql-injection2/
5. Sqlmap description [Електронний ресурс] – Режим доступу до ресурсу: https://sqlmap.org/

Надійшла: 10.01.2022 Рецензент: д.т.н., професор Савченко В.А.