УДК 004.932:[004.056:003.27]

ВИЯВЛЕННЯ ШКІДЛИВИХ СТЕГАНОГРАФІЧНИХ ІНСТРУКЦІЙ У ФАЙЛАХ ЗОБРАЖЕННЯ З ВИКОРИСТАННЯМ ШІ

Лісін О.А.
email: oleksandr.lisin@nure.ua
Науковий керівник – проф. каф. ШІ Бодянський Є. В.
Харківський національний університет радіоелектроніки, каф. ШІ
м. Харків, Україна

This work is a detailed explanation of steganography utilization as a mean of concealed cyberattacks. There is an overview of what steganography is in general as well as a description of how it is used for malicious goals. You will be explained about popular Command & Control tactic that uses steganography to transfer malicious instructions to infected machines via simple image files. As well as Least Significant Bit exploitation which is a foundation for said tactic. Finally, there is a suggestion for a possible solution in detecting such threats using artificial intelligence.

У сучасну епоху кіберзагроз, коли антивіруси, брандмауери та інші різноманітні системи ефективно блокують відомі віруси, зловмисники продовжують дедалі частіше залучати методи цифрової мімікрії. Стеганографія є одним із найнебезпечніших та витончених інструментів у цій справі.

Розглянемо використання стеганографії на прикладі атаки типу Command & Control (C2). Все починається зі звичайного зараження жертви, найчастіше, через фішинг, вивантажується невеликий скрипт, який сам по собі не класифікується антивірусами як шкідливий. Після цього він осідає в системі і чекає на подальші інструкції. Саме для їх передачі зловмисники використовують звичайні растрові зображення, вони служать прихованим каналом зв’язку для передачі команд вже зараженим машинам. Більш того, мережеві фільтри часто ігнорують зображення і не перевіряють їх так ретельно, як виконувані файли, і цим активно користуються для обходу захисту. Для цієї стратегії необхідне непідозріле місце зберігання зображень з інструкціями. Тримати власний сервер для цього – недоцільно, бо його легко заблокувати. Натомість зображення завантажуються на легітимні платформи типу Twitter чи imgur. І це має сенс, адже трафік до цих сервісів не викличе підозри ні у користувача, ні у корпоративного фаєрволу. Таким чином заражена машина за розкладом звертається до певної платформи, завантажує зображення, зчитує приховані дані, розшифровує команду і виконує її. Назовні ж це виглядає як звичайний HTTP-запит до сервісу.

Найпоширенішим методом приховування даних у зображеннях є LSB (Least Significant Bit). Кожен растровий піксель містить значення R, G, B (інколи ще Alpha) від 0 до 255. Значення молодшого біта пікселя

для кожного каналу несе мінімальний вплив на зміну кольору, що неможливо помітити на око. Саме в ці біти записується корисне навантаження. До прикладу:

Оригінальний піксель

$$R = 11001101; G = 10110010; B = 01001111$$

Після запису даних

$$R = 11001100; G = 10110011; B = 01001110$$

Таким чином в один піксель було записано 3 біти інформації: 0, 1, 0. У зображення 800x600 можна сховати близько 175 КБ даних, цього цілком достатньо для передачі команд або навіть невеликих модулів.

Саме для виявлення аномалій у LSB зображення майже ідеально підходять глибокі нейромережі. Як прототип для такої моделі можна взяти архітектуру розроблену для виявлення шкідливого коду у APK-файлах з використанням Трансформера. Згідно з цією архітектурою, для виявлення шкідливого ПЗ іноді застосовують підхід, коли бінарні файли або байтові послідовності перетворюють у зображення. Ці зображення потім аналізують за допомогою CNN, які добре підходять для вилучення просторових ознак із візуальних даних. Щоб мережа фокусувалась на найважливіших регіонах, додається механізм уваги. Це допомагає мережі ігнорувати менш важливі або шумові ділянки і концентруватися на характерних патернах, які є ознаками шкідливого ПЗ.

Подібну архітектуру можна використати для виявлення стеганографії. Для CNN заражена картинка виглядає як специфічний шум або порушення природної текстури. Згорткові шари ідеально підходять для того, щоб помітити ці мікро-зміни в локальних групах пікселів. Вони служать у якості фільтрів, що виділяють аномалії, які не характерні для звичайних фотографій. При цьому можна використовувати як стандартні двовимірні ядра, так і одновимірні (якщо представити картинку як лінійний потік байтів). Але зараження це не просто випадковий шум, а послідовність байтів, яка має певну структуру. Саме тут механізм уваги допомагає моделі виявити взаємозв’язки між аномаліями, виявленими CNN. Він збирає розрізнені частини прихованого коду в єдиний ланцюжок, дозволяючи моделі відрізнити звичайний цифровий шум від структурованого шкідливого коду.

Список використаних джерел:

1. Bodyanskiy Y. Computational intelligence techniques for data analysis. GI Digital Library: Startseite. URL: https://dl.gi.de/items/e7619119-325a-4708-a30b-15bdf1b1df13

2. Rahali A., Akhloufi M. A. MalBERT: using transformers for cybersecurity and malicious software detection. arXiv.org. URL: https://arxiv.org/abs/2103.03806