

ОЦІНКА ЗАГАЛЬНОГО РІВНЯ ЗАХИЩЕНОСТІ ПРОГРАМНОГО ПРОДУКТУ НА ОСНОВІ МЕТРИК ВРАЗЛИВОСТЕЙ

А. В. Крошкіна¹, М. В. Грайворонський¹

¹Національний технічний університет України «Київський політехнічний інститут»

Анотація

В роботі пропонується метод, який застосовує метрики вразливостей і розраховує оцінку загального рівня захищеності певного програмного продукту. Це дає змогу зацікавленим організаціям та установам обрати серед ряду подібних за функціональністю продуктів той, який є більш захищеним, і таким чином зменшити інформаційні ризики, пов'язані з використанням програмного забезпечення.

Ключові слова: вразливості програмного забезпечення, метрики вразливостей, CVSS

Вступ

В сьогоднішній час все більше організацій та державних установ значною мірою залежать від використання ними комп'ютерних систем, прикладних програм та мереж. В таких умовах зростає необхідність забезпечення безпеки інформації, що обробляється в таких системах.

Вразливості програмного забезпечення розцінюються як основна причина порушення політики безпеки інформації, яка призводить до крупномасштабних фінансових збитків.

З розвитком комп'ютерної індустрії та складності програмних продуктів зростає кількість вразливостей наявних у них. Модульна, по-компонентна розробка програм призводить до повторної появи вразливостей в нових версіях продукту або інших програмних продуктах цього розробника. Очевидно, що неможливо забезпечити створення продуктів без вразливостей, а тому вдосконалення заходів для прогнозування і попередження вразливостей є сучасною необхідністю.

Існує багато методів та метрик для оцінки суворості вразливостей програмного забезпечення (*software vulnerability severity*), проте вони розглядають лише окремі вразливості [?] та мають сенс лише для фахівців в цій галузі.

В роботі пропонується метод для оцінки загального рівня захищеності певного програмного продукту на основі застосування метрик вразливостей.

1. Метрика CVSS

Існує декілька різних методів оцінки та вимірювання суворості вразливостей програмного забезпечення. Найбільш розповсюдженою та прийнятною є метрика CVSS [?] версії 2.10, яка була розроблена NIST ¹. Метрики в системі CVSS поділені на групи:

- 1) Базові метрики (*Base metrics*) – вимірюють внутрішні, фундаментальні характеристики вразли-

вості, не змінні з часом та не залежні від середовища.

- 2) Тимчасові метрики (*Temporal metrics*) – вимірюють характеристики вразливості, змінні в часі, але не залежні від середовища.
- 3) Метрика середовища (*Environmental metrics*) – вимірюють характеристики вразливості, унікальні для конкретного середовища.

Є шість базових показників з відповідними балами, які фіксують найбільш фундаментальні риси вразливості [?], [?]:

- 1) Вектор доступу (*AV – Access Vector*) – показує яким чином експлуатується вразливість: локально (0, 365), локальна мережа (0, 646) або з мережі (1);
- 2) Складність доступу (*AC – Access Complexity*) – показує складність проведення атаки для експлуатації вразливості: висока складність (0, 35), середня (0, 61), мала (0, 71);
- 3) Автентифікація (*Au – Authentication*) – показує скільки разів зловмисник має бути автентифікованим для експлуатації вразливості: жодного разу (0, 704), один раз (0, 56), багато разів (0, 45);
- 4) Вплив на конфіденційність (*CI – Confidentiality Impact*) – показує вплив на конфіденційність інформації у разі успішної експлуатації: не впливає (0), частковий вплив (0, 275), повний вплив (0, 66);
- 5) Вплив на цілісність (*II – Integrity Impact*) – показує вплив на цілісність інформації у разі успішної експлуатації: не впливає (0), частковий вплив (0, 275), повний вплив (0, 66);
- 6) Вплив на доступність (*AI – Availability Impact*) – показує вплив на доступність інформації у разі успішної експлуатації: не впливає (0), частковий вплив (0, 275), повний вплив (0, 66);

Для оцінювання, спочатку розраховується базова оцінка S_{base} згідно з рівняння 1, при цьому отримується оцінка в межах від 0 до 10.

¹NIST – National Institute of Standard and Technology

За необхідністю базова оцінка може бути доповнена з урахуванням тимчасової метрики та(або) метрики середовища.

В межах роботи розглядаємо лише базові метрики.

$$S_{base} = round((0,6 \cdot Impact + 0,4 \cdot Exploitability - 1,5) \cdot f(Impact)) \quad (1)$$

де:

$round()$ - округлення до десятих

$Exploitability = 20 \cdot AV \cdot AC \cdot Au$

$Impact = 10,41 \cdot (1 - (1 - I_C) \cdot (1 - I_I) \cdot (1 - I_A))$

$$f(Impact) = \begin{cases} 0, & Impact = 0, \\ 1,176, & \text{інакше} \end{cases}$$

2. Модель загального рівня захищеності програмного продукту

Метрика CVSS розглядає та оцінює окремі вразливості. Запропонуємо модель для оцінки загального рівня захищеності програмного продукту.

Будемо розглядати захищеність продукту на основі сукупності знайдених в ньому вразливостей на даний момент часу. Оскільки програмне забезпечення розвивається, періодично виходять нові версії та оновлення безпеки, з'являться нова характеристика, яка показує наскільки оперативно (швидко) реагує розробник та випускає оновлення безпеки.

Природно, що чим довше вразливість не усувається, тим привабливішою вона є для злоумисників: відбувається накопичення інформації щодо вразливості, технічних деталей для її експлуатації, що призводить до збільшення ймовірності успішної атаки. Найпривабливішими для злоумисників і найнеприємнішими для тих, хто відповідає за захист інформації в інформаційній системі є вразливості нульового дня (*zero-day vulnerabilities*). Введемо, показник «життя вразливості», який є різницею між датою виходу оновлення, яке усуває вразливість та датою знаходження вразливості, нормоване на середню кількість днів у місяці (виходячи з припущення, що в середньому оновлення виходять один раз на місяць) або кількість днів у тижні, тощо. Зауважено, що іноді вразливості знаходять у старих версіях, а в нових версіях вони відсутні. В такому випадку інтерес злоумисників до вразливості значно знижується, тому будемо вважати цей показник рівним нулю. Для більш детальних та точних розрахунків слід враховувати, що навіть після виходу оновлення програмного забезпечення, пройде певний час доки таке оновлення буде встановлено. Цей час залежить від оперативності роботи відділу з безпеки інформації або системних адміністраторів організації. В роботі розглядається частковий випадок, коли оновлення встановлюється в межах робочого дня. Введемо формулу для розрахунку 2:

$$S = round((0,6 \cdot Impact + 0,4 \cdot Exploitability \cdot g(\Delta) - 1,5) \cdot f(Impact)) \quad (2)$$

де:

$date_{fix}$ - дата виходу оновлення

$date_{report}$ - дата знаходження вразливості

$$\Delta = date_{fix} - date_{report}$$

$$g(\Delta) = \begin{cases} 0, & \Delta \leq 0, \\ \frac{\Delta}{30}, & \text{інакше} \end{cases}$$

Розрахувавши середнє арифметичне значення скорегованих базових оцінок по всім вразливостям, отримуємо оцінку рівня захищеності програмного продукту $Q = \frac{1}{n} \sum_{i=1}^n S_i, i = 1 \dots n$. Такі оцінки можна порівнювати між собою для різних продуктів (за умови вибірки вразливостей за одноковий період часу).

3. Приклад обчислення

Розглянемо приклад, в якому за задачу поставлено обрати веб-браузер який є більш захищеним за методом запропонованим у роботі. Кандидати: Internet Explorer 11.0 та FireFox 17.0.8, які були випущенні майже одночасно восени 2013 року. Для збору статистичної інформації про вразливості використовувалася NVD [?]. Розрахунок проводимо згідно з рівняння 2. Дані про вразливості наведені в таблиці 1 та 2:

CVE	CVSS	Δ	$g(\Delta)$	Оцінка
CVE-2013-5051	9,3	58	1,93	13,11
CVE-2013-5048	9,3	58	1,93	13,11
CVE-2013-5047	9,3	58	1,93	13,11
CVE-2013-5046	6,2	58	1,93	7,02
CVE-2013-5045	6,2	58	1,93	7,02
CVE-2013-3917	9,3	31	1,03	9,47
CVE-2013-3916	9,3	31	1,03	9,47
CVE-2013-3915	9,3	31	1,03	9,47
CVE-2013-3914	9,3	31	1,03	9,47
CVE-2013-3912	9,3	31	1,03	9,47
CVE-2013-3897	9,3	-1	0	5,92
CVE-2013-3893	9,3	20	0,67	7,99

Табл. 1. Таблиця вразливостей для оцінки захищеності Internet Explorer 11.0

Середня арифметична оцінка – 9,55

Середня арифметична оцінка – 7,26

Таким чином, за результатами оцінок (7,26 < 9,55), можна зробити висновок, що на момент оцінювання FireFox 17.0.8 – є більш привабливим кандидатом для застосування з точки зору політики безпеки інформації.

CVE	CVSS	Δ	$g(\Delta)$	Оцінка
CVE-2013-6673	5,8	28	0,93	5,47
CVE-2013-6672	4,3	28	0,93	4,06
CVE-2013-6671	9,3	28	0,93	9,07
CVE-2013-5619	6,8	28	0,93	6,53
CVE-2013-5618	10,0	28	0,93	9,68
CVE-2013-5616	10,0	28	0,93	9,68
CVE-2013-5615	10,0	28	0,93	9,68
CVE-2013-5614	4,3	28	0,93	4,06
CVE-2013-5613	9,3	28	0,93	9,68
CVE-2013-5612	9,3	28	0,93	4,06
CVE-2013-5611	9,3	28	0,93	5,47
CVE-2013-5610	9,3	28	0,93	9,68

Табл. 2. Таблиця вразливостей для оцінки захищеності
Firefox 17.0.8

Висновки

В роботі показано, що на основі метрики суворості вразливостей CVSS можна побудувати власні

моделі для оцінювання сукупності вразливостей та програмного продукту загалом. В запропонованій моделі вводиться додатковий показник, який враховує підвищення ймовірності експлуатації вразливості до моменту виходу оновлення безпеки.

Для подальшого вдосконалення та уточнення моделі, пропонується переглянути припущення зроблені в розділі 2, зокрема випадок, коли застосовується стара версія програмного продукту при наявному оновленні: насправді інтерес зломисників може навіть зрости, якщо буде зберігатися тенденція користувачів, з яких-небудь причин, використовувати саме стару версію (наприклад, Windows XP є цілком неприйнятною з точки зору безпеки, але досі велика кількість користувачів використовує саме її).

Врахування цього та інших зовнішніх факторів дасть більш точну оцінку.