

[DOI 10.28925/2663-4023.2026.35.1449](https://doi.org/10.28925/2663-4023.2026.35.1449)

УДК 004.82:004.6

Осійчук Іван

аспірант кафедри кібербезпеки

Тернопільський національний технічний університет імені Івана Пулюя, Тернопіль, Україна

ORCID: 0009-0003-6844-2374

ivan_osiichuk0108@ntu.edu.ua**Загородна Наталія**

кандидат технічних наук, доцент,

завідувач кафедри кібербезпеки

Тернопільський національний технічний університет імені Івана Пулюя, Тернопіль, Україна

ORCID: 0000-0002-1808-835X

zagorodna.n@gmail.com**Ярема Олег**

аспірант кафедри кібербезпеки

Тернопільський національний технічний університет імені Івана Пулюя, Тернопіль, Україна

ORCID: 0009-0009-8709-7813

yarema.oleh.m@gmail.com

УДОСКОНАЛЕННЯ ПІДХОДУ ДО ПОБУДОВИ УНІФІКОВАНОГО ГРАФУ ЗНАЬ НА ОСНОВІ ВІДОМИХ ПУБЛІЧНИХ КАТАЛОГІВ В СФЕРІ БЕЗПЕКИ ПРОГРАМНОГО ТА АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ЗАДАЧ ВАЛІДАЦІЇ

Анотація. Існує чотири відомі публічні каталоги, які описують безпеку програмного та апаратного забезпечення: CVE List реєструє вразливості, CWE узагальнює вразливості через типові слабкі місця (слабкості) проєктування та реалізації інформаційних систем, CAPEC описує шаблони атак, а ATT&CK – техніки зловмисників. Побудова уніфікованого графу знань, що поєднує їх, дозволяє пройти від запису про вразливість до техніки виконання атаки, зумовленою цією вразливістю. Проте більшість відомих підходів до побудови графів не враховують додаткової інформації, яку публікують каталоги: автора мітки вразливості, тип контейнера запису, напрямки зв'язків між каталогами, ієрархію зв'язків CWE та версію релізу. Мета статті – побудувати онтологію узагальненого графу знань цих каталогів, яка максимально точно відображає наявну інформацію та зв'язки між різними сутностями. В основу підходу покладено принцип семантичного перенесення (lifting): кожне міжкаталогове посилання конвертується в RDF із повним збереженням напрямку, авторства, версії релізу та супутніх полів джерела, без генерації штучних фактів. Зазначений принцип реалізується через наступні правила. Зв'язок, який описується додатковими метаданими в каталозі, є вузлом, а не ребром. Забороняється автоматична генерація зворотнього симетричного зв'язку у випадку його відсутності в оригінальних каталогах. Технічна реалізація зазначених вимог забезпечується інструментами SHACL через чотири перевірки: форми, тотожність знімка, перевірка повноти відтворення, контроль неперенесених даних, що верифікують коректність перенесення. Удосконалена модель графу знань дозволяє виявляти різного роду неузгодженості: у частині записів власник і авторизований публікатор даних призначають різні вразливості; зіставлення CAPEC з ATT&CK і посилання ATT&CK на CAPEC не мають жодної спільної пари; частина вразливостей має різних батьків у різних поданнях (view) CWE, що дозволяє її використання для задач валідації. В той самий час існуючі моделі графу знань, побудовані на даних тих самих каталогів дозволяють виконувати лише обхід. Код, онтологія, форми та запити опубліковані у відкритому репозиторії.

Ключові слова: граф знань; онтологія; вразливості програмного та апаратного забезпечення; походження даних; валідація; CVE; CWE; CAPEC; ATT&CK



ВСТУП

Ефективне забезпечення кібербезпеки сучасних систем вимагає цілісного аналізу всієї вертикалі загроз – від конкретних дефектів коду до високорівневих тактик нападників. Для реалізації такого підходу аналітики спираються на екосистему публічних реєстрів та каталогів знань про загрози. Існують чотири найбільш відомі публічні каталоги, які описують одну предметну область безпеки програмного та апаратного забезпечення з різних сторін: CVE List реєструє вразливості [1], CWE класифікує слабкі місця, що за ними стоять [2], CAPEC описує шаблони атак, які користуються цими слабкостями [3], а ATT&CK описує техніки зловмисників [4], [5]. Каталоги посилаються один на одного: запис CVE асоціюється з конкретними категоріями слабкостей CWE, призначеними авторами, CWE і CAPEC посилаються на записи один одного, а CAPEC зіставляє свої шаблони з техніками ATT&CK.

Перехресні посилання в згаданих вище каталогах є твердженнями, які певна сторона зробила у певному релізі та в певному напрямку. Уповноважена організація присвоює запису CVE мітку слабкості (CWE) в конкретному контейнері запису (відокремлений блок метаданих у структурі запису, який розмежовує твердження різних організацій): орган нумерації CVE (CNA), якому запис належить, або авторизований публікатор даних (ADP), який доповнює запис. Зіставлення CAPEC з ATT&CK є твердженням CAPEC, а ATT&CK може заявити зворотнє, а може й ні. Батьківський зв'язок CWE заявлено в межах одного подання (view) CWE, і в іншому поданні ту саму слабкість може мати інший батько.

Постановка проблеми. Структурування зв'язків між каталогами вразливостей, слабких місць і тактик та технік атак дає змогу простежити шлях від конкретного запису про вразливість до технік, які її експлуатація може уможливити. Природною моделлю для представлення таких взаємозв'язків є графи знань. Граф знань – це структуроване представлення знань у вигляді графа, вузли якого відповідають сутностям предметної області, а ребра відображають семантичні відношення між ними. На відміну від звичайних графових моделей, графи знань дають змогу формалізувати семантику зв'язків, забезпечити їхню машинну інтерпретацію та інтегрувати інформацію з різномірних джерел. У науковій літературі вже представлені дослідження, присвячені моделюванню зв'язків між вище згаданими каталогами даних за допомогою графів знань, проте вирішують задачі обходу графа і доступу до інформації. Можливості використання таких моделей для задач валідації даних – перевірки фактів та узгодженості структури з визначеними правилами, залишаються достатньо обмеженими.

Аналіз останніх досліджень і публікацій. Онтологічне моделювання в кібербезпеці почалося з концептуальних онтологій UCO та Stucco [6]–[7]. Вони впорядкували поняття загрозового ландшафту (вразливості, слабкі місця, загрози, атрибути атак) і дозволили об'єднувати різномірні джерела. SEPSES [8]–[9], BRON [10], [11] та OdTM [12] будують графи знань на первинних каталогах CVE, CWE, CAPEC, ATT&CK. Слід зазначити, що SEPSES [8] автоматизує перенесення мовою відображень RML і перевіряє результат формами SHACL [9], проте зв'язок CVE-CWE подає простою властивістю з бази NVD і зберігає лише останній знімок даних, без авторів і контексту тверджень. BRON [10], [11] об'єднує каталоги в графовій базі даних для задач пошуку загроз (threat hunting), але робить усі односторонні посилання між каталогами двосторонніми ребрами, тож напрямок твердження й реліз факту втрачаються. OdTM [12] призначено для автоматизації моделювання загроз на діаграмах потоків даних (DFD): у ній властивості оголошено оберненими, а шляхи будуються ланцюжками властивостей OWL, через що проміжні кроки та походження фактів губляться. Для порівняння проєктних рішень (табл. 1) ми обрали саме ці три графи: вони охоплюють ті самі каталоги, достатньо докладно описані в літературі та представляють підхід, у якому граф оптимізовано для обходу шляхів.

Таблиця 1

Порівняльний огляд підходів до побудови графів на основі каталогів CVE, CWE, CAPEC і ATT&CK

	SEPSES [8]	BRON [10], [11]	OdTM [12]
Мітка CVE-CWE	проста властивість із NVD	просте ребро з NVD	не переноситься
Реліз факту	остання версія; знімки для завантаження	перебудова із завантажень	не описано
Односторонній зв'язок між каталогами	-	прохідний в обидва боки	оголошено оберненим, замкнено системою виведення
Батьківський зв'язок CWE	-	-	-



	SEPSSES [8]	BRON [10], [11]	OdTM [12]
Статус запису	статус CWE як літерал	-	-
Шлях від запису до техніки	-	обходить, не записується	ланцюжок властивостей, кроки губляться
Перевірене перенесення	не описано	не описано	не описано

Наведені вище дослідження добре підходять для обходу графа і допомагають знайти шлях від запису про вразливість до технік, які вона може уможливити. Для цього вони об'єднують каталоги в одну структуру й заради зв'язності жертвують повною відповідністю до оригінальних даних, зокрема мітка стає ребром без автора, односторонній зв'язок стає двостороннім, а новий реліз замінює попередній. Така структура не відповідає потребам валідації, оскільки не забезпечує повного відображення даних вихідних каталогів і може породжувати нові факти, не представлені в оригінальних записях. Сучасні корпоративні графи ризиків вразливостей [13] також застосовують декларативні відображення та мову форм SHACL для валідації схем, проте вони зосереджені на абстрактних оцінках ризиків і не здійснюють повного семантичного перенесення первинних даних каталогів в граф знань (lifting). Отже актуальною є задача удосконалення підходу до перенесення даних та побудови графа, що дозволить проводити більш повну і різносторонню валідацію.

Мета статті. Метою роботи є побудова онтології та графа знань публічних каталогів безпеки (CVE, CWE, CAPEC, ATT&CK), який зберігає напрямок, контекст та походження (provenance) кожного факту без спотворення вихідних тверджень, забезпечуючи можливість валідації зв'язків та відповідності їх структури визначеним правилам.

МЕТОДИКА ДОСЛІДЖЕННЯ

Побудова будь-якої моделі починається з формулювання вимог, що визначаються поставленими задачами та особливостями вихідних даних. Для графа знань, призначеного для валідації даних досліджуваних каталогів кібербезпеки, необхідно забезпечити збереження їхньої семантики, структури та походження інформації

Зокрема, односпрямовані та взаємні зв'язки мають зберігати вихідну напрямленість. Батьківський зв'язок CWE повинен бути пов'язаний із поданням, у якому його задекларовано, а статус запису – зберігатися без змін, із розмежуванням поняття «виведений з ужитку» відповідно до специфіки кожного каталогу. Шлях від запису про вразливість до відповідної техніки має містити твердження, що обґрунтовують кожен його крок, а відповідність перенесених даних оригінальним джерелам повинна бути придатною для перевірки.

Дотримання цих вимог забезпечує достовірність валідації: якщо під час перетворення даних у RDF генеруються нові факти, перевірка стосуватиметься вже результатів перетворення, а не оригінальних вихідних даних. Водночас збереження походження інформації дає змогу в разі виявлення помилок локалізувати проблемний зв'язок у конкретному каталозі та забезпечити належне документування або виправлення виявлених невідповідностей.

На основі описаних вище вимог сформуємо ключове правило перенесення даних каталогів у граф знань: кожне посилання, яке джерело публікує на запис перенесеного каталогу, переноситься з напрямком, автором, релізом і полями, які дає джерело, і нічого не додається, чого джерело не стверджує.

На основі правила перенесення було сформульовано наступні чотири правила для побудови онтології, що формує схему графу знань, описуючи всі сутності та зв'язки між ними. Технічно, онтологія є OWL-словником із власних класів і властивостей в одному файлі Turtle, що публікується разом із конвертерами.

Онтологія не повинна містити аксіом, що зумовлюють виведення нових тверджень, зокрема оголошень областей визначення (domain), областей значень (range), обернених або симетричних властивостей. Наприклад, визначення області значень усіх слабкостей CWE в межах одного класу weakness не дає можливостей для класифікації CWE, зокрема це б дозволило помилково розпізнавати категорію слабкостей CWE-264 як звичайну слабкість та використати як мітку для CVE. Аналогічно, оголошення взаємного зв'язку між CAPEC та ATT&CK на основі існуючого посилання CAPEC на ATT&CK спричинило б автоматичне генерування відсутніх у вихідних даних зворотних зв'язків.

Зворотно-спрямовані посилання між каталогами слід представляти за допомогою анотаційних властивостей (owl:AnnotationProperty), що забезпечує документування взаємозв'язків без активації механізмів логічного виведення (reasoning) та запобігає автоматичній генерації симетричних



фактів.

Для збереження походження даних (provenance) потрібно застосувати механізм реіфікації (reification)[14]: бінарні зв'язки між сутностями необхідно моделювати як самостійні вузли-асоціації, що дає змогу явно фіксувати атрибути зв'язку, зокрема автора, контейнер і контекст. Ребро повинне відображати лише зв'язок між сутностями, тоді як властивості цього зв'язку мають зберігатися окремо для забезпечення повноти метаданих і простежуваності походження. Кожен вузол-зв'язок повинен мати рівно одне джерело й одну ціль (наприклад, джерело CVE та ціль CWE), а всі атрибути зв'язку (автор, контейнер, подання й текст) мають належати відповідному вузлу.

Розглянемо реіфікацію на прикладі простого ребра:

```
cve:CVE-2026-16105 ex:hasWeakness cwe:CWE-639
```

і його реіфікованого вузла:

```
ex:assignment1 a ex:WeaknessLabel ;  
  ex:source      cve:CVE-2026-16105 ;  
  ex:target      cwe:CWE-639 ;  
  ex:assignedBy  party:redhat ;  
  ex:container   ex:CNA ;
```

Ще одним правилом є запобігання змішування самих даних, та їх метаданих, а також технічної інформації. Метадані та виведену технічну інформацію потрібно зберігати окремо від даних, взятих з каталогів.

Отже, для задач валідації потрібна система, яка зберігає складні зв'язки (наприклад, CVE-CWE або CVE-CAPEC) як сутність з власними метаданими: записом, контейнером, ціллю та автором, а також зберігає реліз кожного каталогу, що дає можливість відстежити походження даних.

РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Для забезпечення відтворюваності результатів і достовірності валідації необхідно чітко визначити джерела даних та зафіксувати їхні версії. Розглянемо детальніше каталоги, що використовуються для побудови графа знань. Граф будується над зафіксованими релізами: CVE List 2026-09-02_1800Z, CWE 4.20, CAPEC 3.9 і ATT&CK Enterprise 19.2. Для кожного файлу каталогу обчислюється хеш-сума SHA-256, яка зберігається разом з завантаженим каталогом.

Каталог CVE List 2026-09-02_1800Z містить по одному JSON-запису на кожну вразливість; у зафіксованому релізі налічується 385524 записи. Кожен запис містить один контейнер CNA, сформований організацією, відповідальною за цей запис, а також може містити довільну кількість контейнерів ADP. Кожен контейнер може позначити запис ідентифікаторами-мітками CWE, CAPEC та інших таксономій. Отже, мітка має чотири координати: запис, контейнер, ціль і організацію-автора, що її поставила. У певних випадках мітка запису може бути категорією або поданням або ж бути відсутньою у відповідному релізі каталогу, наприклад це може бути стара мітка, яка була прибрана з останнього каталогу.

Перейдемо до розгляду каталогу CWE, який використовується в даному дослідженні. CWE 4.20 містить 969 слабкостей, які можуть бути використані для позначення CVE, а також категорії та подання: категорії (category) групують слабкості за темою, а подання (view) впорядковують слабкості в ієрархії. Варто зазначити, що в межах одного подання відображаються зв'язки між слабкостями (ChildOf, PeerOf, CanPrecede та ін.), проте одна і та сама слабкість може мати різні батьківські зв'язки в різних поданнях.

Розглянемо детальніше каталоги CAPEC і ATT&CK. CAPEC 3.9 містить 615 шаблонів атак, пов'язаних зі слабкостями CWE та зіставлених з техніками ATT&CK. ATT&CK Enterprise 19.2 публікується пакетом STIX 2.1 [5]: техніки й тактики є об'єктами STIX, а зв'язки між ними – окремими об'єктами-зв'язками з власними ідентифікаторами й датами.

Крім того, записи в каталогах можуть мати різні статуси, які необхідно враховувати під час побудови графа. Статус виведеного з ужитку присвоюється запису лише за наявності відповідної позначки у вихідному каталозі: Deprecated у CWE та CAPEC, revoked або deprecated в ATT&CK. У графі цей статус відображається за допомогою властивості owl:deprecated, водночас його оригінальне значення зберігається без змін. Записи зі статусом Obsolete, що залишаються чинними, але є неактуальними, до категорії виведених з ужитку не належать. Табл. 2 дає розмір графів, що представляють окремі каталоги.

Розмір графів зафіксованих релізів: вузли-зв'язки за видами та кількістю трійок.

Граф	Вузли-зв'язки за видами	Трійки
CVE List	мітки вразливостей: 164 561 від CNA, 39 261 від ADP; мітки шаблонів атак: 28 068; мітки технік: 2; записів: 385 524; сторін: 487	4 084 840
CWE	зв'язки: 1 602; членства: 5 024; спостережені приклади: 3 125; пропозиції зіставлення: 63	56 144
CAPEC	зв'язки: 727; зіставлення з ATT&CK: 272	9 901
ATT&CK	об'єкти-зв'язки: 2 074	23 185

На основі сформульованих правил перенесення, а також з врахуванням особливостей записів та зв'язків між ними, було побудовано онтологію графа знань на основі чотирьох зафіксованих релізів каталогів CWE, CVE, CAPEC, ATT&CK. Набір даних складається з іменованих графів [15] трьох видів: граф змісту, граф метаданих та похідний граф.

Граф змісту будується окремо для кожного релізу джерела та містить виключно твердження, зафіксовані в цьому релізі. Його ім'я складається з абрєвіатури каталогу-джерела та перших дванадцяти шістнадцяткових цифр хеш-суми фіксованого завантаженого релізу цього каталогу, тому однакові завантаження дають однакові імена (за аналогією з Trusty URI [16], але ім'я походить від вмісту вхідного файлу, а не самого графа). Рисунок 1 відображає ядро графа, у якому окремі рамки відповідають каталогам, наведено основні типи записів і пронумеровано зв'язки між ними. Одинарна стрілка позначає просте відношення, подвійна рамка – вузол-зв'язок, до якого від запису-джерела йде лінія, а від вузла до запису-цілі – стрілка. Лінія та стрілка, що представляють один зв'язок, мають спільний номер. Напрямок стрілок відповідає тому, як зв'язки визначені в каталозі-публікаторі, а пара зустрічних стрілок позначає два окремі відношення, жодне з яких не виводиться з іншого.

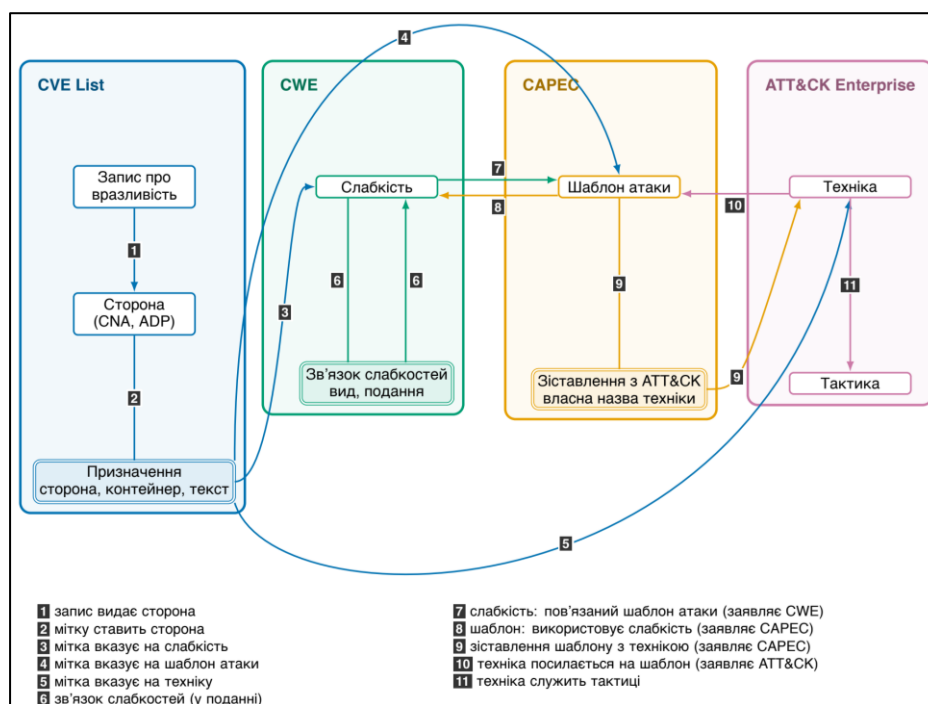


Рис. 1. Ядро графа змісту: види записів кожного каталогу та зв'язки між ними

Граф метаданих (рис.2) виступає єдиним спеціалізованим іменованим графом, який виділено зберігає технічні відомості про завантаження всіх джерел даних, а саме у ньому автоматично реєструються точна часова позначка завантаження, URL джерела даних, заявлена версія та хеш-сума SHA-256 кожного релізу.

З технічного погляду, система використовує цей граф для автоматизованої перевірки повноти перенесення даних із каталогів до графа змісту шляхом зіставлення спеціальних вузлів-лічильників із вихідними файлами, а також для фіксації неперенесених даних у разі наявності посилань на каталоги, що

не охоплюються дослідженням. Граф метаданих забезпечує відокремлення службових параметрів від основних фактів і похідних графів.

Для спеціаліста з кібербезпеки граф метаданих забезпечує повну простежуваність походження даних від конкретного запису про вразливість до оригінального джерела, гарантує абсолютну відтворюваність системи завдяки використанню хешів файлів каталогів, а також надає прозорий звіт про повноту перенесення даних вихідних каталогів та свідомо виключені дані.

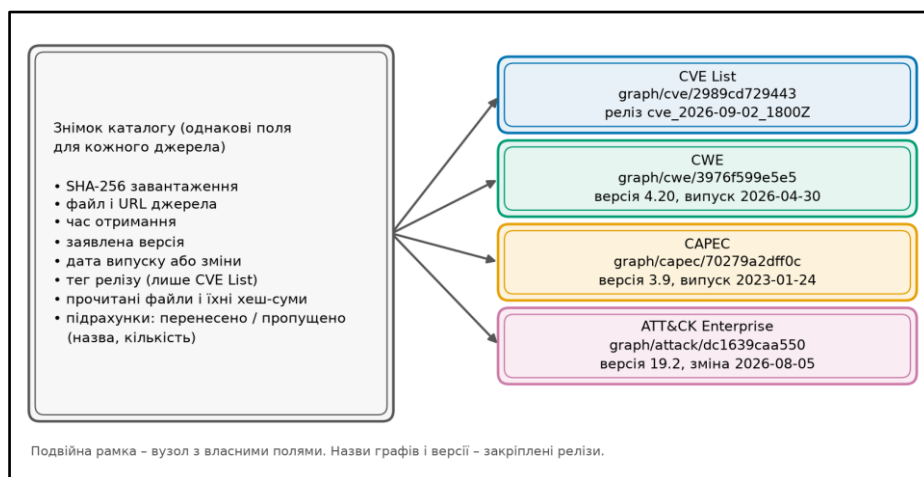


Рис. 2. Граф метаданих: права частина є окремими каталогами в межах графа вмісту

Похідний граф містить результати обчислень, виконаних на основі графів вмісту, зокрема похідні шляхи. Трійки цього графа не додаються до графа вмісту, що запобігає змішуванню виведених даних з оригінальними. Похідний шлях, збережений в цьому графі, іде від вразливості через слабкість і шаблон атаки до техніки, і кожен його крок зберігається разом із твердженнями, які цей крок підтверджують. Перший крок підтверджує мітка слабкості в записі CVE. Другий крок підтверджує або пов'язаний шаблон атаки в CWE (зв'язок заявлено в напрямку руху, тому він прямий), або пов'язана слабкість у CAPEC (зв'язок заявлено з іншого кінця, тому він зворотний). Третій крок підтверджує зіставлення CAPEC з ATT&CK (прямий зв'язок) або посилання ATT&CK на CAPEC (зворотний).

Загальна кількість підтверджених варіантів для ланцюжка зв'язків розраховується як добуток фактів на кожному його кроці. Наприклад, на рисунку 3 наведено приклад похідного графу, де вказано, що для запису CVE-2018-12538 присвоєна одна мітка слабкості CWE-6 (1 факт). На кроці 2 зв'язок між CWE-6 та CAPEC-21 підтверджено з обох боків (2 факти: від CWE до CAPEC і навпаки), а перехід від CAPEC-21 до техніки T1134 (крок 3) задекларовано лише з боку CAPEC (1 факт).

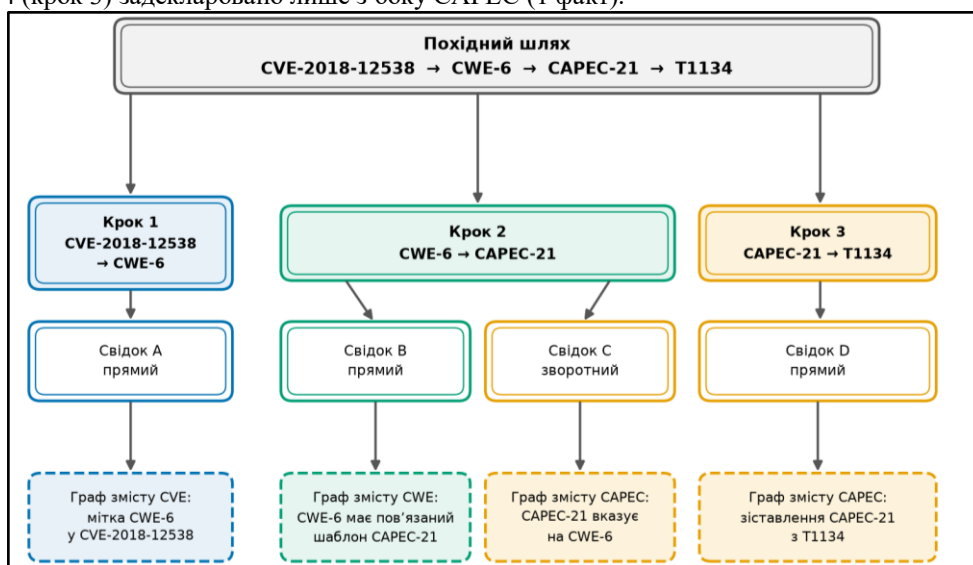


Рис. 3. Приклад похідного графу



Роздільне зберігання цих трьох графів гарантує, що дані джерела, спосіб їх отримання та обчислені на їхній основі результати ніколи не змішуються.

З метою збереження чіткої концептуальної межі графа, до графа змісту свідомо не переносяться зв'язки на зовнішні таксономії (OWASP, WASC тощо) та об'єкти кіберрозвідки (CTI) з каталогу ATT&CK, такі як угруповання нападників (Groups), використовуване програмне забезпечення (Software) та кампанії атак (Campaigns). Це зумовлено тим, що фокус дослідження зосереджено на фундаментальному класифікаційному ланцюжку кібербезпеки: «запис про вразливість (CVE) → слабкість коду чи архітектури (CWE) → шаблон атаки (CAPEC) → техніка нападника (ATT&CK)». Включення динамічних об'єктів оперативного CTI-рівня та зовнішніх ресурсів призвело б до розмивання семантичних меж графа та появи незамкнених зовнішніх посилань.

Перетворення кожного каталогу у формат RDF супроводжується чотирма перевітками цілісності: структурної відповідності, тотожності знімка, повноти відтворення та контролю неперенесених даних (залишку). Сукупність цих перевірок дає змогу оцінити коректність перетворення, детермінованість процесу та повноту перенесення вихідних даних.

Форми. Графи змісту (для CVE — вибірка з 21 694 записів і 332 299 трійок), граф метаданих і похідний граф перевіряються на відповідність формам SHACL Core [9]. Перевірки охоплюють наявність єдиної мітки для кожної сутності призначення (assignment), належність внутрішніх міток до очікуваного класу сутностей і відповідність зовнішніх посилань встановленим шаблонам просторів імен. Наприклад, посилання на батьківський запис CWE має вказувати на сутність того самого класу, а не на сутність іншого типу. Відсутність цільового запису в зафіксованому релізі іншого каталогу не розглядається як порушення структурної цілісності графа, а фіксується за результатами аналітичного запиту як потенційна прогалина або дефект вихідних даних.

Тотожність знімка. Для підтвердження детермінованості конвертації кожен каталог перетворюється у RDF двічі у двох незалежних процесах, після чого отримані RDF-файли звіряються побайтово.

Перевірка повноти відтворення. Для кожного типу зв'язків кількість об'єктів у графі метаданих порівнюється з кількістю елементів, підрахованою безпосередньо з вихідного файлу (використовуючи підходи, відмінні від перенесення в RDF, наприклад, пошук по JSON).

Контроль неперенесених даних (залишку). За допомогою SHACL-форм усі згенеровані RDF-факти звіряються із шаблонами (еталонними описами допустимої структури вузлів) конвертера (інструменту переведення вихідних даних у RDF). Відсутність залишків, що не в шаблоні, доводить, що вхідні каталоги перенесено повністю та без помилок. Кількість неперенесених даних фіксується в графі метаданих.

Отже, SHACL-форми забезпечують перевірку структурної відповідності встановленим правилам, тотожність знімка підтверджує детермінованість перетворення, а перевірки повноти та залишку дають змогу виявити втрати або викривлення даних.

На відміну від графів, орієнтованих виключно на обхід шляхів, запропонована модель дозволяє виявити аналітичні деталі, які зазвичай втрачаються при спрощенні. Завдяки врахуванню авторства міток стає можливим визначати випадки, коли вразливість отримує різні мітки, встановлені різними учасниками, а також розмежовувати записи, для яких класифікацію здійснив лише авторизований публікатор (ADP). Наприклад, для запису CVE-2026-16105 можна встановити, що власник запису Red Hat визначив слабкість CWE-639, тоді як CISA-ADP вказав CWE-862. Спрощений граф із бінарним зв'язком «вразливість-слабкість» містив би лише два однорангових ребра, повністю приховуючи такі випадки.

ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

У результаті дослідження розроблено онтологію та побудовано граф знань публічних каталогів безпеки (CVE, CWE, CAPEC, ATT&CK), який забезпечує точне збереження напрямку, контексту та походження (provenance) кожного факту без внесення штучної додаткової інформації. В основу побудованого рішення покладено правило перенесення, за яким кожен зв'язок конвертується в RDF із повним збереженням автора, контейнера, релізу та напрямку.

Дотримання цього принципу гарантовано за допомогою чотирьох розроблених онтологічних правил. Насамперед, онтологія свідомо не містить аксіом логічного виведення, що запобігає виведенню некоректних фактів, як-от хибному використанню окремих категорій CWE як слабкостей в записах CVE, або автоматичному генеруванню зворотних ребер. Зворотно-спрямовані посилання між каталогами пов'язані через анотаційні властивості, що забезпечує збереження взаємозв'язку без активації виведення фактів (reasoning). Крім того, рефіковані вузли-зв'язки мають сувору бінарну структуру з одним джерелом та однією ціллю, де атрибути автора, контейнера й тексту належать безпосередньо самому вузлу. Нарешті, терміни графа змісту чітко відокремлені від термінів похідного графа та графа метаданих, що повністю



виключає змішування виведених тверджень із первинними опублікованими даними.

Правдивість і точність перенесення даних підтверджено за допомогою чотирьохрівневого контролю, який включає перевірку форм, тотожності знімка, повноти відтворення та контроль неперенесених даних. На відміну від систем, орієнтованих виключно на обхід шляхів, побудована модель дозволяє виявити реальні аналітичні розбіжності у вихідних джерелах: розходження між мітками власників записів (CNA) та авторизованих публікаторів (ADP), односторонній характер зіставлень CAPEC із ATT&CK, а також наявність різних батьків у одних і тих же слабкостей залежно від обраного подання CWE.

Отримане рішення відкриває кілька ключових напрямів подальших досліджень. Перший напрям передбачає повну формалізацію онтології з математичним обґрунтуванням та розробкою специфікації її розширення на нові каталоги. Другий напрям орієнтований на задачі валідації, зокрема на перевірку вмісту самих каталогів за їх внутрішніми правилами, для прикладу, проведення аналізу відповідності міток слабкостей настановам CWE та оцінювання частоти відхилень у різних публікаторів. Третій напрям передбачає побудову й аналіз спрощених проєкцій графа для оцінювання втрат аналітичної інформації внаслідок виведення нових зв'язків або ігнорування авторства міток. Нарешті, окремою практичною задачею є перенесення послідовних релізів кожного джерела для аналізу часової динаміки зміни міток, статусу записів та міжкаталожних зв'язків.

Використання штучного інтелекту: під час підготовки статті автори користувалися інструментами на основі штучного інтелекту: Claude для допомоги в написанні та редагуванні тексту, підготовки програмного коду конвертерів і перевірок, оформлення рисунків й таблиць, а також NotebookLM для роботи з літературними джерелами.

Доступність: код, словник, форми й запити опубліковані в <https://github.com/IvanOsiichuk/cskg>. Дані каталогів залишаються на умовах їхніх публікаторів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. CVE Program. (2026). *CVE List V5 and the CVE JSON 5 Record Format*. <https://github.com/CVEProject/cvelistV5>
2. The MITRE Corporation. (2026). *Common Weakness Enumeration (CWE)*. <https://cwe.mitre.org/>
3. The MITRE Corporation. (2023). *Common Attack Pattern Enumerations and Classifications (CAPEC)*. <https://capec.mitre.org/>
4. The MITRE Corporation. (2026). *MITRE ATT&CK Enterprise, STIX 2.1 Bundle*. <https://github.com/mitre-attack/attack-stix-data>
5. OASIS Cyber Threat Intelligence Technical Committee. (2021). *STIX Version 2.1 (OASIS Standard)*. OASIS. <https://docs.oasis-open.org/cti/stix/v2.1/stix-v2.1.html>
6. Syed, Z., Padia, A., Finin, T., Mathews, L., & Joshi, A. (2016). UCO: A Unified Cybersecurity Ontology. In *AAAI Workshop on Artificial Intelligence for Cyber Security*. AAAI Press.
7. Iannacone, M., Bohn, S., Nakamura, G., Gerth, J., Huffer, K., Bridges, R., Ferragut, E., & Goodall, J. (2015). Developing an Ontology for Cyber Security Knowledge Graphs. In *Proceedings of the 10th Annual Cyber and Information Security Research Conference (CISRC)* (pp. 1–4). ACM. <https://doi.org/10.1145/2746266.2746278>
8. Kiesling, E., Ekelhart, A., Kurniawan, K., & Ekaputra, F. (2019). The SEPSES Knowledge Graph: An Integrated Resource for Cybersecurity. In *The Semantic Web – ISWC 2019* (pp. 198–214). Springer. https://doi.org/10.1007/978-3-030-30796-7_13
9. Knublauch, H. & Kontokostas, D. (2017). *Shapes Constraint Language (SHACL)* (W3C Recommendation). W3C. <https://www.w3.org/TR/shacl/>
10. Hemberg, E., Turner, M. J., Rutar, N., & O'Reilly, U.-M. (2024). Enhancements to Threat, Vulnerability, and Mitigation Knowledge for Cyber Analytics, Hunting, and Simulations. *Digital Threats: Research and Practice*, 5(1), 1–33. <https://doi.org/10.1145/3615668>
11. Hemberg, E., Kelly, J., Shlapentokh-Rothman, M., Reinstadler, B., Xu, K., Rutar, N., & O'Reilly, U.-M. (2020). *Linking Threat Tactics, Techniques, and Patterns with Defensive Weaknesses, Vulnerabilities and Affected Platform Configurations for Cyber Hunting*. arXiv preprint arXiv:2010.00533.
12. Brazhuk, A. (2021). *Towards Automation of Threat Modeling Based on a Semantic Model of Attack Patterns and Weaknesses*. arXiv preprint arXiv:2112.04231.
13. Egaña Aranguren, M., Fernández-Breis, J. T., Leon Balentzia, B., Rompe, M., & García Castro, A. (2026). A Comprehensive View of Software Vulnerability Risks through Enterprise Knowledge Graphs. *Computers & Security*, 163, 104815. <https://doi.org/10.1016/j.cose.2025.104815>
14. Hernández, D., Hogan, A., & Krötzsch, M. (2015). Reifying RDF: What Works Well With Wikidata? In



- Proceedings of the 11th International Workshop on Scalable Semantic Web Knowledge Base Systems (SSWS 2015).*
15. Carroll, J. J., Bizer, C., Hayes, P., & Stickler, P. (2005). Named Graphs, Provenance and Trust. In *Proceedings of the 14th International Conference on World Wide Web (WWW '05)* (pp. 613). ACM. <https://doi.org/10.1145/1060745.1060835>
 16. Kuhn, T. & Dumontier, M. (2014). Trusty URIs: Verifiable, Immutable, and Permanent Digital Artifacts for Linked Data. In *The Semantic Web: Trends and Challenges (ESWC 2014)* (pp. 395–410). Springer. https://doi.org/10.1007/978-3-319-07443-6_27

**Osiichuk Ivan**

PhD student of the Cybersecurity Department
Ternopil Ivan Puluj National Technical University, Ternopil, Ukraine
ORCID: 0009-0003-6844-2374
ivan_osiichuk0108@tntu.edu.ua

Zagorodna Nataliya

PhD (Technical Sciences), Head of the Cybersecurity Department
Ternopil Ivan Puluj National Technical University, Ternopil, Ukraine
ORCID: 0000-0002-1808-835X
zagorodna.n@gmail.com

Yarema Oleh

PhD student of the Cybersecurity Department
Ternopil Ivan Puluj National Technical University, Ternopil, Ukraine
ORCID: 0009-0009-8709-7813
yarema.oleh.m@gmail.com

IMPROVING THE APPROACH TO CONSTRUCTING A UNIFIED KNOWLEDGE GRAPH BASED ON KNOWN SOFTWARE AND HARDWARE VULNERABILITY CATALOGS FOR VALIDATION TASKS

Abstract. Four well-known public catalogues describe software and hardware security: the CVE List registers vulnerabilities, CWE generalizes vulnerabilities through typical design and implementation weaknesses of information systems, CAPEC describes attack patterns, and ATT&CK outlines adversary techniques. Constructing a unified knowledge graph that links these catalogues enables tracing a path from a vulnerability record to an attack technique enabled by this vulnerability. However, most existing graph construction approaches fail to account for additional metadata published by the catalogues: the author of a weakness assignment, the record container type, cross-catalogue link directions, the CWE hierarchy, and release versions. The goal of this paper is to build an ontology for a generalized knowledge graph of these catalogues that faithfully represents the available information and relationships between entities. The approach is grounded in a strict semantic lifting principle: every cross-catalogue link is converted into RDF while fully preserving direction, authorship, release version, and accompanying source fields, without generating artificial facts. This principle is implemented through the following rules: a link described by additional metadata in a catalogue is modeled as a node rather than an edge, and automatic generation of inverse or symmetric links in the absence of explicit source statements is prohibited. Technical implementation of these requirements is realized using SHACL tools via four validation checks (Shape validation, Snapshot identity, Round-trip check, and Residue control), which verify the correctness of the lifting process. The improved knowledge graph model enables the detection of various discrepancies: in a portion of records, the record owner (CNA) and Authorized Data Publisher (ADP) assign different weaknesses; CAPEC mappings to ATT&CK and ATT&CK references to CAPEC share no common pairs; and certain weaknesses have different parents across different CWE views, making the graph suitable for validation tasks. At the same time, existing knowledge graph models built on data from the same catalogues only support graph traversal. The code, ontology, shapes, and queries are published in an open repository.

Keywords: knowledge graph; ontology; software and hardware vulnerabilities; data provenance; validation; CVE; CWE; CAPEC; ATT&CK

REFERENCES (TRANSLATED AND TRANSLITERATED)

1. CVE Program. (2026). *CVE List V5 and the CVE JSON 5 Record Format*. <https://github.com/CVEProject/cvelistV5>
2. The MITRE Corporation. (2026). *Common Weakness Enumeration (CWE)*. <https://cwe.mitre.org/>
3. The MITRE Corporation. (2023). *Common Attack Pattern Enumerations and Classifications (CAPEC)*. <https://capec.mitre.org/>



4. The MITRE Corporation. (2026). *MITRE ATT&CK Enterprise, STIX 2.1 Bundle*. <https://github.com/mitre-attack/attack-stix-data>
5. OASIS Cyber Threat Intelligence Technical Committee. (2021). *STIX Version 2.1* (OASIS Standard). OASIS. <https://docs.oasis-open.org/cti/stix/v2.1/stix-v2.1.html>
6. Syed, Z., Padia, A., Finin, T., Mathews, L., & Joshi, A. (2016). UCO: A Unified Cybersecurity Ontology. In *AAAI Workshop on Artificial Intelligence for Cyber Security*. AAAI Press.
7. Iannaccone, M., Bohn, S., Nakamura, G., Gerth, J., Huffer, K., Bridges, R., Ferragut, E., & Goodall, J. (2015). Developing an Ontology for Cyber Security Knowledge Graphs. In *Proceedings of the 10th Annual Cyber and Information Security Research Conference (CISRC)* (pp. 1–4). ACM. <https://doi.org/10.1145/2746266.2746278>
8. Kiesling, E., Ekelhart, A., Kurniawan, K., & Ekaputra, F. (2019). The SEPSES Knowledge Graph: An Integrated Resource for Cybersecurity. In *The Semantic Web – ISWC 2019* (pp. 198–214). Springer. https://doi.org/10.1007/978-3-030-30796-7_13
9. Knublauch, H. & Kontokostas, D. (2017). *Shapes Constraint Language (SHACL)* (W3C Recommendation). W3C. <https://www.w3.org/TR/shacl/>
10. Hemberg, E., Turner, M. J., Rutar, N., & O'Reilly, U.-M. (2024). Enhancements to Threat, Vulnerability, and Mitigation Knowledge for Cyber Analytics, Hunting, and Simulations. *Digital Threats: Research and Practice*, 5(1), 1–33. <https://doi.org/10.1145/3615668>
11. Hemberg, E., Kelly, J., Shlapentokh-Rothman, M., Reinstadler, B., Xu, K., Rutar, N., & O'Reilly, U.-M. (2020). *Linking Threat Tactics, Techniques, and Patterns with Defensive Weaknesses, Vulnerabilities and Affected Platform Configurations for Cyber Hunting*. arXiv preprint arXiv:2010.00533.
12. Brazhuk, A. (2021). *Towards Automation of Threat Modeling Based on a Semantic Model of Attack Patterns and Weaknesses*. arXiv preprint arXiv:2112.04231.
13. Egaña Aranguren, M., Fernández-Breis, J. T., Leon Balentzia, B., Rompe, M., & García Castro, A. (2026). A Comprehensive View of Software Vulnerability Risks through Enterprise Knowledge Graphs. *Computers & Security*, 163, 104815. <https://doi.org/10.1016/j.cose.2025.104815>
14. Hernández, D., Hogan, A., & Krötzsch, M. (2015). Reifying RDF: What Works Well With Wikidata? In *Proceedings of the 11th International Workshop on Scalable Semantic Web Knowledge Base Systems (SSWS 2015)*.
15. Carroll, J. J., Bizer, C., Hayes, P., & Stickler, P. (2005). Named Graphs, Provenance and Trust. In *Proceedings of the 14th International Conference on World Wide Web (WWW '05)* (pp. 613). ACM. <https://doi.org/10.1145/1060745.1060835>
16. Kuhn, T. & Dumontier, M. (2014). Trusty URIs: Verifiable, Immutable, and Permanent Digital Artifacts for Linked Data. In *The Semantic Web: Trends and Challenges (ESWC 2014)* (pp. 395–410). Springer. https://doi.org/10.1007/978-3-319-07443-6_27

Отримано редакцією журналу / Received: 01.09.26

Прорецензовано / Revised: 16.09.26

Схвалено до друку / Accepted: 26.10.26



This work is licensed under Creative Commons Attribution-noncommercial-sharealike 4.0 International License.