УДК 004.056.5

Д. М. Скаліуш
К. В. Безпалий

МОДЕЛЮВАННЯ ШЛЯХІВ АТАК У ГІБРИДНИХ ХМАРНИХ СЕРЕДОВИЩАХ НА ОСНОВІ SECURITY GRAPH

Вінницький національний технічний університет

Анотація

У роботі розглянуто підхід до моделювання шляхів атак у гібридних хмарних середовищах на основі Security Graph. Запропоновано використання легкої Middleware-архітектури, що поєднує збір метаданих з хмарних API, середовищ контейнерної оркестрації та джерел інформації про вразливості. Інформаційну модель подано як типізований орієнтований мультиграф, у якому вузли відображають обчислювальні активи, ідентичності, мережеві об'єкти, ресурси даних і сутності ризику, а ребра описують відношення доступу, розміщення, делегування та компрометації. Обґрунтовано застосування алгоритмів BFS, Дейкстри, метрик центральності та подієво-орієнтованого інкрементального оновлення графа для аналізу досяжності й пріоритезації ризикових маршрутів.

Ключові слова: Security Graph, шлях атаки, гібридне хмарне середовище, Kubernetes, IAM, аналіз досяжності, граф безпеки.

Abstract

The paper considers an approach to attack path modeling in hybrid cloud environments based on Security Graph. A lightweight Middleware architecture is proposed to combine metadata collection from cloud APIs, container orchestration environments, and vulnerability data sources. The information model is represented as a typed directed multigraph in which nodes describe compute assets, identities, network objects, data resources, and risk entities, while edges define access, placement, delegation, and compromise relations. The use of BFS, Dijkstra's algorithm, centrality metrics, and event-driven incremental graph updates is substantiated for reachability analysis and prioritization of risky attack routes.

Keywords: Security Graph, attack path, hybrid cloud environment, Kubernetes, IAM, reachability analysis, security graph.

Вступ

Сучасні хмарно-нативні інфраструктури поєднують сервіси хмарних провайдерів, віртуальні обчислювальні ресурси, контейнерні платформи, механізми керування доступом та зовнішні джерела даних про вразливості. У таких умовах окремі помилки конфігурації або вразливості не завжди є критичними ізольовано, проте їх комбінація може утворювати повноцінний шлях атаки від зовнішньої точки входу до критичного ресурсу даних.

Ключовою проблемою є семантичний розрив між статичними деклараціями інфраструктури та фактичним станом об'єктів у середовищі виконання. Наприклад, хмарна роль доступу, сервісний акаунт Kubernetes, контейнерне навантаження, Security Group та вразливий образ можуть належати до різних рівнів абстракції, але з погляду безпеки формувати єдиний причинно-наслідковий ланцюг. Тому актуальним є застосування графової моделі, здатної поєднувати ці сутності в єдиному просторі аналізу [1-4].

Метою роботи є проєктування підходу до моделювання шляхів атак у гібридних хмарних середовищах на основі Security Graph, а також обґрунтування архітектурних та алгоритмічних рішень для аналізу досяжності, пошуку пріоритетних маршрутів і динамічного оновлення стану графа.

Результати дослідження

Для реалізації підходу доцільно використовувати Middleware-рішення, яке виконує роль проміжного шару між API хмарних платформ, API середовищ оркестрації та аналітичними компонентами. На відміну від повноцінних графових СУБД, легковагова in-memory модель дозволяє зменшити інфраструктурні накладні витрати й забезпечити швидке виконання алгоритмів пошуку шляхів та метрик центральності. Такий підхід узгоджується з потребою оперативного аналізу великої кількості взаємозв'язків між об'єктами інфраструктури.

Архітектуру системи доцільно поділити на три функціональні площини. Площина збору даних містить колектори для хмарної інвентаризації, середовища виконання та метаданих про вразливості. Графовий рушій і модуль “зшивання” перетворюють нормалізовані дані на орієнтований мультиграф, встановлюючи зв’язки між об’єктами різних рівнів. Аналітичний модуль Risk Engine застосовує алгоритми обходу та ранжування для виявлення attack paths і передає результати до інтерфейсу візуалізації або API [1, 2, 5].

Інформаційну модель Security Graph запропоновано подавати як типізований орієнтований мультиграф $G = (V, E, \tau V, \tau E, P)$, де V є множиною вузлів, E - множиною ребер, τV та τE визначають типи вузлів і ребер, а P містить властивості об’єктів та зв’язків. Така модель дозволяє одночасно відображати кілька різних відношень між одними й тими самими сутностями, наприклад мережевий доступ, розміщення контейнера на вузлі та делегування ролі доступу [6].

До основних типів вузлів графа належать обчислювальні активи, ідентичності, мережеві абстракції, ресурси даних та сутності ризику. Обчислювальні активи можуть включати віртуальні екземпляри, поди Kubernetes і безсерверні функції. Ідентичності відображають IAM-ролі, користувачів і сервісні акаунти. Мережеві абстракції описують підмережі, Security Groups і узагальнений вузол Internet. Ресурси даних можуть бути представлені сховищами об’єктів, базами даних або реєстрами контейнерів. До сутностей ризику належать CVE, секрети та помилкові конфігурації [7-12].

Ребра графа описують змістові відношення між вузлами. Зокрема, ALLOW_INBOUND відображає дозволений мережевий вхідний трафік, CAN_ASSUME_ROLE - можливість прийняття ролі, RUNS_ON - розміщення навантаження на обчислювальному ресурсі, HAS_VULNERABILITY - зв’язок активу з відомою вразливістю, ACCESS_TO_DATA - доступ ідентичності до ресурсу даних. Завдяки такій онтології підграф може інтерпретуватися як шлях атаки, якщо він з’єднує зовнішнє джерело впливу з критичним активом через допустимі переходи доступу, розміщення, делегування або компрометації.

Базовим алгоритмом для аналізу досяжності є пошук у ширину (BFS), який дозволяє визначити множину вузлів, потенційно досяжних з початкової точки компрометації. У задачах безпеки така множина інтерпретується як blast radius - область можливого поширення впливу після компрометації вихідного вузла. Для пошуку пріоритетних маршрутів доцільно застосовувати алгоритм Дейкстри у зваженому графі, де вага ребра відображає евристичну складність переходу між вузлами. Менша вага може надаватися переходам, пов’язаним з відкритим мережевим доступом, критичною CVE або доступним механізмом делегування привілеїв [3, 13].

Окреме значення мають метрики структурної важливості. Betweenness Centrality дає змогу виявити критичні проміжні вузли, через які проходить значна кількість найкоротших шляхів, а PageRank може використовуватися для ранжування структурно впливових активів. У контексті захисту це дозволяє визначати вузли, посилення контролю над якими найсильніше скорочує кількість потенційних маршрутів атаки [14, 15].

Для підтримання актуальності графа безпеки у динамічному середовищі необхідно застосовувати інкрементальне оновлення. У Kubernetes це може здійснюватися через механізми Watch API, які дозволяють отримувати потік подій про створення, зміну або видалення об’єктів [16]. Для IaaS-частини доцільним є гібридний підхід: події журналювання API-викликів фіксуються сервісами аудиту, маршрутизуються через шину подій і надходять до черг повідомлень для подальшої асинхронної обробки [17-20]. Такий підхід дозволяє оновлювати лише змінені вузли та ребра, не виконуючи повного повторного сканування інфраструктури.

Таблиця 1 - Узагальнення алгоритмічного забезпечення Security Graph

Алгоритм або підхід	Основне призначення	Очікуваний результат
BFS	Аналіз досяжності від стартового вузла	Множина потенційно доступних активів
Dijkstra	Пошук маршруту з мінімальною евристичною вартістю	Пріоритетний шлях атаки
Betweenness Centrality	Виявлення критичних проміжних вузлів	Рейтинг вузлів-посередників
PageRank	Оцінювання структурної важливості вузлів	Рейтинг впливових активів
Подієве оновлення	Інкрементальна синхронізація стану графа	Актуальна модель без повного сканування

Висновки

У результаті дослідження обґрунтовано доцільність використання Security Graph для моделювання шляхів атак у гібридних хмарних середовищах. Запропонована модель дозволяє об’єднати в одному

графі обчислювальні активи, ідентичності, мережеві об’єкти, ресурси даних і сутності ризику, що створює основу для формального аналізу небезпечних комбінацій конфігурацій.

Запропонована Middleware-архітектура з поділом на площину збору даних, графовий рушій і аналітичний модуль забезпечує логічне розмежування відповідальностей та підтримує розширюваність системи. Використання in-memory обробки графа є доцільним для легковагового рішення, орієнтованого на швидке виконання алгоритмів аналізу досяжності та пошуку attack paths.

Алгоритмічний базис системи доцільно формувати на основі BFS, алгоритму Дейкстри, метрик центральності та структурного ранжування вузлів. Поєднання цих методів дозволяє не лише знаходити потенційні шляхи атаки, а й визначати критичні проміжні вузли, що мають найбільший вплив на безпеку інфраструктури.

Застосування подієво-орієнтованого інкрементального оновлення через Watch API, журнали аудиту, шини подій і черги повідомлень дає змогу підтримувати граф безпеки в актуальному стані. Це є важливою передумовою для своєчасного виявлення нових ризикових зв’язків у хмарно-нативних середовищах.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Boto3 documentation [Електронний ресурс]. - Режим доступу: https://docs.aws.amazon.com/boto3/latest/
2. Kubernetes Python Client [Електронний ресурс]. - Режим доступу: https://github.com/kubernetes-client/python
3. NetworkX shortest paths documentation [Електронний ресурс]. - Режим доступу: https://networkx.org/documentation/stable/reference/algorithms/shortest_paths.html
4. Streamlit documentation [Електронний ресурс]. - Режим доступу: https://docs.streamlit.io
5. PyVis documentation [Електронний ресурс]. - Режим доступу: https://pyvis.readthedocs.io/en/latest
6. NetworkX MultiDiGraph documentation [Електронний ресурс]. - Режим доступу: https://networkx.org/documentation/stable/reference/classes/multidigraph.html
7. IAM best practices [Електронний ресурс]. - Режим доступу: https://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html
8. Amazon EKS: Associate IAM roles with Kubernetes service accounts [Електронний ресурс]. - Режим доступу: https://docs.aws.amazon.com/eks/latest/userguide/associate-service-account-role.html
9. AWS IAM roles documentation [Електронний ресурс]. - Режим доступу: https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles.html
10. AWS STS AssumeRole API Reference [Електронний ресурс]. - Режим доступу: https://docs.aws.amazon.com/STS/latest/APIReference/API_AssumeRole.html
11. AWS VPC subnets documentation [Електронний ресурс]. - Режим доступу: https://docs.aws.amazon.com/vpc/latest/userguide/configure-subnets.html
12. Amazon EC2 security groups documentation [Електронний ресурс]. - Режим доступу: https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-security-groups.html
13. FIRST. Common Vulnerability Scoring System v4.0 Specification [Електронний ресурс]. - Режим доступу: https://www.first.org/cvss/v4-0/cvss-v40-specification.pdf
14. NetworkX centrality algorithms documentation [Електронний ресурс]. - Режим доступу: https://networkx.org/documentation/stable/reference/algorithms/centrality.html
15. NetworkX PageRank documentation [Електронний ресурс]. - Режим доступу: https://networkx.org/documentation/stable/reference/algorithms/generated/networkx.algorithms.link_analysis.pagerank_alg.pagerank.html
16. Kubernetes API concepts [Електронний ресурс]. - Режим доступу: https://kubernetes.io/docs/reference/using-api/api-concepts/
17. AWS CloudTrail events documentation [Електронний ресурс]. - Режим доступу: https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudtrail-events.html
18. Amazon EventBridge CloudTrail events [Електронний ресурс]. - Режим доступу: https://docs.aws.amazon.com/eventbridge/latest/userguide/eb-service-event-cloudtrail.html
19. Amazon EventBridge event patterns [Електронний ресурс]. - Режим доступу: https://docs.aws.amazon.com/eventbridge/latest/userguide/eb-event-patterns.html
20. Amazon EventBridge Pipes with SQS [Електронний ресурс]. - Режим доступу: https://docs.aws.amazon.com/eventbridge/latest/userguide/eb-pipes-sqs.html

Скаліуш Дмитро Миколайович - студент групи 2КІТС-22б, факультет менеджменту та інформаційної безпеки, Вінницький національний технічний університет, м. Вінниця, e-mail: skaliush3@gmail.com

Безпалий Кирило Валерійович - асистент кафедри менеджменту та безпеки інформаційних систем, Вінницький національний технічний університет, м. Вінниця, e-mail: kyrylo.bezpalyi@vntu.edu.ua

Skaliush Dmytro M. - student of group 2KITS-22b, Faculty of Management and Information Security, Vinnytsia National Technical University, Vinnytsia, e-mail: skaliush3@gmail.com

Bezpalyi Kyrylo V. - Assistant of the Department of Management and Security of Information Systems, Vinnytsia National Technical University, Vinnytsia, e-mail: kyrylo.bezpalyi@vntu.edu.ua