УДК 004.7:519.711

В. Ю. Зубок
Інститут проблем моделювання в енергетиці ім. Г. Є. Пухова
вул. Генерала Наумова, 15, 03164 Київ, Україна
тел. (+38044) 4241063, e-mail: vitaly.zubok@gmail.com

Факторний аналіз ризиків на прикладі інциденту з програмним забезпеченням реєстру глобальної маршрутизації

При проектуванні та розробці програмних систем будь-якої складності важливим і необхідним є управління проектним ризиком. Методологія базується на аналізі загроз, реалізація яких може певним чином вплинути на систему та її власника. Впровадження нових технологій RPKI призвело до появи нової єдиної точки відмови в системі глобальної маршрутизації мережі Інтернет. Питанню поводження з ризиками в процесі розробки та експлуатації програмного забезпечення для реєстрів глобальної маршрутизації було приділено недостатньо уваги, в результаті чого стався глобальний інцидент безпеки, який класифікується як «перехоплення маршрутів». Проведено аналіз помилок ризик-менеджменту методом декомпозиції основного ризику та подальшого факторного аналізу.

Ключові слова: *управління ризиками, глобальна маршрутизація, безпека програмного забезпечення, перехоплення маршрутів, кібербезпека.*

Вступ

Інтернет порівнюють зі складною екосистемою з переплетеними зв'язками, в якій втручання в будь-який зв'язок матиме непрогнозовані наслідки на інших рівнях системи. Розвиток Інтернету, а також його трансформації обумовлюються мільйонами чинників, а спирається мережа на систему глобальної маршрутизації, яка, в свою чергу, забезпечується протоколом маршрутизації BGP-4. Протокол надзвичайно ефективний і масштабований, проте в силу того, що він розроблявся понад 30 років тому, має невиліковні вади інформаційної безпеки [1]. Головною вадою є відсутність механізмів верифікації маршруту та валідації джерела, що пропонує маршрут. Тому кожного року трапляється декілька глобальних інцидентів через викрадення та витік маршрутів (route hijack та route leak), які є однією з масштабних проблем кібербезпеки. Викрадення маршруту, чи викрадення префіксу — це явище, при якому Автономна система (AS) нелегітимно оголошує себе як

джерело маршруту (route origin) замість справжнього джерела. Витік маршруту означає, що AS нелегітимна, з порушенням політики маршрутизації, пропонує маршрути до чужих префіксів. Ці нелегітимні маршрути забруднюють таблиці маршрутизації BGP, спотворюють шляхи проходження мережевого трафіка та впливають на конфіденційність, цілісність і доступність IP-комунікацій. Інколи вони пояснюються помилками конфігурації маршрутизаторів, але значна міра таких інцидентів визнається цілеспрямованою атакою. Такі атаки використовуються для маніпуляцій із трафіком з метою дестабілізації телекомунікаційної мережі Інтернет, перехоплення трафіка, шпигунства, крадіжок даних, нанесення матеріальної шкоди, дезінформації тощо [2].

В оригінальній специфікації BGP не було визначено заходів безпеки для запобігання навмисних чи ненавмисних помилок конфігурації мережі. Відсутність вбудованих заходів безпеки робить BGP вразливим як до викрадення префікса, так і до витоків маршруту. Тому приблизно 25 років тому були введені в експлуатацію кілька баз даних Інтернет-маршрутів (Internet Routing Registry — IRR), за якими «доглядають» п’ять авторизованих регіональних реєстрів, що уповноважені реєструвати розподіл IP-адрес і номерів автономних систем серед інтернет-провайдерів [3]. Провайдери, налаштовуючи на маршрутизаторах політику маршрутизації (які маршрути від кого приймати) покладаються на публічно доступні реєстраційні дані, які є IRR. Протягом 10 років повільно розвивається новий додатковий механізм — електроні сертифікати для адресного ресурсу, що дозволяє валідацію джерел анонсів. RPKI вирішує питання авторизації внесення змін до IRR та валідації інформації про «належність» мережевих префіксів. Об’єктом валідації є запис про джерело маршруту (Route Origin Authority — ROA) [4]. Але допоки глобальна маршрутизація була розподіленою задачею без наявного «центрального маршрутизатора», що авторизує всі шляхи. І саме технологія валідації джерел створила єдину точку відмови (чи, точніше, п’ять точок — за кількістю регіональних реєстрів), впливом на яку можна зашкодити всій системі. Так, нещодавній локальний інцидент у адміністратора реєстру призвів до масштабного збою в глобальній маршрутизації.

Хронологія інциденту безпеки

31 березня 2020 року в ході роботи з програмним забезпеченням реєстру з бази даних випадково видалили 4100 записів ROA. Адміністратор європейського реєстру повідомив у середині дня 1 квітня 2020, що «це сталося під час технічного обслуговування нашого внутрішнього програмного забезпечення». Більш детальну інформацію було опубліковано у пост-фактумному звіті, з якого стали зрозумілі тривалість і глобальність збоїв, що виникли в результаті проблем підчас оновлення програмного забезпечення [5].

Оновлення програмного забезпечення та видалення записів ROA трапилось у неробочий час, що призвело до невчасного детектування проблеми. Повідомлення постраждалих клієнтів були оброблені вже наступного ранку, 1 квітня 2020 року. Відновлення видалених записів не вдалося без втручання наших інженерів і загалом тривало до середини дня 2 квітня. Після того проводилося розслідування, чи страждали якісь IP-префікси підчас збою від витоку чи перехоплення маршруту.

Помилкове видалення записів, випадково чи ні, співпало з іншою проблемою, масштаб якої описаний спеціалістами компанії QRATOR [6]. Можливо через інший збій програмного забезпечення, але в той самий час, коли відновлювали БД з ROA, 8877 маршрутів від 200 автономних систем були неправомірно анонсовані державним російським провайдером Ростелеком. Масштаб був би менший, проте неправомірність анонсів неможливо було встановити через відсутність сертифікатів походження маршруту, що були видалені. Понад годину були недосяжні великі сегменти сервісів Hetzner, Amazon AWS, Akamai, Cloudflare, Digital Ocean, і це набуло розголосу [7].

Викладені матеріли інциденту дають привід вважати, що з точки зору безпеки глобальної маршрутизації в Інтернеті, локальне програмне забезпечення інтернет-реєстру на сьогодні є новою єдиною точкою відмови (single point of failure — SPoF), саме завдяки розвитку механізму RPKI, що призваний захистити глобальну маршрутизацію. Це означає, що мова йде про критичний характер цього програмного забезпечення, а на етапі його розробки, впровадження та оновлення були допущені помилки.

Декомпозиція факторів ризику, що призвели до інциденту

Забезпечення захищеності таких критичних програмних систем є комплексною задачею та потребує уваги в процесі розробки, впровадження та експлуатації (включаючи оновлення). Розробка та впровадження програмного забезпечення — це діяльність, яка використовує різноманітні технологічні досягнення і вимагає високого рівня знань. Через ці та інші фактори кожен проєкт з розробки, впровадження та експлуатації програмного забезпечення містить елементи невизначеності. Це відомо як проєктний ризик. Управління ризиками включає такі завдання [8]:

1) визначення ризиків і їхніх тригерів, або факторів, що призводять до настання ризику;

2) класифікування та визначення пріоритетів ризиків;

3) розробка плану з усунення чи мінімізації наслідків кожного ризику;

4) моніторинг стану тригерів ризику в ході проєкта;

5) у разі матеріалізації ризику — виконання плану з усунення чи мінімізації наслідків.

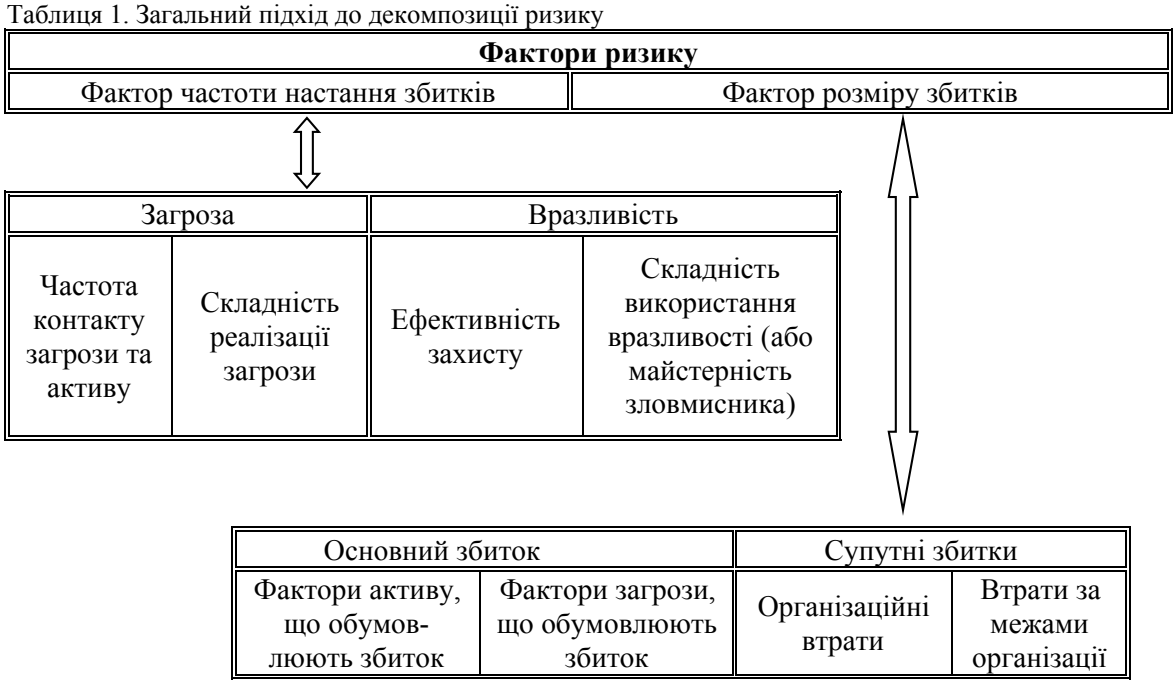
Один лише факт того, що інцидент тривав майже 2 доби і набув планетарного масштабу, свідчить про те, що при плануванні та (або) виконанні перелічених вище заходів було допущено суттєвих помилок. Розберемо їх.

Проблема з базою даних стала зрозумілою на ранок 1 квітня, проте інцидент не було усунено, і масштабне перехоплення маршрутів завдяки цій проблемі відбулося ввечері 1 квітня. Вочевидь, *при аналізі факторів ризику* та *визначенні пріоритетів були допущені помилки*.

Більшість проектів програмної інженерії за своєю суттю ризиковані через різноманітність потенційних джерел. Досвід інших проектів програмного забезпечення може допомогти менеджерам класифікувати ризик шляхом якомога точнішого визначення та опису всіх реальних загроз. Є велика кількість класифікації, ранжування та оцінки ризиків. Для аналізу ризиків даного проєкта застосуємо ме-

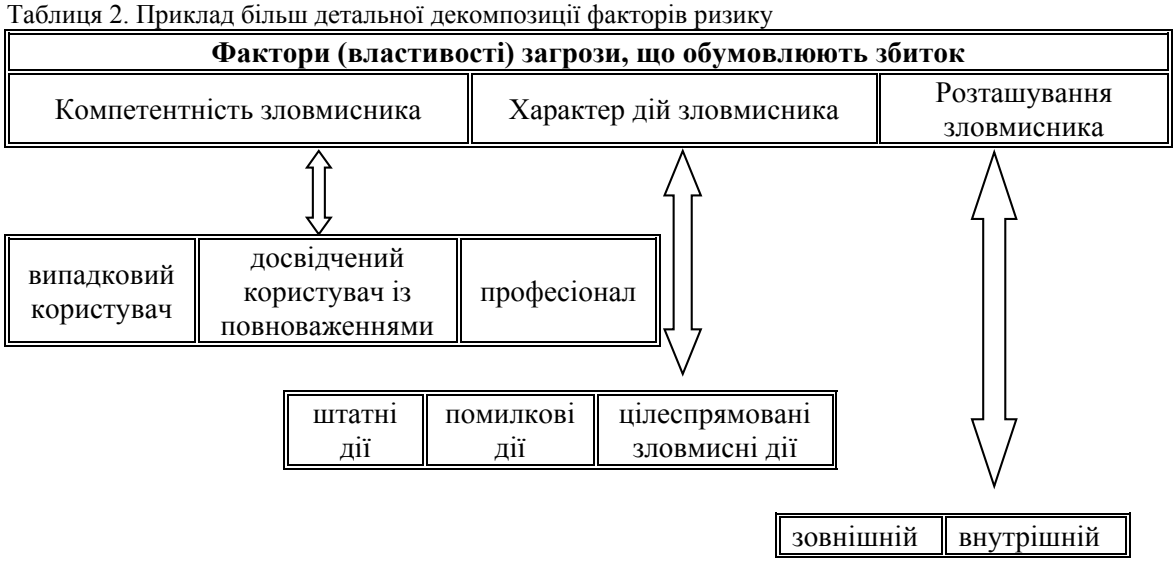
тод декомпозиції, де кожен фактор ризику розкладається на фактори більш низького рівня. Розкладання (drill-down) факторів на більш дрібні проводиться, допоки не з'являється можливість надати факторам кількісну оцінку [9]. Формальний підхід до декомпозиції ризику наведений у табл. 1.

Таблиця 1. Загальний підхід до декомпозиції ризику



Кожен із факторів нижнього рядка таблиці або може бути охарактеризований кількісно, або має бути декомпонований на ще дрібніші фактори. Наприклад, фактори (властивості) загрози, що обумовлюють збиток, декомпонуються наступним чином (табл. 2).

Таблиця 2. Приклад більш детальної декомпозиції факторів ризику



Очевидним є те, що сукупність факторів «випадковий користувач, який виконує штатні дії ззовні» має найменший вплив на ризик, порівняно із сукупністю «професіонал, який виконує цілеспрямовані зловмисні дії зсередини».

Проведемо декомпозиційний аналіз факторів ризику інциденту з реєстром маршрутів відповідно до формальної моделі. Наголошуємо, що ризик-аналіз мав робитися до настання інциденту на етапі планування дій з реєстром, які стали тригером (подією), що призводить до настання ризику. В даному випадку тригером було оновлення програмного забезпечення. Спочатку роздивимося ланцюг факторів, починаючи з тригеру.

1. Тригер: оновлення ПЗ.
2. Фактор ризику: втрата даних.
3. Максимальний збиток: високий; аргументи надані в ході опису інциденту.
4. Частота настання загрози: низька, оскільки оновлення ПЗ є рутинною процедурою, але рідкісною, порівняно з іншими операціями, що супроводжують процес експлуатації ПЗ.
5. Складність реалізації загрози: середня; помилки в програмному коді чи процесі оновлення з'являються з вини персоналу. Не слід переоцінювати рівень розробників, тестувальників та інженерів dev-ops (інженери розробки та операційної підтримки), які задіяні в життєвому циклі ПЗ.
6. Заходи захисту (захищеність): пропонується додаткова декомпозиція захищеності і оцінка дрібніших факторів ризику відповідно до табл. 3. Стовпчики захисту конфіденційності не заповнені через те, що в даному інциденті мала місце проблема з публічними даними.

Таблиця 3. Декомпозиція факторів захищеності

Захищеність							
Цілісність			Доступність			Конфіденційність	
моніторинг цілісності	резервне копіювання	процедури відновлення даних	моніторинг доступності	наявність дублюючої системи (резерву)	процедури відкату	–	–

Декомпозиція факторів втрат, особливо аналіз втрат внутрішніх і втрат за межами організації, мав би дати важливі результати, але в цій статті акцентуємо увагу на факторах, що мають впливати на настання ризику. Отже, на етапі декомпозиції захищеності власнику системи має бути зрозуміло, які фактори впливають на ймовірність настання ризику для активу.

Аналіз помилок ризик-менеджменту, що супроводжували інцидент, та можливі заходи до запобігання

Інцидент розпочався підчас проведення планових робіт з оновлення, а виявлено проблему було лише на ранок. Це означає, що мав місце *брак моніторингу* факторів ризику. Моніторинг має включати:

— публікацію звітів про стан проєкту, включно з питаннями управління ризиками;

— перегляд планів ризику відповідно до будь-яких основних змін у графіку проєкту;

— перегляд і репріоритізація ризиків;

— мозковий штурм щодо потенційно нових ризиків підчас непланованих змін у проєкті.

Протягом проєкту важливо забезпечити ефективну комунікацію між усіма зацікавленими сторонами, менеджерами, розробниками, тестувальниками, інженерами dev-ops, користувачами. Проте, перші скарги реєстр отримав не від власного персоналу, а від зовнішніх користувачів, і отримав «у неробочий час» відповідно до опублікованого пост-аналізу, тому dev-ops дізналися про наявність проблеми тільки наступного робочого дня. Служба підтримки не вважала скарги користувачів важливими, бо не знала про оновлення ПЗ і не була готова до посиленого моніторингу. Очевидним є *брак комунікацій*.

Якщо виникає ризик, відповідна реакція на пом'якшення наслідків повинна бути взята з уже підготовленого плану управління ризиками. Наприклад, доступні такі варіанти пом'якшення:

— «прийняти»: визнати, що ризик впливає на проєкт; це означає погодитися з можливими наслідками (таке рішення припустиме лише, якщо інші міри з пом'якшення коштуватимуть дорожче, ніж наслідки);

— «уникнути керувати наслідками»: вживання заходів для мінімізації впливу або зменшення інтенсифікації ризику;

— «передача ризику»: здійснити організаційну зміну підзвітності, відповідальності чи повноважень іншим зацікавленим сторонам, які приймуть ризик;

— «продовжити моніторинг»: часто підходить для ризиків з низьким впливом; можливо, цей було обрано в даному випадку.

Можна аргументовано показати, що в даному інциденті мала місце *відсутність чи недосконалість плану пом'якшення ризику*. Оскільки, як показано раніше, попередній ризик-аналіз був проведений недостатньо, власник системи не прийняв найкращого рішення — уникнути ризику. В даному випадку, принаймні, не проводити оновлення ПЗ реєстру маршрутизації у вечірні та нічні години, якщо вночі нікому буде спостерігати за оновленою системою.

Також є обґрунтованою думка, що не відбулося вчасної передачі ризику: черговий персонал побачив скарги користувачів, та не насмілився потурбувати більш компетентні підрозділи, оскільки то було пізно ввечері.

Окрім того, персонал реєстру не був готовий до швидкого відновлення даних. Попри значний вплив інциденту, його наслідки було усунуто більше ніж як за добу: реальне відновлення помилково видалених записів почалося на ранок 2 квітня.

Висновки

Питання поводження з ризиками в процесі розробки та експлуатації програмного забезпечення для реєстрів глобальної маршрутизації в Інтернеті потребує значно більшої уваги через те, що впровадження нових технологій електрон-

них сертифікатів джерела маршруту призвело до появи нової єдиної точки відмови у системі глобальної маршрутизації.

Помилки в поводженні з проєктним ризиком у процесі розробки та оновлення ПЗ європейської БД реєстру маршрутизації призвели до того, що інцидент з перехопленням маршрутів Ростелекомом 1 квітня 2020 р. набув глобального масштабу. Факторний аналіз показав, що головними проблемами безпеки циклу розробки і експлуатації програмного забезпечення стали недостатність моніторингу та відсутність чи недостатність програми пом'якшення наслідків ризику.

1. Andreas Reuter, Randy Bush, Italo Cunha, Ethan Katz-Bassett, Thomas C Schmidt, and Matthias Wählisch. Towards a rigorous methodology for measuring adoption of RPKI route validation and fltering. ACM SIGCOMM Computer Communication Review, 48(1):19–27, 2018.

2. Зубок В. Ретроспективний аналіз інцидентів кібербезпеки, пов'язаних з атаками на глобальну маршрутизацію. *Моделювання та інформаційні технології*. 2019. Вип. 86. С. 42–49.

3. MERIT. List of Routing Registries. URL: http://www.irr.net/docs/list.html (Дата звернення: 20.04.2020).

4. RIPE NCC. BGP Origin Validation. URL: https://www.ripe.net/manage-ips-and-asns/resource-management/certification/bgp-origin-validation (Дата звернення: 20.04.2020).

5. RPKI ROA Deletion: Post-mortem. URL: https://www.ripe.net/ripe/mail/archives/routing-wg/2020-April/004072.html (Дата звернення: 21.04.2020).

6. This is how you deal with route leaks. Qrator Labs corporate blog. Information Security, Network technologies. URL: https://habr.com/en/company/qrator/blog/495260/ (Дата звернення: 21.04.2020).

7. Russian Telco Hijacked Internet Traffic of Major Networks — Accident or Malicious Action?: URL: https://www.securityweek.com/russian-telco-hijacked-internet-traffic-major-networks-accident-or-malicious-action (Дата звернення: 21.04.2020).

8. NASA-GB-9719.13: NASA Software Safety Guidebook. NASA Technical Standard. — Washington D.C. National Aeronautics and Space Administration, 2004.

9. Measuring and Managing Information Risk: A FAIR Approach. URL: https://www.fairinstitute.org/ fair-book (Дата звернення: 10.06.2020).

Надійшла до редакції 25.04.2020