Engineer

Аварійне відновлення

повернення сервісу після аварії за перевіреним планом

Що це таке

Аварійне відновлення (disaster recovery) — частина планування на випадок надзвичайних ситуацій, що охоплює найгірший сценарій: місце, де працює система, втрачено або воно надовго недосяжне, і систему доводиться відбудовувати деінде з копій. Посібник NIST з планування на випадок надзвичайних ситуацій для федеральних інформаційних систем США, SP 800-34 Revision 1 від 2010 року, визначає план аварійного відновлення як письмовий план відновлення однієї чи кількох інформаційних систем на альтернативному об’єкті після серйозного збою обладнання чи програмного забезпечення або знищення приміщень. Сфера в нього вузька. Посібник залишає такий план для великих, зазвичай фізичних, порушень, які надовго відрізають доступ до основного об’єкта, і охоплює ним лише ті порушення, що вимагають переїзду.

Саме ця вузькість відрізняє його від сусідніх понять. План на випадок надзвичайних ситуацій для інформаційної системи, у термінах посібника information system contingency plan, відновлює одну систему там, де вона стоїть, скажімо, після відмови диска чи пошкодження бази даних. План аварійного відновлення прив’язаний до місця: він переносить роботу на альтернативний майданчик, а вже там кожну систему піднімають за її власним планом. План для однієї бази даних і план на випадок втрати цілого дата-центру, хмарного регіону чи облікового запису в провайдера — це різні документи з різними власниками, і хто їх плутає, той зазвичай лишається без жодного з них.

Три показники перетворюють план із наміру на те, що можна перевірити. Максимально допустимий час простою (maximum tolerable downtime, MTD) — загальна тривалість збою, яку власник системи готовий прийняти. Цільовий час відновлення (recovery time objective, RTO) — найдовший час, протягом якого системний ресурс може бути недоступний, перш ніж шкода

для інших систем і для бізнесу стане неприйнятною; а оскільки перервану роботу потім ще треба обробити заново, посібник зазначає, що RTO зазвичай має бути коротшим за MTD. Цільова точка відновлення (recovery point objective, RPO) дивиться назад: це момент до збою, на який можна відновити дані з найсвіжішої резервної копії, і вона показує, скільки втрачених даних бізнес може пережити. Простіше кажучи, RTO — це скільки ви лежите, а RPO — скільки роботи втрачаєте.

Альтернативний майданчик буває різного ступеня готовності. Холодний (cold site) дає лише приміщення та інфраструктуру, теплий (warm site) обладнаний частково, а гарячий (hot site) обладнаний повністю й має персонал. Посібник додає мобільні майданчики, які в багатьох випадках можна доставити протягом 24 годин, перш ніж почнеться розгортання, і дзеркальні — найдорожчий варіант, якому посібник приписує практично 100% доступності. Холодний майданчик найдешевше утримувати, але на його оснащення може піти чимало часу. Сьогодні для більшості програмного забезпечення ту саму роль відіграє другий хмарний регіон або другий провайдер, а компроміс між тим, що стоїть напоготові, і тим, скільки коштує тримати його напоготові, лишається тим самим.

Застосуйте поняття до рішення

Застосуйте «Що таке аварійне відновлення?» до конкретного рішення. Ми можемо пов'язати поняття із системою, яку ви створюєте або купуєте. Резервні копії, які перевірено, а не просто вважають робочими, і записаний план на той ранок, коли сервер не ввімкнеться.

Переглянути наші послуги Зв'язатися з нами →

Перевірте припущення за терміном

Використайте пояснення та пов'язані роботи, щоб сформулювати запитання для проєкту. Ми можемо перевірити припущення разом до вибору інструмента. Автоматизувала 100 критичних резервних копій — 100 критичних резервних копій автоматизовано в усіх базах

Автоматизувала 100 критичних резервних копій Зв'язатися з нами →

Чому це важливо

Резервна копія — це лише копія, а копія сама по собі нічого не запускає. Щоб від копії дійти до робочої системи, потрібні ще програми, які її прочитають, ключі, які її розшифрують, сервери й мережа, на яких усе працюватиме, конфігурація, що зв’язує це докупи, і людина, яка знає порядок дій. Усе з цього, що існувало лише на втраченому майданчику, зникло разом із ним. Аварійне відновлення — це дисципліна, яка дбає, щоб усе це пережило ту саму подію і щоб шлях від копії до сервісу хтось пройшов задовго до того, як він знадобиться.

Дані, які можна відновити, не новіші за останню придатну для відновлення копію, хоч би якого RPO мав дотримуватися розклад резервного копіювання. Якщо нічне завдання тиждень мовчки падало, під загрозою опиняються дані за цілий тиждень, хоч би що було написано в політиці. Посібник NIST каже те саме про стрічкові носії: резервні копії слід регулярно перевіряти, щоб переконатися, що дані збережено правильно і файли можна дістати без помилок і втрат. Зелений індикатор біля завдання резервного копіювання доводить, що щось записалося. Лише відновлення доводить, що це можна прочитати назад.

План теж старіє. Люди звільняються, сервери замінюють, провайдери переробляють свої консолі, а паролі в інструкції з відновлення втрачають чинність. Із семи кроків, які NIST описує для планування на випадок надзвичайних ситуацій, останні два — тестування, навчання й тренування та підтримка плану; посібник хоче, щоб план був живим документом, який переглядають щонайменше раз на рік і щоразу, коли система чи ресурси для відновлення суттєво змінюються. План, який ніколи не відпрацьовували, зазвичай ламається на дрібницях, які нікому не спало на думку записати: порядок запуску сервісів, DNS-запис із TTL у добу, той єдиний ключ доступу, що зберігався на машині, якої більше немає.

Для команд, що пишуть програмне забезпечення, катастрофа — це часто обліковий запис, а не будівля. Видалений хмарний проєкт, зламаний вхід адміністратора, заблокований акаунт у провайдера чи руйнівна команда, виконана на продакшені, відрізають доступ до системи не гірше за повінь. Сервери в режимі очікування (standby) від останнього не захищають: такий сервер застосує помилковий `DROP TABLE` за лічені миті — так само, як застосовує будь-яку іншу зміну.

Як це виглядає на практиці

Для бази даних PostgreSQL RPO, набагато коротший за інтервал між двома резервними копіями, дає безперервне архівування журналу попереднього запису (write-ahead log, WAL) поверх базової резервної копії. Сервер копіює кожен заповнений сегмент WAL у безпечне місце, а під час відновлення ці сегменти програються поверх базової копії аж до будь-якого обраного моменту після того, як її зробили:

```
# postgresql.conf на захищеному сервері
wal_level = replica
archive_mode = on
archive_command = 'test ! -f /mnt/archive/%f && cp %p /mnt/archive/%f'
```

Щоб відновитися, базову резервну копію розпаковують у новий каталог даних, кладуть туди порожній файл `recovery.signal`, а параметри відновлення вказують, де лежить архів і де зупинитися:

```
restore_command = 'cp /mnt/archive/%f %p'
recovery_target_time = '2026-03-10 17:14:00+01'
recovery_target_action = 'pause'
```

Документація PostgreSQL розповідає історію, заради якої ці параметри існують: хтось видаляє критичну таблицю о 17:15 у вівторок увечері, помічає це в середу опівдні й відновлює стан на 17:14. Досягнувши цілі, сервер за замовчуванням стає на паузу, щоб запитами можна було переконатися, що момент правильний, перш ніж сервер стане основним і почне приймати з’єднання; якщо момент не той, сервер зупиняють, задають пізнішу ціль і запускають знову. Точка зупинки має лежати після завершення базової резервної копії, а документація радить налаштувати й перевірити архівування ще до того, як зроблять першу базову копію. Коли відновлення завершується, сервер видаляє `recovery.signal`, щоб випадково не перейти в режим відновлення знову.

Engineer ApS надає резервне копіювання та відновлення як послугу: резервні копії, перевірені на ділі, а не на віру, і письмовий план на той ранок, коли сервер не підніметься. Що це означає, показують дві сторінки портфоліо. У проєкті резервних копій і відновлення як коду 2025 року термін зберігання резервних копій налаштовано в Вісер поруч із базою даних, тож відновлення на довільний момент часу — властивість шаблону, а не налаштування, яке колись хтось виставив у порталі, а середовище навколо бази теж описано як код, бо базу, відновлену в середовище, яке відбудовували з пам’яті, навряд чи можна вважати відновленою. Потім готовність до відновлення перевірили й задокументували, чесно назвавши прогалини: географічне резервування вимкнено, а цільовий час відновлення лише запропоновано, а не виміряно, бо навчань не проводили.

Автоматизацію резервного копіювання та тестового відновлення компанії спроєктовано, щоб розширити можливості відновлення за межі знімка провайдера. Роль Ansible встановлює скрипти restic і юніти systemd для дампа бази даних, якщо його налаштовано, надсилання зашифрованих знімків із дедуплікацією до сховища іншого провайдера через SFTP, очищення за політикою зберігання та перевірок цілісності. Окремий щомісячний процес відновлює відомий файл у тимчасове місце й порівнює його байти. Резервне копіювання й відновлення мають окремі сигнали про успіх, парольну фразу читають із захищеного файла, а таймери залежать від налаштованого репозиторію. Ці засоби реалізовано; реальне відновлення є окремим результатом, який потрібно забезпечити. У записі про реалізацію від 2026-10-27 зазначено, що таймери були вимкнені за відсутності репозиторію; жодного завершеного реального відновлення не зафіксовано. Тож запис підтверджує роботу з автоматизації, а не зобов'язання щодо часу відновлення чи твердження, що компанія вже може безпечно втратити хост.

Де виникають проблеми

Резервну копію ні разу не відновлювали. Завдання, яке щоночі звітує про успіх, нічого не каже про те, чи можна його результат прочитати, розшифрувати й завантажити, а перше справжнє відновлення — поганий момент, щоб це з'ясовувати. Плануйте відновлення так само, як плануєте резервне копіювання: автоматична перевірка, що дістає відомі дані й звіряє їх, і повне відновлення в чисте середовище достатньо часто, щоб його тривалість була виміряним числом.

Копія поділяє долю оригіналу. Знімок у тому самому хмарному обліковому записі, диск із резервними копіями в тій самій стійці чи архівний бакет за тим самим входом адміністратора падають разом із тим, що вражає основну систему. Посібник NIST радить обирати віддалені сховища й альтернативні майданчики так, щоб та сама катастрофа навряд чи могла їх зачепити. Тримайте принаймні одну копію в іншого провайдера або в іншому обліковому записі з окремими обліковими даними, а ключі розшифрування зберігайте там, де вони не залежать від систем, які захищають.

Планували тільки дані. Відновити базу даних — мала частина повернення сервісу, якщо сервери, правила брандмауера, DNS, сертифікати й секрети навколо неї доведеться відтворювати з пам'яті. Опишіть середовище як інфраструктуру як код під контролем версій і зробіть його відбудову частиною тренування, а не лише завантаження даних.

Цілі взято зі стелі. RTO, вписаний у документ без жодного тренування, — це лише пропозиція, а RPO, який припускає, що кожна резервна копія вдалася, — оптимізм. Виміряйте обидва під час навчань. NIST розрізняє штабні навчання (tabletop), які зводяться до обговорення й не задіюють обладнання, і функціональні навчання, що імітують роботу, аж до повномасштабних, які охоплюють кожен елемент плану; цифри дають саме другі.

Сервер у режимі очікування вважають резервною копією. Такий сервер чи георепліковані база даних захищає від утрати обладнання, але копіює видалення, пошкодження й файли, зашифровані програмою-вимагачем, так само сумлінно, як і правильні записи. Тримайте копії стану на визначені моменти часу, яких помилка на основній системі не дістане, і зберігайте їх довше, ніж зазвичай минає до того, як помилку помічають.

План описує торішню систему. Інструкція, у якій значаться виведені з експлуатації сервери, люди, що вже звільнилися, і паролі, які більше не працюють, гальмує відновлення, замість того щоб його вести. Переглядайте план після кожної суттєвої зміни й після кожного тренування, записуйте, що тренування виявило, а частини, які змінюються найчастіше, як-от списки контактів, переглядайте ще частіше — так радить і NIST.

Що почитати далі

- SP 800-34 Rev. 1, Contingency Planning Guide for Federal Information Systems — повний посібник: як він визначає та окреслює DRP, сім кроків планування, MTD, RTO і RPO, альтернативні майданчики та види навчань, разом із шаблонами планів.
- Continuous Archiving and Point-in-Time Recovery (PITR) — розділ документації PostgreSQL 18 про архівування WAL, базові резервні копії й відновлення на обраний момент.
- Write Ahead Log — довідник параметрів `archive_command`, `restore_command` і налаштувань того, де має зупинитися відновлення.

Словник

Обирайте за фактичними умовами

Опишіть шлях розгортання, експлуатаційні ризики та чергову команду. Ми можемо визначити зміну з перевірками, шляхом відновлення та чіткою відповідальністю.

Дивитися, скільки це коштує Зв'язатися з нами →

Зробіть термінологію практичною

Запитайте, як «Що таке аварійне відновлення?» стосується вашої організації. Конкретного прикладу з поточної системи достатньо для початку розмови.

Переглянути наші послуги

welcome@engineer.company +45 50 10 61 84 Копенгаген, Данія

Візьміть цю сторінку із собою

Візитівка

Дбайливо зроблено з любов'ю — D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Авторське право © 2025 – дотепер · Engineer ApS · Усі права захищено.

Надруковано з https://engineer.company/uk/glossary/disaster-recovery/