Engineer

Багатофакторна автентифікація

вхід, що вимагає двох або більше різних видів доказів

Що це таке

Багатофакторна автентифікація, або скорочено MFA, — це вхід, який приймає заявлену особу лише тоді, коли її підтверджує більше ніж один вид доказу. NIST SP 800-63B-4 — федеральна настанова США щодо автентифікації, якої дотримується й багато організацій поза державним сектором. Вона називає три види: те, що ви знаєте, наприклад пароль; те, що ви маєте, наприклад телефон або апаратний ключ; і те, чим ви є, наприклад відбиток пальця. Визначення тримається на одному слові. Система має вимагати більше ніж один відмінний тип фактора, тож пароль і відповідь на контрольне запитання разом дають лише один фактор, бо обидва належать до того, що ви знаєте. Двофакторна автентифікація, або 2FA, — поширений окремий випадок рівно з двома факторами.

Настанова допускає два шляхи до цього. Перший поєднує окремі однофакторні автентифікатори: пароль плюс застосунок на телефоні, що показує одноразові коди, або список надрукованих резервних кодів. Другий — один багатофакторний автентифікатор, наприклад апаратний ключ чи телефон, що зберігає приватний ключ і користується ним лише після PIN-коду або відбитка пальця. У такому разі PIN-код або відбиток називають фактором активації. Його перевіряє сам пристрій, а сервер отримує лише результат, підпис, який може перевірити, тож один дотик пред’являє два фактори.

Біометрії тут відведено вузьке місце. NIST допускає відбиток пальця чи обличчя лише як частину багатофакторної автентифікації з фізичним автентифікатором, ніколи як окремий фактор, і вимагає, щоб завжди була альтернатива без біометрії. Настанова прямо каже, що біометричні характеристики не є таємницею: обличчя можна сфотографувати, а відбиток зняти з речі, якої торкався власник.

Застосуйте поняття до рішення

Застосуйте «Що таке багатофакторна автентифікація?» до конкретного рішення. Ми можемо пов'язати поняття із системою, яку ви створюєте або купуєте. Облікові записи, паролі й права налаштовані так, щоб доступ відповідав роботі, а звільнені втрачали його того ж дня.

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. Написала 1 000 правил RBAC для Linux-серверів — 1,000 правил доступу написано, щоб усунути вразливість

Написала 1 000 правил RBAC для Linux-серверів Зв'язатися з нами →

Чому це важливо

Пароль можна скопіювати так, що власник цього не помітить: його вводять на переконливій копії сторінки входу, вгадують або повторно використовують на сайті, чию базу даних згодом викрадають. MFA робить скопійований пароль недостатнім самим по собі, бо зловмисникові бракує ще чогось іншого виду.

NIST робить це базовою вимогою свого другого рівня гарантії. На AAL2 автентифікація має використовувати багатофакторний автентифікатор або комбінацію двох факторів, причому комбінація мусить містити фізичний автентифікатор разом із паролем або біометричним порівнянням. Верифікатор на цьому рівні мусить пропонувати принаймні один варіант, стійкий до фішингу, а федеральні відомства мають вимагати стійкої до фішингу автентифікації від своїх працівників, підрядників і партнерів.

Методи, що продаються під тими самими трьома літерами, відрізняються один від одного більше, ніж MFA відрізняється від самого пароля, і настанова чітко їх розмежовує:

- **Коди в SMS або голосовому дзвінку** — єдиний тип автентифікатора, який NIST позначає як обмежений. Ними ще можна користуватися, але верифікатор мусить запропонувати альтернативу без такої позначки, попередити користувача про ризик і мати план відмови від цього методу. Зміна зареєстрованого номера телефону вважається прив’язкою нового автентифікатора з усіма перевірками, які це передбачає.
- **Коди із застосунку та push-запити на телефоні** прийнятні на AAL2, але не стійкі до фішингу: те, що людина вводить або підтверджує, ніяк не прив’язане до сайту, який про це попросив.

- **Криптографічні автентифікатори, що прив’язують відповідь до сайту**, як-от ключі доступу (passkeys) і смарт-картки, стійкі до фішингу. AAL3 вимагає такого, чий приватний ключ неможливо експортувати з його апаратного забезпечення.

Електронної пошти в цьому переліку немає навмисно: настанова не допускає її як окремий канал для автентифікації поза основним з’єднанням (out-of-band).

Як це виглядає на практиці

Код у застосунку-автентифікаторі — найпоширеніший другий фактор, і RFC 6238 (травень 2011 року) описує його на кількох сторінках. Під час реєстрації сервер генерує секретний ключ і передає його застосунку, зазвичай як QR-код, тож обидві сторони мають той самий ключ. Щоб отримати код, кожна сторона ділить поточний час Unix на крок часу, типово 30 секунд, і бере цілу частину як лічильник. Далі вона обчислює HMAC цього лічильника з ключем і скорочує результат до короткого десяткового числа методом HOTP, який RFC 4226 визначив у 2005 році і який вимагає щонайменше шести цифр. У Python усе обчислення вміщується в кілька рядків:

```
import hashlib, hmac, struct

def totp(key: bytes, unix_time: int, digits: int = 6, step: int = 30) -> str:
    counter = struct.pack(">Q", unix_time // step)
    mac = hmac.new(key, counter, hashlib.sha1).digest()
    offset = mac[-1] & 0x0F
    code = struct.unpack(">I", mac[offset:offset + 4])[0] & 0x7FFFFFFF
    return str(code % 10**digits).zfill(digits)

print(totp(b"12345678901234567890", 59, digits=8))  # 94287082
```

Тестові вектори в RFC використовують цей 20-байтовий ключ ASCII і вісім цифр. На 59-й секунді від початку епохи Unix лічильник дорівнює 1, а варіант із SHA-1 дає 94287082, що й друкує функція; шестизначний код, який показав би застосунок, — це останні шість цифр, 287082. Обчислення ніяк не звертається до мережі, тому застосунок працює на телефоні в режимі польоту, і тому ж сервер мусить зберігати власну копію ключа кожного користувача.

Перевірка коду — те саме обчислення на сервері, а потім порівняння. RFC 6238 рекомендує приймати щонайбільше один попередній крок часу, щоб покрити затримку в мережі, і забороняє приймати код удруге після того, як його вже успішно використано.

Ключ доступу (passkey) замінює спільний ключ парою ключів. За Web Authentication Level 3, що є рекомендацією W3C з 25 серпня 2026 року, автентифікатор створює пару ключів для одного сайту, визначеного його доменом, і віддає серверові лише відкритий ключ. Під час входу браузер передає походження (origin), з яким він справді з'єднаний, а автентифікатор підписує виклик сервера разом із цим походженням, тож підпис, створений на схожому домені, на справжньому нічого не важить. Якщо для використання ключа потрібен PIN-код або відбиток пальця, ключ доступу стає багатофакторним автентифікатором за один крок. WebAuthn називає ключі доступу, які копіюються між пристроями однієї людини, multi-device credentials, і NIST приймає такі синхронізовані ключі на AAL2, але не на AAL3, бо ключ, який можна синхронізувати, за визначенням можна й експортувати.

Увімкнути MFA — найпростіша частина. Вирішити, хто мусить нею користуватися, які методи де зараховуються і що відбувається, коли хтось губить телефон, — це адміністрування облікових записів, частина ширшої організації доступу. Послуга з безпеки та керування доступом на цьому сайті описує саме таку організацію: облікові записи, паролі й права налаштовані так, щоб доступ відповідав роботі, а звільнені втрачали його того ж дня.

Де виникають проблеми

Фішингова сторінка пересилає код. Одноразовий код — це число, яке користувач читає і вводить, а сам код не несе відомостей про те, на який сайт дивився користувач. Схожа сторінка, що пересилає пароль і код на справжній сайт у межах того самого кроку часу, входить як цей користувач. Тому NIST прямо зазначає, що автентифікатор, вихід якого вводять вручну, ніколи не вважається стійким до фішингу, і додає, що неважливо, як користувач потрапив на підроблену сторінку. Виправлення — автентифікатор, що прив'язує відповідь до верифікатора: через доменне ім'я, як це робить WebAuthn, або до самого каналу TLS, як клієнтські сертифікати й смарт-картки. Вимагайте такого для адміністраторів і пропонуйте його всім іншим.

Фактором стає номер телефону. Код у SMS отримує той, кому оператор мобільного зв'язку зараз спрямовує цей номер. У власному переліку загроз NIST є зловмисник, який умовив оператора переадресувати номер жертви, а потім отримує код. Перш ніж надсилати код у SMS чи голосовому дзвінку, верифікатор має перевірити ознаки на кшталт недавньої заміни SIM-картки, перенесення номера до іншого оператора чи зміни пристрою. Залишайте SMS запасним варіантом, а не основним методом, і вимагайте повноцінного входу, перш ніж номер в обліковому записі можна буде змінити.

Підтвердити, підтвердити, підтвердити. Push-запит з єдиною кнопкою підтвердження просить рішення, а не доказу. Зловмисник, який уже має пароль, може надсилати запити, доки втомлений від цього шуму користувач не підтвердить один із них. NIST називає це втомою від автентифікації і більше не вважає прийнятними сценарії, де достатньо просто підтвердити: секретне значення має перейти з одного каналу в інший, наприклад число на екрані входу, яке користувач вводить на телефоні. Вибрати його з короткого списку недостатньо, бо користувач може просто вгадати. Обмежте кількість запитів, які можна надіслати після останнього успішного входу, як радить і сама настанова.

Відновлення оминає другий фактор. Посилання «Загубили телефон?», яке просить лише пароль або надсилає посилання для скидання електронною поштою, перетворює MFA на опцію, від якої зловмисник може відмовитися. Для облікових записів на AAL2 NIST вимагає, щоб відновлення спиралося на два коди відновлення, отримані різними способами, на один код відновлення плюс однофакторний автентифікатор, який досі прив'язаний до облікового запису, або на повторну перевірку особи. Про кожне відновлення користувач має отримати сповіщення. Загалом один фактор ніколи не повинен давати змогу отримати автентифікатор іншого виду: самого пароля не має вистачати, щоб надрукувати новий набір резервних кодів.

Код приймають двічі або надто довго. Шестизначний код має мільйон можливих значень, тож його стійкість залежить від правил навколо нього. Верифікатор, що приймає кілька кроків часу до і після поточного або приймає вже використаний код, дає зловмисникові, який бачив один код, вікно, щоб скористатися ним знову. Зберігайте для кожного користувача останній прийнятий крок часу й відхиляйте все, що припадає на нього або раніше. Тримайте вікно в межах одного кроку, який RFC 6238 дозволяє на затримку, і обмежте кількість невдалих спроб поспіль: NIST допускає щонайбільше 100, після чого автентифікатор вимикають.

Ключі лежать поруч із паролями. Коди із застосунку симетричні: сервер зберігає ключ кожного користувача, і той, хто скопіює ці ключі, зможе генерувати чинні коди для будь-якого облікового запису. NIST вимагає надійно захищати ключі й обмежувати доступ до них лише тими програмними компонентами, яким вони потрібні. Не зберігайте їх у відкритому вигляді в стовпцях поруч із хешами паролів. Зашифруйте їх ключем, що лежить в окремому сховищі секретів, і дозвольте читати їх лише сервісу, який перевіряє коди.

Що почитати далі

- NIST SP 800-63B-4, Definitions and Acronyms — визначення багатофакторної автентифікації, багатофакторного автентифікатора та стійкості до фішингу в чинній настанові.
- NIST SP 800-63B-4 у Computer Security Resource Center — картка публікації: видана в липні 2025 року, замінює SP 800-63B.
- RFC 6238: TOTP — алгоритм одноразових паролів на основі часу, його тестові вектори та правила щодо розбіжності годинників.
- RFC 4226: HOTP — алгоритм на основі лічильника, що лежить в основі TOTP, зокрема скорочення до шести або більше цифр.
- Web Authentication, Level 3 — специфікація W3C, на якій побудовані ключі доступу, з обліковими даними, прив’язаними до походження, і синхронізованими ключами доступу.

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 багатофакторна автентифікація?» 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/multi-factor-authentication/