Engineer

Пароль

✳ таємний рядок, що підтверджує, хто входить у систему

Що це таке

Пароль — це рядок символів, який людина тримає в пам’яті або в менеджері паролів і пред’являє, щоб показати, що обліковий запис належить їй. Визначення в глосарії NIST сходяться в головному: це літери, цифри та інші символи, за допомогою яких автентифікують особу або підтверджують право доступу, а в деяких документах також виводять криптографічні ключі. Чинна настанова щодо цифрової ідентичності, SP 800-63-4, називає його фактором «щось, що ви знаєте» і зазначає, що перша редакція вживала назву memorized secret. Парольна фраза, складена з послідовності слів, і PIN-код, що складається лише з цифр, — теж паролі в ширшому сенсі.

Той самий рядок може виконувати дві схожі на вигляд роботи. Пароль, яким входять у систему, надсилають верифікаторові, що зберігає рівно стільки, скільки треба, щоб його розпізнати, і ніколи не повинен зберігати сам пароль. Пароль, що щось розблоковує локально, лишається на пристрої. SP 800-63B називає такий пароль секретом активації (activation secret), і типовий приклад — PIN-код багатофакторного автентифікатора: його використовують на пристрої, щоб дістатися до секрету, який зберігає автентифікатор.

Складність у тому, що пароль — спільний секрет. Користувач його знає, сервіс мусить уміти його розпізнати, а кожен, хто його дізнається, може скористатися ним точнісінько як користувач. Ніщо в рядку не прив’язує його до сайту, який його попросив, тому SP 800-63B прямо пише, що паролі не стійкі до фішингу. Більша частина інженерної роботи з паролями зводиться до того, щоб обмежити ціну цього спільного знання, коли воно витікає.

Застосуйте поняття до рішення

Застосуйте «Що таке пароль?» 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

Часто пароль — єдине, що стоїть на сторожі облікового запису. Важать два види вгадування, і кожен стримує щось своє.

Онлайн-вгадування — це спроби паролів на живій формі входу, і темп тут задає сервіс. SP 800-63B обмежує кількість невдалих спроб поспіль з одним автентифікатором для одного облікового запису сотнею і вимагає перевіряти нові паролі за списком заблокованих паролів, тобто поширених, очікуваних і скомпрометованих значень, щоб варіанти, які зловмисник спробував би першими, ніколи не приймали.

Офлайн-вгадування починається, коли копія збережених даних про паролі вже покинула будівлю. Тоді немає ні форми, ні ліміту. Настанова зазначає, що зловмисники здатні офлайн обчислювати багато мільярдів хешів на секунду і що паролі, які мають це витримати, мусять бути на порядки складнішими за ті, що стикаються лише з формою входу. Тож результат вирішує зберігання. SP 800-63B вимагає солити й хешувати паролі схемою хешування паролів, яка бере пароль, сіль і коефіцієнт вартості й існує для того, щоб кожна спроба коштувала дорого. Сіль має бути щонайменше 32-бітною і такою, щоб її значення рідко збігалися, тоді два облікові записи з однаковим паролем не матимуть однакового збереженого хешу. Коефіцієнт вартості має бути таким високим, як верифікатор може собі дозволити, і з часом його підвищують. Настанова також радить записувати, яка схема і яка вартість створили кожен хеш, щоб згодом систему можна було перевести на сильнішу.

RFC 9106 описує Argon2, функцію з інтенсивним використанням пам’яті, створену саме для цього, і рекомендує її варіант Argon2id: один прохід по 2 ГіБ пам’яті як типове налаштування для всіх середовищ або три проходи по 64 МіБ там, де пам’яті мало. RFC пише, що перше налаштування максимізує

витрати зловмисника на спеціалізованому обладнанні для повного перебору, і саме ця властивість важить, коли файл із хешами витік. OWASP Password Storage Cheat Sheet наводить нижчі робочі мінімуми й запасні варіанти для систем, які не можуть використовувати Argon2id, і формулює правило, що лежить під усіма ними. Швидкі хеш-функції на кшталт SHA-256 не годяться, бо дають зловмисникові змогу дуже швидко перебирати варіанти, а паролі хешують, а не шифрують, бо шифрування працює в обидва боки і з нього можна відновити оригінал.

Як це виглядає на практиці

До того, як у компанії запровадили загальнокорпоративну систему керування паролями, різні люди зберігали й передавали облікові дані різними, довільними способами. Сторінка описує, як це виглядає щодня: рідко драматичний злам, частіше пароль у повідомленні в чаті, спільний логін, пароль до якого ніхто не змінює, і повільне накопичення дрібних вразливостей. Система дала організації один послідовний спосіб зберігати й передавати облікові дані, а головна думка сторінки в тому, що окремі люди не можуть від неї відмовитися, бо менеджер паролів, яким користується лише половина команди, майже не допомагає, коли ризик живе саме в тій половині, яка ним не користується. Щойно облікові дані опинилися в одному керованому місці, керування доступом стало простішим і послідовнішим. SP 800-63B вказує той самий напрям з боку сервісу: верифікатори мусять дозволяти менеджери паролів і автозаповнення, а вставлення їм слід дозволяти. Настанова також називає менеджер паролів способом мати окремий пароль для кожного сервісу.

Коли другий сервер створили з коду, аргумент дала історія першої машини. Її створили вручну, а брандмауер додали згодом, і так лишається вікно, коли щойно створений хост зі стандартною політикою паролів доступний з інтернету. Виправлення — порядок: нова машина з’являється одразу за своїм брандмауером, бо плей брандмауера виконується перед плеєм сервера.

Під час входу в обліковий запис великого поштового провайдера застосунок сказав користувачеві, що пароль неправильний. Пароль був правильний. Цей провайдер вимагає для сторонніх клієнтів окремого пароля застосунку, тож один обліковий запис мав два види паролів, а повідомлення, що знало лише про один, відсилало користувача вводити знову те, що ніколи не

спрацювало б. Переписане повідомлення називає провайдера, пояснює, що він вимагає окремого пароля застосунку, і вказує, де такий пароль створити, а тест перевіряє назву провайдера й тип облікових даних.

Де виникають проблеми

Один пароль на багато облікових записів. Коли той самий пароль захищає кілька сервісів, злам найслабшого відчиняє решту. SP 800-63B називає це password stuffing: пароль, викрадений з одного сайту, пробують на облікових записах тієї самої людини на інших сайтах. Користувачі захищаються окремим паролем для кожного сервісу, на практиці — менеджером паролів, а сервіси — перевіркою нових паролів за значеннями з попередніх витоків.

Правила, що роблять паролі слабшими. Вимоги до складу пароля здаються суворими, а дають передбачувані результати. Приклад із самої настанови — користувач, який обрав би «password», а коли вимагають велику літеру й цифру, обирає «Password1», а коли ще й символ — «Password1!». Настанова відкидає правила щодо складу, планову зміну паролів, підказки, які можна прочитати до входу, і контрольні запитання на кшталт клички першого домашнього улюбленця, але й далі вимагає зміни, щойно з’являються ознаки компрометації.

Швидкий хеш або одна сіль на всі рядки. Таблиця паролів, захешованих функцією загального призначення або посолених одним і тим самим значенням, дає офлайн-зловмисникові дешеві спроби, які можна перевіряти проти всіх облікових записів одночасно. Системи, що успадкували старі хеші, можуть перехешувати кожен пароль новою схемою під час наступного входу користувача, як описує шпаргалка OWASP, а збережена позначка схеми показує, які рядки ще хешовано по-старому.

Пароль обрізають або його не можна вставити. SP 800-63B вимагає перевіряти весь надісланий пароль, а не обрізану частину, і хоче, щоб вставлення було дозволене: тоді менеджери паролів працюють і там, де автозаповнення не спрацьовує. Деякі функції хешування мають власну межу: шпаргалка OWASP зазначає, що bcrypt у більшості реалізацій приймає щонайбільше 72 байти вхідних даних, і радить сервісам, які його використовують, забезпечувати цей максимум.

Пароль живе там, де йому не місце. Пароль, вставлений у чат, записаний у тікет чи виведений у журнал, — це копія поза всіма засобами контролю, які має справжнє сховище. Єдиний спосіб забрати цю копію назад — змінити пароль, і саме тому одне кероване місце важить більше за будь-яке правило про те, як має виглядати пароль.

Що почитати далі

- Password у глосарії NIST CSRC — визначення з FIPS 140-2, SP 800-63-4 та інших документів, де парольна фраза — окремий випадок.
- NIST SP 800-63B-4 — чинні правила вибору, перевірки й зберігання паролів з обґрунтуванням у додатку A.
- RFC 9106 — Argon2 із двома рекомендованими налаштуваннями Argon2id для хешування паролів.
- OWASP Password Storage Cheat Sheet — робочі параметри для Argon2id, scrypt, bcrypt і PBKDF2 та оновлення старих хешів.

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 пароль?» 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/password/