Engineer

Authentifizierung

der Beleg, dass eine Anfrage vom angegebenen Absender stammt

Was es ist

Authentifizierung ist die Prüfung, bevor ein System eine behauptete Identität annimmt. Jemand sagt „Ich bin dieses Konto“, und das System fordert einen Nachweis, den nur der Kontoinhaber vorlegen können sollte. Entwickler verkürzen das oft zu authn. Oberflächen nennen denselben Moment „Anmelden“ oder „Einloggen“.

Identifikation ist die Behauptung selbst, etwa die Eingabe eines Nutzernamens. Authentifizierung ist ihr Nachweis. Autorisierung ist die anschließende, getrennte Entscheidung darüber, was die nun bekannte Identität darf. Selbst HTTP verwischt die Grenze: RFC 9110 nennt Statuscode 401 „Unauthorized“, definiert ihn aber als Anfrage ohne gültige Authentifizierungsdaten. Der Server muss mit einem WWW-Authenticate-Header erklären, wie diese zu liefern sind. „Wir wissen, wer Sie sind, und lehnen trotzdem ab“ heißt 403 Forbidden.

```
HTTP/1.1 401 Unauthorized
WWW-Authenticate: Bearer realm="api"
```

Nachweise gehören zu Arten, meist Faktoren genannt. Wissen ist etwa ein Passwort oder eine PIN. Besitz ist ein Telefon, ein Hardware-Sicherheitsschlüssel oder eine private Schlüsseldatei. Ein körperliches Merkmal ist ein Fingerabdruck oder ein Gesicht. Mehrfaktorauthentifizierung verbindet zwei unterschiedliche Arten, damit ein Dieb, der einen Faktor erlangt, den anderen weiterhin nicht besitzt. Zwei Passwörter bleiben ein Faktor.

Vieles betrifft überhaupt keinen Menschen. Dienste zeigen einander API-Tokens. SSH macht die maschinelle Form deutlich: RFC 4252 macht „publickey“ zur einzigen Methode, die jede Implementierung unterstützen muss. Der Client

beweist den Besitz eines privaten Schlüssels durch eine Signatur, sodass der Schlüssel nie das Netzwerk überquert. Single Sign-on verlagert die Prüfung. Bei OpenID Connect authentifiziert ein Identitätsanbieter den Nutzer und übergibt der Anwendung ein signiertes ID-Token. Sie sieht deshalb nie ein Passwort. Passkeys auf Basis der W3C Web Authentication API halten einen auf eine Website begrenzten privaten Schlüssel auf dem Gerät. Sie signieren die Server-Challenge zusammen mit dem Ursprung, mit dem der Browser tatsächlich spricht. Eine nachgeahmte Domain erhält daher nichts Wiederverwendbares.

Das Konzept für eine Entscheidung nutzen

Nutzen Sie „Was ist Authentifizierung?“ für eine konkrete Entscheidung. Wir können helfen, das Konzept mit dem System in Verbindung zu bringen, das Sie entwickeln oder einkaufen. Konten, Passwörter und Berechtigungen werden so geordnet, dass alle genau den beruflich erforderlichen Zugriff haben und ihn am Tag ihres Ausscheidens verlieren.

Unsere Leistungen entdecken Kontakt aufnehmen →

Die Annahmen hinter dem Begriff prüfen

Nutzen Sie die Erklärung und die verlinkten Arbeiten, um Fragen für Ihr Projekt zu formulieren. Wir können die Annahmen gemeinsam mit Ihnen prüfen, bevor ein Werkzeug ausgewählt wird. 1.000 RBAC-Regeln für sichere Linux-Server — 1,000 Zugriffsregeln zum Schließen einer Sicherheitslücke

1.000 RBAC-Regeln für sichere Linux-Server Kontakt aufnehmen →

Warum es wichtig ist

Jede andere Kontrolle setzt voraus, dass diese funktioniert. Berechtigungen, Auditprotokolle, Ratenlimits und Abrechnung beziehen sich auf eine Identität. Ist deren Feststellung schwach, werden sie alle zuverlässig auf die falsche Partei angewendet. Eine Protokollzeile über eine von einem Administrator gelöschte Tabelle hält das Konto fest. Erst gute Authentifizierung macht daraus eine Person.

Praktisch ist zu entscheiden, wie stark die Prüfung an jeder Tür sein muss. NIST SP 800-63B bietet dafür die klarste Skala. Die erste Vertrauensstufe AAL1 akzeptiert einen Faktor. AAL2 braucht zwei und verlangt vom Prüfer mindestens eine phishingresistente Option. AAL3 verlangt einen phishingresistenten Authentikator mit nicht exportierbarem privatem Schlüssel. Zu einer beliebten Wahl ist die Leitlinie deutlich: Von Hand eingegebene Codes sind nicht phishingresistent, ob sie per SMS kommen oder in einer App stehen. Nichts bindet die getippten Ziffern an die anfragende Website.

Revision 4 desselben Dokuments, die Revision 3 am 1. August 2025 ablöste, klärt auch mehrere Passwortfragen. Ein Passwort als einziger Faktor muss mindestens 15 Zeichen lang sein, ein ausschließlich bei Mehrfaktoranmeldung verwendetes

Passwort mindestens 8. Andere Zusammensetzungsregeln und regelmäßige Wechsel dürfen nicht verlangt werden. Bei Hinweisen auf Kompromittierung muss aber ein Wechsel erzwungen werden. Neue Passwörter werden gegen eine Sperrliste häufiger und geleakter Werte geprüft. Nutzer erfahren den Ablehnungsgrund. Gespeichert wird ein gesalzener Hash mit einem Verfahren samt Kostenfaktor. Aufeinanderfolgende Fehlversuche gegen einen Authentikator sind auf 100 begrenzt. Für AAL2-Sitzungen wird erneute Prüfung mindestens alle 24 Stunden und nach einer Stunde Inaktivität empfohlen.

Daraus folgt unsere Empfehlung. Besitzen Nutzer bereits ein Konto bei einem kompetenten Identitätsanbieter, sollte dieser prüfen und die Anwendung seine signierte Antwort akzeptieren. Ein Passwort, das die Anwendung nie speichert, kann sie auch nie offenlegen. Muss die Anwendung Passwörter halten, folgen Sie 800-63B wörtlich statt aus dem Gedächtnis. Schützen Sie alles Administrative mit einer phishingresistenten Methode.

Wie es in der Praxis aussieht

Beim Umzug des Quellcodes auf den eigenen Git-Server stand Authentifizierung an erster Stelle der Forge-Betriebsaufgaben. Die gewählte Forge ist SSH-zentriert und hat keine administrative Weboberfläche. Der Zugang erfolgt deshalb per Schlüssel. Anonyme Sitzungen werden ebenso abgewiesen wie solche ohne Schlüssel. Zusätzlich ist jedes deklarierte Repository privat. Der Listener an Port 23231, der diese Schlüssel prüft, nimmt nur Loopback-Verbindungen an. Jeder externe Login passiert zuerst einen Sprunghost, und die deklarierte Firewalloberfläche bleibt gleich groß.

Auf Organisationsebene sieht dieselbe Sorge anders aus. In einer früheren Rolle als Infrastructure Engineer wurden Identitäten und Zugriffe über M365, 1Password, Red Hat SSO und OKTA SSO zusammengeführt. Unterschiedliche Menschen hatten über die Zeit Rechte nach Bedarf erteilt. Das Prinzip der geringsten Rechte wurde durchgesetzt, Ein- und Austrittsverfahren wurden vereinheitlicht. Zugänge wurden zügig entzogen, statt nach dem Ausscheiden bestehen zu bleiben. Das Risiko unberechtigter Zugriffe sank stark, und Berechtigungen wurden prüfbar. Das Team konnte endlich beantworten, wer etwas erreicht und warum.

Authentifizierung hat auch ein Gesicht. Nutzer beurteilen sie an Fehlermeldungen. Bei der Anmeldung einer Anwendung bei einem großen Mailanbieter hieß es, das Passwort sei falsch. Es war richtig. Der Anbieter

verlangt für Drittclients ein app-spezifisches Passwort und lehnt das Kontopasswort unabhängig von sorgfältiger Eingabe ab. Die neue Meldung nennt den Anbieter, erklärt, dass er für andere Clients ein app-spezifisches Passwort verlangt, und sagt, wo sich dieses erstellen lässt. Ein Test löst diese Ablehnung aus und verlangt Anbietername und erforderlichen Anmeldedatentyp im Text. Die alte Formulierung kann so nicht unbemerkt zurückkommen.

Wo es schiefgeht

- **Die Meldung macht den Nutzer für eine fremde Ursache verantwortlich.** Ein gesperrtes Konto, ein abgelaufenes Passwort, eine vom Codegenerator abweichende Uhr und die Anforderung anderer Zugangsdaten erscheinen als „falsches Passwort“. Der Nutzer tippt erneut etwas, das nie funktionieren wird. Nennen Sie die behebbare Ursache, halten Sie aber Antworten für existierende und nicht existierende Konten gleich, damit das Formular keine Liste gültiger Nutzernamen preisgibt.
- **Eine Phishingseite leitet den Code in Echtzeit weiter.** Nutzer geben Passwort und sechsstelligen Code in eine Kopie der echten Seite ein. Diese leitet beides innerhalb der von RFC 6238 als Standardzeitintervall empfohlenen 30 Sekunden an die echte Website weiter. Einmalcodes verhindern die Wiederverwendung gestohlener Passwörter, helfen aber wenig gegen unmittelbare Weiterleitung. Ursprungsgebundene Schlüssel wie Passkeys signieren nur für eine Website. Deshalb gelten sie bei NIST als phishingresistent, eingetippte Codes dagegen nicht.
- **Zugangsdaten überleben ihre Besitzer.** Beim Austritt wird das Verzeichniskonto geschlossen. Ein SSH-Schlüssel in `authorized_keys`, ein persönliches Zugriffstoken und ein gemeinsames Administratorpasswort bleiben dennoch gültig. Jedes Zugangsmittel braucht einen Verantwortlichen, einen Ort, an dem es dokumentiert ist, und einen Termin für die erneute Prüfung.
- **Die Sitzung bleibt lange nach der Anmeldung gültig.** Ein monatelanges Cookie oder Token ist ein Passwort mit langer Zündschnur. Abmelden muss die Serversitzung beenden, nicht bloß das Cookie eines Browsers löschen. Ein in sich geschlossenes Token, das der Server nie nachprüft, bleibt bis zum Ablauf verwendbar, egal was mit dem Konto geschieht.
- **Die Kontowiederherstellung ist der schwächste Eingang.** Ein starker Login neben einem per E-Mail verschickten Rücksetzlink macht das Postfach zum eigentlichen Zugangsmittel. Die Wiederherstellung verdient dieselbe Prüfung wie die Haupttür. Ein Angreifer wählt die schwächere.

Weiterführende Lektüre

- NIST SP 800-63B-4 — der aktuelle Text zu Vertrauensstufen, mindestens 15 Passwortzeichen und Phishingresistenz.
- NIST SP 800-63B, Revision 3 — die Leitlinie von 2017, weiterhin veröffentlicht und seit dem 1. August 2025 als durch Revision 4 ersetzt gekennzeichnet.
- RFC 9110, Abschnitt 11: HTTP Authentication — Challenge und Antwort hinter 401, WWW-Authenticate und dem Authorization-Header.
- RFC 4252: The SSH Authentication Protocol — öffentliche Schlüssel, Passwort und hostbasierte Anmeldung nach SSH.
- RFC 6238: TOTP — wie Authenticator-Apps aus gemeinsamem Geheimnis und Uhrzeit kurze Codes ableiten.
- Web Authentication, Level 3 — die W3C-API hinter Passkeys und ursprungsgebundenen öffentlichen Schlüsseln.

Glossar

Für Ihre tatsächlichen Rahmenbedingungen auswählen

Beschreiben Sie den Bereitstellungsweg, die Betriebsrisiken und die Bereitschaftsverantwortlichen. Wir können eine Änderung mit Prüfungen, einem Wiederherstellungsweg und klaren Zuständigkeiten abgrenzen.

Preise ansehen Kontakt aufnehmen →

Die Begriffe praktisch nutzbar machen

Fragen Sie, wie „Was ist Authentifizierung?“ auf Ihre Organisation zutrifft. Ein konkretes Beispiel aus Ihrem bestehenden System reicht aus, um das Gespräch zu beginnen.

Unsere Leistungen entdecken

welcome@engineer.company +45 50 10 61 84 Kopenhagen, Dänemark

Nehmen Sie diese Seite mit

Kontaktkarte

Mit Sorgfalt und Liebe gestaltet von D.

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

Copyright © 2025 – heute · Engineer ApS · Alle Rechte vorbehalten.

Gedruckt von https://engineer.company/de/glossary/authentication/