Engineer

認証

要求が、その名乗る相手から来たものであることの証明

どのようなものか

認証は、システムが申告された身元を受け入れる前に行う確認です。「このアカウントの本人です」という相手に、本人だけが用意できるはずの証拠を求めます。エンジニアはauthnと略し、画面では同じ場面を「サインイン」「ログイン」と呼びます。

識別は、利用者名の入力など、名乗ること自体です。認証は、その根拠です。認可は、その後、確認済みのIDに何を許すかという別の判断です。HTTPでさえ混同しています。RFC 9110は401を「Unauthorized」と名付けながら、有効な認証情報のない要求と定義し、その提供方法を示すWWW-Authenticateヘッダーを必須とします。「誰かは分かるが、それでも拒否する」のは403 Forbiddenです。

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

証拠には、通常、要素と呼ぶ種類があります。知っているものはパスワードやPIN、持っているものは携帯電話、ハードウェアセキュリティキー、秘密鍵ファイル、自分自身の特徴は指紋や顔です。多要素認証は異なる二種類を組み合わせ、一つを盗まれても他方が足りないようにします。パスワード二つは、一要素のままです。

人が関わらない認証も多くあります。サービス同士はAPIトークンを提示し、SSHを見ると、マシンでの認証の仕組みがよく分かります。RFC 4252は、すべての実装に対応が必須とされる唯一の方式を「publickey」とし、クライアントは署名で秘密鍵の所持を示すため、鍵自体はネットワークを渡りません。シングルサインオンは確認を外へ移します。OpenID Connectでは、IDプロバイダーが認証して署名付きIDトークンを渡すため、アプリケーションはパスワードを見ません。W3C Web Authentication APIに基づくパスキーは、機器に、一つのサイトに

限定した秘密鍵を保持します。サーバーのチャレンジと、ブラウザーが実際に通信するオリジンを一緒に署名するため、似せたドメインは再利用できるものを得られません。

概念を判断に生かす

「認証とは？」を実際の判断に生かしましょう。その概念を、開発または購入するシステムに結び付けて考えるお手伝いをします。 アカウント、パスワード、権限を整理し、業務に必要なアクセスだけを付与し、退職者の権限は当日に解除します。

サービスを見る お問い合わせ →

用語の背後にある前提を検証する

解説とリンク先の実績を参考に、プロジェクトについての質問を整理してください。ツールを選ぶ前に、前提を一緒に検討できます。 Linuxサーバーを守る1,000件のRBACルールを作成 — 1,000 セキュリティ上の穴を塞ぐアクセスルール

Linuxサーバーを守る1,000件のRBACルールを作成 お問い合わせ →

なぜ重要なのか

他のすべての制御は、この確認の成功を前提にします。権限、監査ログ、レート制限、請求はIDをキーにし、そのIDを確定する確認が弱ければ、すべてが忠実に誤った相手へ適用されます。管理者がテーブルを削除したというログは、どのアカウントかを記録するだけです。それを誰だったかに変えるのは、適切な認証です。

実務上の判断は、各入り口にどれほど強い確認が必要かです。NIST SP 800-63Bは、そのための明確な尺度を示します。最初の保証レベルAAL1は一要素を認めます。AAL2は二要素を求め、少なくとも一つのフィッシング耐性のある選択肢を検証者が提供する必要があります。AAL3は、秘密鍵をエクスポートできない、フィッシング耐性のある認証器を求めます。よく使われる方法にも明快です。SMSでもアプリ表示でも、手入力するコードは、要求したサイトと数字が結び付かないため、フィッシング耐性がありません。

2025年8月1日に改訂3に代わった改訂4は、パスワードの論点も定めました。単独要素のパスワードは少なくとも15文字、多要素ログイン内だけで使うものは少なくとも8文字です。他の文字構成ルールや定期変更を要求してはいけませんが、侵害の証拠があれば変更を強制します。新しいパスワードは一般的・漏えい済みのものの拒否一覧と照合し、拒否理由を伝え、保存形式は、計算負荷を調整するコストパラメーターを持つ方式でソルト付きハッシュにします。一つの認証器への連続失敗は100回を上限とします。AAL2のセッションには、少なくとも24時間ごとに、また一時間操作がなかった後に、再確認することを勧めています。

当社の推奨も、この考え方に従います。信頼できるIDプロバイダーのアカウントがすでにあるなら、そこで認証し、署名付きの答えを受け入れます。保持しないパスワードを、アプリケーションが漏らすことはないためです。保持が必要なら、記憶ではなく800-63Bの記述どおりに従い、管理用のものには、フィッシング耐性のある方法を設けます。

実際の活用例

ソースコードを自社のGitサーバーへ移した際、フォージ運用の最初の項目は認証でした。選んだフォージはSSHを中心にし、管理用ウェブ画面がないため、入り口は鍵です。匿名のセッションも、鍵のないものも拒否し、すべての宣言したリポジトリを非公開にしています。鍵を確認するポート23231のリスナーは、ループバックからだけ接続を受けます。そのため、外部のログインは踏み台ホストを通り、ファイアウォールの公開範囲は増えません。

組織全体では、同じ関心も別の形になります。以前のインフラエンジニアの仕事では、長年、別々の人が場当たり的に権限を与えていたM365、1Password、Red Hat SSO、OKTA SSOのIDとアクセスを統合しました。最小権限を適用し、参加・離任の手順を標準化して、離れた後も残さず、速やかに取り消せるようにしました。不正アクセスのリスクは大きく下がり、監査可能になり、チームは「誰がここにアクセスでき、それはなぜか」に答えられるようになりました。

認証には利用者に見える面もあり、エラーメッセージで判断されます。あるアプリが大手メール事業者へログインする場面では、パスワードが誤りだと表示していました。実際は正しく、事業者が第三者クライアントにアプリ専用パスワードを求め、どれほど正確に入力しても通常のものを拒否していました。事業者名、専用パスワードが必要なこと、作成先を伝える内容へ書き直しました。現在は、その拒否を再現し、事業者名と必要な認証情報の種類を含むことをテストして、古い文章が黙って戻らないようにしています。

問題が起きるところ

- **別の原因を利用者のせいにします。** アカウントのロック、パスワードの期限切れ、コード生成器との時計のずれ、別の認証情報の要求を、すべて「パスワードが違う」と伝え、決して通らないものを再入力させます。利用者が対処できる原因を示します。ただし、有効な利用者名の一覧を漏らさないよう、存在するアカウントと存在しないものへの返答は同じにします。

- **偽ページがリアルタイムにコードを中継します。** 本物をまねたページへ入力したパスワードと六桁コードを、RFC 6238が標準の時間幅に勧める30秒以内に、本物へ転送します。一回限りのコードは、盗んだパスワードの再利用には効きますが、即時中継にはほとんど効きません。パスキーなど、オリジンに結び付く鍵は一つのサイトだけに署名するため、NISTはそれを耐性あり、入力コードを耐性なしとします。
- **認証情報が、持ち主の離任後も残ります。** 名簿のアカウントを閉じても、`authorized_keys` のSSH鍵、個人アクセストークン、共有管理者パスワードが動き続けます。各認証情報には所有者、記録場所、再確認日が必要です。
- **ログイン後、セッションが長く有効です。** 数か月続くCookieやトークンは、長い導火線のパスワードです。ログアウトは、一つのブラウザーのCookie削除だけでなく、サーバーのセッションを終了させる必要があります。サーバーが再確認しない自己完結型トークンは、アカウントに何が起きても期限まで使えます。
- **アカウント復旧が最も弱い入り口です。** 強いログインの横にメールの再設定リンクがあれば、メールボックスが本当の認証情報になります。攻撃者は弱いほうを選ぶため、復旧にも正面と同じ検討が必要です。

参考資料

- NIST SP 800-63B-4：現行の保証レベル、15文字の最小長、フィッシング耐性の条件です。
- NIST SP 800-63B、改訂3：2017年の指針。2025年8月1日から改訂4に置き換わったことを明記し、公開を続けています。
- RFC 9110、第11節：HTTP Authentication：401、WWW-Authenticate、Authorizationヘッダーを支えるチャレンジと応答の仕組みです。
- RFC 4252：The SSH Authentication Protocol：SSHの公開鍵、パスワード、ホストベースのログインです。
- RFC 6238：TOTP：共有シークレットと時計から、認証アプリが短いコードを生成する仕組みです。
- Web Authentication、Level 3：パスキーの基盤となる、オリジンに限定した公開鍵認証のW3C APIです。

用語集

実際の制約に合わせて選ぶ

デプロイの流れ、運用上のリスク、緊急対応の担当者をお知らせください。検査、復旧手段、明確な責任の所在を伴う変更の範囲を定められます。

料金を見る お問い合わせ →

用語を実務に結び付ける

「認証とは？」がお客様の組織にどう当てはまるかをご相談ください。現在のシステムの具体例が一つあれば、話を始められます。

サービスを見る

welcome@engineer.company +45 50 10 61 84 デンマーク、コペンハーゲン

このページを持ち歩く

連絡先カード

心を込めて丁寧に制作： D.

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

著作権 © 2025 – 現在 · Engineer ApS · すべての権利を留保します。

印刷元： https://engineer.company/ja/glossary/authentication/