

Uitleg

Verificatie vs. Autorisatie

Authenticatie (verificatie): Bevestigen dat een gebruiker is wie hij beweert te zijn. Methoden zijn onder andere:

- **Wachtwoorden:** Gebruikers kiezen een geheim wachtwoord. Eenvoudig, maar vatbaar voor sterke/brute-force aanvallen als niet goed ingericht.
- **Multi-factor authenticatie (MFA):** Combineert iets dat je weet (wachtwoord) met iets dat je hebt (smartphone-app, SMS-code) of iets dat je bent (biometrie).
- **Biometrie:** Vingerafdruk, gezichtsherkenning of irisscan. Comfortabel, maar kan privacyrisico's en false positives/negatives hebben.

Autorisatie: Na succesvolle authenticatie bepaalt het systeem welke resources en acties een gebruiker mag uitvoeren. Veelgebruikte modellen:

- **Role-Based Access Control (RBAC):** Gebruikers krijgen rollen, en rollen krijgen permissies. Eenvoudig beheer bij groeiende aantallen rechten.
- **Attribute-Based Access Control (ABAC):** Beslissingen op basis van attributen (tijdstip, locatie, type apparaat) naast gebruikerskenmerken.

Wachtwoordbeveiliging

- **Sterkte eisen:** Minimaal 8-12 tekens, combinatie van hoofdletters, cijfers en speciale tekens.
- **Hashing en salting:** Sla wachtwoorden nooit in platte tekst op. Gebruik een cryptografische hashfunctie (bijvoorbeeld bcrypt of Argon2) en voeg een unieke salt per wachtwoord toe om rainbow-table aanvallen te voorkomen.
- **Opslag en beleid:** Denk aan periodieke reset, lockout na mislukte pogingen en geblokkeerde accounts na verdachte activiteiten.

MFA en moderne methoden

- **SMS- of e-mailcodes:** Gemakkelijk, maar kwetsbaar voor SIM-swaps of onderschepping.
- **TOTP-apps (Time-based One-Time Password):** Google Authenticator, Authy. Veiliger omdat codes lokaal gegenereerd worden.
- **Push-notificaties:** Gebruiksvriendelijk: een push naar de telefoon ter bevestiging.
- **WebAuthn en FIDO2:** Standaarden voor wachtwoordloze authenticatie met hardware tokens (zoals YubiKey) of biometrie.

Autorisatiemodellen

- **RBAC:** Eenvoudig in te richten: definieer rollen zoals “student”, “docent” en “beheerder”. Wijs per rol rechten toe (bijv. lezen, schrijven, beheren).
- **ABAC:** Complexer, biedt fijnmazige controle. Voorbeeld: alleen docenten mogen lesmateriaal bewerken én alleen binnen werktijden.
- **Least privilege:** Geef gebruikers altijd de minimaal noodzakelijke rechten.

Beveiligingspatronen en anti-patronen

- **Fail-secure vs. fail-open:** Bepaal hoe je systeem reageert bij fouten. Bij fail-secure is toegang geblokkeerd; bij fail-open blijft het open, wat risico's geeft.
- **Session management:** Gebruik secure flags op cookies, stel een redelijke sessietimeout in en forceer herauthenticatie voor kritieke acties.
- **Password reset flows:** Vereisen kennisvragen of extra verificatie om social engineering te voorkomen.

Praktijkvoorbeeld

Stel je bouwt een onderwijsportal:

1. Gebruiker logt in met username+password (bcrypt-hashing).
2. Bij gevoelige data (cijfers, persoonlijke gegevens) is MFA verplicht via TOTP-app.
3. De portal werkt met RBAC:
 - **Student:** Bekijkt opdrachten en resultaten.
 - **Docent:** Plaatst opdrachten, beoordeelt werk.
 - **Beheerder:** Beheert gebruikers en systeeminstellingen.
4. Session cookies bevatten een secure en HttpOnly-flag en verlopen na 30 minuten inactiviteit.
5. Voor het resetten van wachtwoord krijgt de gebruiker een e-mail met een eenmalige link die na 10 minuten verloopt.

Revision #1

Created 13 June 2025 08:24:02 by Admin

Updated 13 June 2025 08:31:47 by Admin