17.3 Benutzer, Gruppen, Rechte und Zugriffskontrolle in der Praxis

In der Systemadministration müssen Benutzer, Gruppen, Rollen und Berechtigungen sauber verwaltet werden.

Ziel ist:

```
Benutzer sollen genau die Zugriffe bekommen,
die sie für ihre Arbeit benötigen.
```

Nicht mehr.

Nicht weniger.

Merksatz:

```
Rechteverwaltung sorgt dafür,
dass Benutzer nur auf erlaubte Systeme,
Dienste
und Daten zugreifen können.
```

Warum Rechteverwaltung wichtig ist

Fehlerhafte Rechte können zu schweren Sicherheitsproblemen führen.

Mögliche Folgen:

```
unberechtigter Datenzugriff
versehentliches Löschen
Manipulation von Daten
Missbrauch von Adminrechten
Datenabfluss
Datenschutzverletzung
Ransomware-Ausbreitung
fehlende Nachvollziehbarkeit
```

Merksatz:

```
Falsche Rechte sind ein Sicherheitsrisiko.
```

Benutzerkonto

Ein Benutzerkonto ist eine digitale Identität in einem System.

Es enthält typischerweise:

```
Benutzername
Kennwort oder andere Anmeldemethode
Gruppenmitgliedschaften
Rollen
Berechtigungen
Profilinformationen
Anmeldeinformationen
Status aktiv oder deaktiviert
```

Merksatz:

```
Benutzerkonto = digitale Identität eines Benutzers.
```

Persönliche Benutzerkonten

Jeder Benutzer sollte ein eigenes Konto haben.

Vorteile:

```
Aktionen sind nachvollziehbar.
Rechte können gezielt vergeben werden.
Offboarding ist einfacher.
Passwörter müssen nicht geteilt werden.
Sicherheitsvorfälle lassen sich besser untersuchen.
```

Merksatz:

```
Ein Benutzer,
ein persönliches Konto.
```

Sammelkonten

Sammelkonten sind Konten, die von mehreren Personen gemeinsam genutzt werden.

Beispiele:

```
admin
support
praktikant
lager
service
teamkonto
```

Probleme:

```
keine eindeutige Zuordnung
Passwortweitergabe
schwieriges Offboarding
schlechte Protokollierung
höheres Missbrauchsrisiko
```

Merksatz:

```
Sammelkonten vermeiden,
besonders bei administrativen Zugängen.
```

Gruppen

Gruppen fassen Benutzer zusammen.

Beispiel:

```
Alle Benutzer der Buchhaltung kommen in Gruppe Buchhaltung.
```

Dann können Rechte der Gruppe gegeben werden, statt jedem Benutzer einzeln.

Vorteile:

übersichtlich
leichter zu verwalten
einfacher bei neuen Benutzern
einfacher bei Rollenwechsel
weniger Einzelberechtigungen

Merksatz:

Rechte möglichst über Gruppen vergeben.

Rollen

Eine Rolle beschreibt eine Aufgabe oder Funktion.

Beispiele:

Helpdesk
Netzwerkadministrator
Serveradministrator
Datenbankadministrator
Backup-Operator
Standardbenutzer
Abteilungsleitung
Leser
Bearbeiter
Prüfer

Merksatz:

Rolle beschreibt Aufgabe,
Gruppe kann Rechte technisch bündeln.

Rollenbasierte Zugriffskontrolle

Rollenbasierte Zugriffskontrolle wird abgekürzt:

RBAC

RBAC steht für:

```
Role-Based Access Control
```

Dabei werden Rechte über Rollen vergeben.

Beispiel:

```
Rolle Helpdesk darf Passwörter zurücksetzen.

Rolle Backup-Operator darf Backups prüfen.

Rolle Netzwerkadministrator darf Switches verwalten.
```

Merksatz:

```
RBAC vergibt Rechte nach Rollen.
```

Attributbasierte Zugriffskontrolle

Attributbasierte Zugriffskontrolle wird abgekürzt:

```
ABAC
```

ABAC steht für:

```
Attribute-Based Access Control
```

Dabei werden Zugriffe anhand von Eigenschaften entschieden.

Beispiele für Attribute:

```
Abteilung
Standort
Gerätetyp
Uhrzeit
Sicherheitsstatus des Geräts
Datenklassifizierung
```

Benutzerrolle

Merksatz:

ABAC entscheidet nach Eigenschaften und Bedingungen.

RBAC und ABAC vergleichen

Modell	Grundlage	Beispiel
RBAC	Rolle	Helpdesk darf Passwörter zurücksetzen
ABAC	Attribute	Zugriff nur aus Firmennetz und mit verwaltetem Gerät

Merksatz:

RBAC arbeitet mit Rollen.
ABAC arbeitet mit Eigenschaften.

Berechtigung

Eine Berechtigung beschreibt, was ein Benutzer oder Dienst tun darf.

Beispiele:

Datei lesen
Datei schreiben
Datei löschen
Benutzer anlegen
Passwort zurücksetzen
Server neu starten
Firewall-Regel ändern
Datenbank abfragen
Backup wiederherstellen

Merksatz:

Berechtigung = erlaubte Aktion auf ein Ziel.

Recht und Rolle unterscheiden

Recht:

einzelne Erlaubnis

Rolle:

Bündel aus mehreren Rechten für eine Aufgabe

Beispiel:

Recht:
Benutzer darf Ticket lesen.

Rolle:
Supportmitarbeiter darf Tickets lesen,
bearbeiten
und schließen.

Merksatz:

Rechte sind einzelne Erlaubnisse.
Rollen bündeln Rechte.

Least Privilege

Least Privilege bedeutet:

Jeder Benutzer,
Dienst
und Prozess erhält nur die Rechte,
die wirklich benötigt werden.

Beispiele:

Standardbenutzer ohne lokale Adminrechte

Dienstkonto nur mit Zugriff auf benötigte Datenbank

Praktikant nur mit Zugriff auf Schulungsunterlagen

Backup-Operator darf Backups prüfen,
aber keine Benutzer verwalten

Merksatz:

So wenig Rechte wie möglich,
so viele wie nötig.

Need to Know

Need to Know bedeutet:

Zugriff nur auf Informationen,
die für die Aufgabe erforderlich sind.

Beispiel:

Personalabteilung darf Personalakten sehen.

IT-Support muss nicht automatisch Gehaltsdaten sehen.

Projektteam sieht nur Projektdaten.

Merksatz:

Nicht jeder muss alles sehen können.

Trennung von Aufgaben

Aufgabentrennung bedeutet:

kritische Aufgaben werden auf mehrere Personen oder Rollen verteilt.

Beispiele:

Eine Person beantragt Änderung.
Andere Person gibt Änderung frei.

Eine Person erstellt Zahlung.
Andere Person bestätigt Zahlung.

Eine Person richtet Backup ein.
Andere Person prüft Restore-Test.

Merksatz:

Aufgabentrennung reduziert Fehler und Missbrauch.

Vier-Augen-Prinzip

Beim Vier-Augen-Prinzip prüft oder genehmigt mindestens eine zweite Person eine kritische Aktion.

Typische Aktionen:

Adminrechte vergeben
Firewall nach außen öffnen
Backup löschen
Produktivsystem ändern
Benutzer mit hohen Rechten anlegen
kritische Daten exportieren

Merksatz:

Kritische Änderungen nicht allein freigeben.

Administrative Konten

Administrative Konten haben erhöhte Rechte.

Beispiele:

```
lokaler Administrator
Domänenadministrator
Cloud-Administrator
Firewall-Administrator
Datenbankadministrator
Backup-Administrator
Root-Konto
```

Merksatz:

```
Administrative Konten sind besonders schützenswert.
```

Admin- und Alltagskonto trennen

Administratoren sollten zwei Konten haben:

```
normales Benutzerkonto für Alltag

separates Adminkonto für Administration
```

Vorteile:

```
weniger Risiko bei Phishing
weniger Malware-Schaden
bessere Nachvollziehbarkeit
weniger dauerhafte Rechte
klarere Trennung von Aufgaben
```

Merksatz:

```
Nicht dauerhaft mit Adminrechten arbeiten.
```

Lokale Adminrechte

Lokale Adminrechte auf Clients sind riskant.

Risiken:

Malware kann mehr Schaden anrichten.
Benutzer kann Sicherheitssoftware deaktivieren.
Programme können ungeprüft installiert werden.
Einstellungen können verändert werden.
Angreifer kann Rechte ausnutzen.

Merksatz:

Lokale Adminrechte nur vergeben,
wenn sie wirklich nötig sind.

Domänenadministrator

Ein Domänenadministrator hat sehr weitreichende Rechte in einer Windows-Domäne.

Risiko:

Kompromittierung kann das gesamte Netzwerk betreffen.

Schutz:

nur wenige Personen
MFA
getrennte Admin-Konten
keine Nutzung für Alltag
Protokollierung
regelmäßige Rechteprüfung
sichere Admin-Arbeitsstationen

Merksatz:

Domänenadminrechte gehören zu den kritischsten Rechten.

Root

Root ist das höchste Benutzerkonto auf vielen Unix- und Linux-Systemen.

Root kann:

```
Dateien ändern
Benutzer verwalten
Dienste steuern
Rechte setzen
Systemkonfiguration ändern
Sicherheitsmechanismen beeinflussen
```

Merksatz:

```
Root hat volle Kontrolle über das System.
```

---

sudo

sudo erlaubt, einzelne Befehle mit erhöhten Rechten auszuführen.

Vorteile:

```
keine dauerhafte Root-Sitzung nötig
Aktionen besser nachvollziehbar
Rechte können gezielter vergeben werden
direkte Root-Anmeldung kann vermieden werden
```

Merksatz:

```
sudo ermöglicht kontrollierte Rechteerhöhung.
```

Dienstkonto

Ein Dienstkonto ist ein Konto, das von einem Dienst, einer Anwendung oder einem automatisierten Prozess genutzt wird.

Beispiele:

```
Datenbankzugriff einer Webanwendung
Backupdienst
Monitoring-Agent
```

Synchronisationsdienst
API-Integration
geplante Aufgabe

Merksatz:

Dienstkonten sind technische Identitäten für Dienste.

Risiken bei Dienstkonten

Dienstkonten sind oft kritisch, weil sie dauerhaft laufen.

Risiken:

zu viele Rechte
Passwort läuft nie ab
Passwort wird im Klartext gespeichert
Konto wird nicht überwacht
Konto bleibt nach Projektende aktiv
Nutzung ist schwer nachvollziehbar
Schlüssel oder Tokens werden nicht rotiert

Merksatz:

Dienstkonten müssen genauso verwaltet werden wie Benutzerkonten.

Dienstkonten sicher betreiben

Maßnahmen:

nur notwendige Rechte
eindeutiger Zweck
Dokumentation
keine interaktive Anmeldung,
wenn nicht nötig
starke Geheimnisse
Secret Manager nutzen
regelmäßige Rotation

Monitoring
Deaktivierung nach Nutzungsende
keine gemeinsamen Allzweckkonten

Merksatz:

Dienstkonto immer zweckgebunden und rechtearm einrichten.

Gruppenrichtlinien

Gruppenrichtlinien werden in Windows-Umgebungen genutzt, um Einstellungen zentral zu verwalten.

Beispiele:

Passwortregeln
Bildschirmsperre
Firewall-Einstellungen
Laufwerkszuordnung
Softwareverteilung
Sicherheitsoptionen
Einschränkung von Makros
lokale Administratoren
Update-Einstellungen

Merksatz:

Gruppenrichtlinien setzen zentrale Regeln für Benutzer und Computer.

Verzeichnisdienst

Ein Verzeichnisdienst verwaltet Identitäten und Ressourcen zentral.

Typische Inhalte:

Benutzer
Gruppen
Computer

Rollen
Organisationseinheiten
Richtlinien
Zugriffsrechte

Beispiele:

Active Directory
LDAP-basierte Verzeichnisse
Cloud-Verzeichnisdienste

Merksatz:

Verzeichnisdienst = zentrale Verwaltung von Identitäten und Objekten.

Active Directory

Active Directory ist ein Verzeichnisdienst von Microsoft.

Es wird häufig genutzt für:

Benutzerverwaltung
Gruppenverwaltung
Computerverwaltung
zentrale Anmeldung
Gruppenrichtlinien
Rechteverwaltung
Domänenstruktur

Merksatz:

Active Directory verwaltet Benutzer,
Gruppen,
Computer
und Richtlinien zentral.

LDAP

LDAP steht für:

```
Lightweight Directory Access Protocol
```

LDAP ist ein Protokoll, mit dem Verzeichnisdienste abgefragt und genutzt werden können.

Beispiele:

```
Benutzer suchen
Gruppen prüfen
Anmeldung gegen Verzeichnisdienst
Informationen aus Verzeichnis abrufen
```

Merksatz:

```
LDAP ist ein Protokoll für den Zugriff auf Verzeichnisdienste.
```

Authentifizierung

Authentifizierung bedeutet:

```
Nachweis der Identität.
```

Frage:

```
Wer bist du?
```

Beispiele:

```
Anmeldung mit Passwort
Anmeldung mit Zertifikat
Anmeldung mit MFA
Anmeldung per Smartcard
Anmeldung mit SSH-Schlüssel
```

Merksatz:

Authentifizierung prüft die Identität.

Autorisierung

Autorisierung bedeutet:

Prüfung der Berechtigung.

Frage:

Was darfst du?

Beispiele:

Darf Benutzer Datei öffnen?
Darf Benutzer Drucker verwalten?
Darf Benutzer Server neu starten?
Darf Benutzer Adminbereich nutzen?
Darf Benutzer Datenbank ändern?

Merksatz:

Autorisierung prüft die erlaubten Aktionen.

Authentifizierung und Autorisierung unterscheiden

Begriff	Frage	Beispiel
Authentifizierung	Wer bist du?	Benutzer meldet sich an
Autorisierung	Was darfst du?	Benutzer darf Datei lesen

Merksatz:

Erst Identität prüfen,
dann Rechte prüfen.

ACL

ACL steht für:

```
Access Control List
```

Eine ACL ist eine Zugriffskontrollliste.

Sie legt fest, wer auf ein Objekt zugreifen darf und welche Aktionen erlaubt sind.

Beispiele:

```
Datei lesen
Ordner schreiben
Freigabe nutzen
Netzwerkverkehr erlauben
Drucker verwalten
```

Merksatz:

```
ACL legt Zugriffsrechte auf Objekte fest.
```

Dateirechte

Dateirechte bestimmen, wer Dateien oder Ordner nutzen darf.

Typische Rechte:

```
lesen
schreiben
ausführen
ändern
löschen
Besitz übernehmen
Rechte ändern
```

Merksatz:

```
Dateirechte schützen Daten vor unberechtigtem Zugriff.
```

Lesen, Schreiben und Ausführen

Lesen:

```
Dateiinhalt anzeigen
```

Schreiben:

```
Datei verändern oder erstellen
```

Ausführen:

```
Programm oder Skript starten
```

Merksatz:

```
Lesen sieht Daten.
Schreiben verändert Daten.
Ausführen startet Code.
```

Vererbung von Rechten

Rechte können von übergeordneten Ordnern an Unterordner und Dateien vererbt werden.

Vorteil:

```
weniger Einzelkonfiguration
```

Risiko:

```
falsche Rechte können sich auf viele Dateien auswirken.
```

Merksatz:

```
Vererbte Rechte erleichtern Verwaltung,
können aber große Wirkung haben.
```

Freigaberechte und NTFS-Rechte

In Windows-Umgebungen gibt es oft:

```
Freigaberechte

und

NTFS-Rechte
```

Freigaberechte:

```
gelten beim Zugriff über Netzwerkfreigabe
```

NTFS-Rechte:

```
gelten auf Dateisystemebene
```

Wichtig:

```
Beim Netzwerkzugriff wirkt die restriktivere Kombination.
```

Merksatz:

```
Freigaberechte und NTFS-Rechte zusammen betrachten.
```

Besitz von Dateien

Der Besitzer einer Datei oder eines Ordners hat besondere Bedeutung.

Er kann je nach System Rechte beeinflussen oder ändern.

Wichtig:

```
Besitzrechte nicht unkontrolliert vergeben.
```

Merksatz:

```
Besitz kann Rechteverwaltung beeinflussen.
```

Umask

Umask ist ein Konzept in Unix- und Linux-Systemen.

Sie legt fest, welche Standardrechte bei neuen Dateien oder Ordnern entfernt werden.

Beispielprinzip:

```
Neue Datei erhält nicht automatisch alle möglichen Rechte,
weil umask bestimmte Rechte wegnimmt.
```

Merksatz:

```
umask beeinflusst Standardrechte neuer Dateien und Ordner.
```

---

chmod

chmod ist ein Befehl unter Unix- und Linux-Systemen, um Dateirechte zu ändern.

Rechtearten:

```
lesen
schreiben
ausführen
```

für:

```
Besitzer
Gruppe
andere
```

Merksatz:

```
chmod ändert Dateirechte.
```

---

chown

chown ist ein Befehl unter Unix- und Linux-Systemen, um Besitzer und Gruppe einer Datei oder eines Ordners zu ändern.

Merksatz:

```
chown ändert Besitzer oder Gruppe.
```

Dateirechte unter Linux: Grundidee

Linux unterscheidet bei Dateirechten typischerweise:

```
Besitzer

Gruppe

Andere
```

Rechte:

```
r = read = lesen

w = write = schreiben

x = execute = ausführen
```

Merksatz:

```
Linux-Dateirechte bestehen aus Besitzer,
Gruppe,
Andere
und den Rechten r,
w,
x.
```

Beispiel für Linux-Rechte

Beispiel:

```
rwxr-x---
```

Bedeutung:

```
Besitzer:
lesen,
schreiben,
ausführen

Gruppe:
lesen,
ausführen

Andere:
keine Rechte
```

Merksatz:

```
Linux-Rechte werden für Besitzer,
Gruppe
und Andere getrennt betrachtet.
```

Besondere Linux-Rechte

Neben normalen Rechten gibt es besondere Rechte.

Beispiele:

```
SetUID
SetGID
Sticky Bit
```

Diese Rechte können sicherheitskritisch sein, weil sie Ausführung, Gruppenzuordnung oder Löschverhalten beeinflussen.

Merksatz:

```
Sonderrechte unter Linux bewusst prüfen.
```

SetUID

SetUID bewirkt, dass ein Programm mit den Rechten des Dateibesitzers ausgeführt wird.

Risiko:

```
Wenn ein SetUID-Programm unsicher ist,
kann es Rechteausweitung ermöglichen.
```

Merksatz:

```
SetUID kann Rechte erhöhen und ist sicherheitskritisch.
```

SetGID

SetGID kann bewirken, dass ein Programm mit Gruppenrechten ausgeführt wird oder dass neue Dateien in einem Ordner eine bestimmte Gruppe erhalten.

Merksatz:

```
SetGID beeinflusst Gruppenrechte.
```

Sticky Bit

Sticky Bit wird häufig bei gemeinsam genutzten Verzeichnissen verwendet.

Es sorgt dafür, dass Benutzer dort nicht beliebig Dateien anderer Benutzer löschen können.

Beispiel:

```
/tmp
```

Merksatz:

```
Sticky Bit schützt Dateien in gemeinsamen Verzeichnissen vor fremdem Löschen.
```

Rechteprüfung in der Praxis

Bei Zugriffsproblemen prüfen:

```
Welcher Benutzer ist angemeldet?
In welchen Gruppen ist der Benutzer?
Welche Rechte hat die Datei?
Welche Rechte hat der Ordner?
Gibt es vererbte Rechte?
Gibt es Freigaberechte?
Gibt es NTFS-Rechte?
Gibt es ACLs?
Gibt es lokale oder zentrale Gruppen?
Gibt es Deny-Regeln?
Gibt es Sonderrechte?
```

Merksatz:

```
Zugriffsprobleme immer systematisch von Benutzer,
Gruppe,
Objekt
und Regel prüfen.
```

Deny-Regeln

Deny bedeutet:

```
Zugriff ausdrücklich verweigern.
```

Deny-Regeln können erlaubende Rechte übersteuern.

Risiko:

```
falsch gesetzte Deny-Regeln verursachen schwer erkennbare Zugriffsprobleme.
```

Merksatz:

```
Deny-Regeln sparsam und bewusst verwenden.
```

Zugriff verweigert

Wenn ein Benutzer „Zugriff verweigert“ erhält, kann das viele Ursachen haben.

Mögliche Ursachen:

Benutzer nicht in richtiger Gruppe
fehlende Dateirechte
fehlende Freigaberechte
Deny-Regel
falscher Besitzer
Vererbung unterbrochen
falsche Anmeldung
abgelaufenes Konto
gesperrtes Konto
fehlende Lizenz
Anwendung prüft eigene Rechte

Merksatz:

Zugriff verweigert ist nicht automatisch ein Passwortproblem.

Onboarding

Onboarding bedeutet:

neue Benutzer erhalten benötigte Zugänge,
Geräte
und Rechte.

Gutes Onboarding enthält:

persönliches Benutzerkonto
passende Gruppen
passende Rollen
MFA-Einrichtung
Geräteausgabe
Schulung
Passwortmanager
Sicherheitsrichtlinien

```
Dokumentation
```

Merksatz:

```
Onboarding vergibt Zugänge kontrolliert nach Aufgabe.
```

Rollenwechsel

Bei einem Rollenwechsel müssen Rechte angepasst werden.

Beispiel:

```
Benutzer wechselt von Support zu Einkauf.
```

Dann müssen alte Rechte entfernt und neue Rechte vergeben werden.

Wichtig:

```
Rechte nicht nur hinzufügen,
sondern auch nicht mehr benötigte entfernen.
```

Merksatz:

```
Rollenwechsel braucht Rechteentzug und Rechtevergabe.
```

Offboarding

Offboarding bedeutet:

```
Zugänge werden entfernt,
wenn Benutzer ausscheiden
oder keinen Zugriff mehr benötigen.
```

Zu prüfen:

```
Benutzerkonto deaktivieren
Gruppen entfernen
```

```
VPN-Zugang entfernen
E-Mail-Zugriff regeln
Cloud-Zugänge entfernen
SSH-Schlüssel löschen
API-Tokens widerrufen
Geräte zurückgeben
lokale Adminrechte entfernen
Weiterleitungen prüfen
```

Merksatz:

```
Offboarding verhindert alte,
vergessene Zugänge.
```

Regelmäßige Rechteprüfung

Rechte sollten regelmäßig überprüft werden.

Zu prüfen:

```
aktive Benutzer
deaktivierte Benutzer
Adminrechte
Gruppenmitgliedschaften
Dienstkonten
externe Dienstleister
lokale Administratoren
Cloud-Rollen
Freigaben
SSH-Schlüssel
```

Merksatz:

```
Rechte altern und müssen regelmäßig geprüft werden.
```

Rezertifizierung von Rechten

Rezertifizierung bedeutet:

Verantwortliche bestätigen regelmäßig,
ob Rechte noch benötigt werden.

Beispiel:

Abteilungsleiter prüft,
ob seine Mitarbeitenden noch Zugriff auf bestimmte Daten brauchen.

Merksatz:

Rezertifizierung verhindert angesammelte Altberechtigungen.

Rechteausweitung

Rechteausweitung bedeutet:

Ein Benutzer oder Angreifer erhält höhere Rechte als vorgesehen.

Arten:

vertikale Rechteausweitung:
normaler Benutzer wird Administrator

horizontale Rechteausweitung:
Benutzer greift auf Daten anderer Benutzer gleicher Ebene zu

Merksatz:

Rechteausweitung ist Zugriff über das erlaubte Maß hinaus.

Privilegieneskalation

Privilegieneskalation bedeutet:

Rechte werden erhöht,
zum Beispiel von normalem Benutzer zu Administrator.

Ursachen:

```
Schwachstelle
Fehlkonfiguration
unsichere Dienste
falsche Dateirechte
zu breite sudo-Regeln
unsichere Dienstkonten
```

Merksatz:

```
Privilegieneskalation ist Erhöhung von Rechten.
```

Horizontale Rechteausweitung

Horizontale Rechteausweitung bedeutet:

```
Ein Benutzer greift auf Daten eines anderen Benutzers mit ähnlicher Rolle zu.
```

Beispiel:

```
Benutzer 1001 sieht Rechnung von Benutzer 1002,
obwohl er nur eigene Rechnungen sehen darf.
```

Merksatz:

```
Horizontal bedeutet:
Zugriff auf fremde Daten gleicher Ebene.
```

Vertikale Rechteausweitung

Vertikale Rechteausweitung bedeutet:

```
Ein Benutzer erhält höhere Rechte.
```

Beispiel:

normaler Benutzer kann Adminfunktion ausführen.

Merksatz:

Vertikal bedeutet:
Zugriff auf höhere Berechtigungsstufe.

Auditierung von Rechten

Auditierung bedeutet:

Rechte und Zugriffe werden geprüft und dokumentiert.

Ziele:

Missbrauch erkennen
alte Rechte entfernen
Compliance erfüllen
Datenschutz verbessern
Nachvollziehbarkeit schaffen

Merksatz:

Rechte müssen nicht nur vergeben,
sondern auch geprüft werden.

Typische Praxisfehler bei Benutzerrechten

Häufige Fehler:

Rechte direkt auf einzelne Benutzer vergeben
zu viele lokale Adminrechte
Sammelkonten
alte Benutzer aktiv
Dienstkonten mit zu vielen Rechten
keine Dokumentation

keine regelmäßige Prüfung
Admin- und Alltagskonto nicht getrennt
Gruppen unübersichtlich
Deny-Regeln falsch gesetzt
Rechte bei Rollenwechsel nur hinzugefügt

Merksatz:

Rechteverwaltung wird unsicher,
wenn sie nicht gepflegt wird.

Checkliste: Rechte sauber verwalten

Persönliche Konten verwenden.
Sammelkonten vermeiden.
Gruppen statt Einzelrechte nutzen.
Rollen definieren.
Least Privilege anwenden.
Need to Know beachten.
Admin- und Alltagskonto trennen.
Dienstkonten dokumentieren.
Dienstkonten rechtearm einrichten.
Onboarding standardisieren.
Rollenwechsel prüfen.
Offboarding konsequent durchführen.
Rechte regelmäßig prüfen.
Adminrechte besonders kontrollieren.
Audit-Logs aktivieren.
Notfallkonten absichern.

Merksatz:

Gute Rechteverwaltung ist dauerhaft gepflegt,
dokumentiert
und nachvollziehbar.

Typische IHK-Fragen

In AP1 und AP2 kann zum Beispiel gefragt werden:

- Was ist ein Benutzerkonto?
- Warum sind persönliche Benutzerkonten wichtig?
- Warum sind Sammelkonten problematisch?
- Warum sollte man Rechte über Gruppen vergeben?
- Was ist eine Rolle?
- Was bedeutet RBAC?
- Was bedeutet ABAC?
- Was ist der Unterschied zwischen Authentifizierung und Autorisierung?
- Was bedeutet Least Privilege?
- Was bedeutet Need to Know?
- Warum sollten Admin- und Alltagskonto getrennt werden?
- Was ist ein Dienstkonto?
- Warum sind Dienstkonten kritisch?
- Was ist ein Verzeichnisdienst?
- Was ist Active Directory?
- Was ist LDAP?
- Was ist eine ACL?
- Was ist der Unterschied zwischen Freigaberechten und NTFS-Rechten?
- Was bedeutet chmod?
- Was bedeutet chown?
- Was ist SetUID?
- Was ist Offboarding?
- Warum ist regelmäßige Rechteprüfung wichtig?

Typische Prüfungsfallen

Authentifizierung und Autorisierung nicht verwechseln.

Benutzerkonto ist Identität,
Berechtigung ist erlaubte Aktion.

Rechte möglichst über Gruppen vergeben.

Rollen bündeln Rechte.

Sammelkonten vermeiden.

Least Privilege bedeutet minimale notwendige Rechte.

Need to Know bezieht sich auf Informationsbedarf.

Admin- und Alltagskonto trennen.

Lokale Adminrechte erhöhen Risiko.

Dienstkonten sind technische Identitäten.

Dienstkonten brauchen klare Rechtebegrenzung.

Active Directory ist ein Verzeichnisdienst.

LDAP ist ein Protokoll für Verzeichniszugriff.

ACL ist Zugriffskontrollliste.

Freigaberechte und NTFS-Rechte zusammen betrachten.

Deny-Regeln können erlaubende Regeln übersteuern.

Zugriff verweigert kann viele Ursachen haben.

Rollenwechsel bedeutet alte Rechte entfernen.

Offboarding muss alle Zugänge entfernen.

Rechte regelmäßig prüfen.

SetUID ist sicherheitskritisch.

Wichtige Begriffe kurz erklärt

Begriff	Kurze Erklärung
Benutzerkonto	digitale Identität eines Benutzers
persönliches Konto	Konto für genau eine Person
Sammelkonto	gemeinsam genutztes Konto

Begriff	Kurze Erklärung
Gruppe	Zusammenfassung von Benutzern
Rolle	Aufgabe mit passenden Rechten
RBAC	rollenbasierte Zugriffskontrolle
ABAC	attributbasierte Zugriffskontrolle
Berechtigung	erlaubte Aktion
Recht	einzelne Erlaubnis
Least Privilege	minimale notwendige Rechte
Need to Know	Zugriff nur bei Informationsbedarf
Aufgabentrennung	kritische Aufgaben trennen
Vier-Augen-Prinzip	Prüfung durch zweite Person
Adminkonto	Konto mit erhöhten Rechten
lokaler Administrator	Admin auf einem einzelnen System
Domänenadministrator	Admin mit weitreichenden Domänenrechten
Root	höchstes Konto auf Unix/Linux
sudo	kontrollierte Rechteerhöhung
Dienstkonto	technische Identität für Dienste
Gruppenrichtlinie	zentrale Windows-Einstellung
Verzeichnisdienst	zentrale Verwaltung von Identitäten
Active Directory	Microsoft-Verzeichnisdienst
LDAP	Protokoll für Verzeichnisdienste
Authentifizierung	Identität prüfen
Autorisierung	Berechtigung prüfen
ACL	Zugriffskontrollliste
Dateirechte	Rechte auf Dateien und Ordner
Vererbung	Weitergabe von Rechten
Freigaberechte	Rechte auf Netzwerkfreigabe
NTFS-Rechte	Rechte auf Windows-Dateisystemebene
umask	Standardrechte unter Unix/Linux beeinflussen
chmod	Dateirechte ändern
chown	Besitzer oder Gruppe ändern
SetUID	Ausführung mit Besitzerrechten
SetGID	Ausführung oder Dateien mit Gruppenbezug

Begriff	Kurze Erklärung
Sticky Bit	Schutz in gemeinsamen Verzeichnissen
Deny	ausdrückliche Verweigerung
Onboarding	geregelte Zugangsvergabe
Offboarding	geregelter Rechteentzug
Rezertifizierung	regelmäßige Bestätigung von Rechten
Rechteausweitung	Zugriff über erlaubtes Maß hinaus
Privilegieneskalation	Erhöhung von Rechten
Auditierung	Prüfung und Dokumentation

IHK-sichere Kurzformulierung

Benutzer-, Gruppen- und Rechteverwaltung sorgt dafür, dass Personen und Dienste nur die Zugriffe erhalten, die sie für ihre Aufgaben benötigen. Benutzer sollten persönliche Konten verwenden, Sammelkonten sollten vermieden werden. Rechte werden möglichst über Gruppen und Rollen vergeben, zum Beispiel nach dem RBAC-Prinzip. Authentifizierung prüft die Identität eines Benutzers, Autorisierung prüft seine Berechtigungen. Wichtige Grundsätze sind Least Privilege, Need to Know, Trennung von Admin- und Alltagskonto, kontrollierte Dienstkonten, regelmäßige Rechteprüfung und konsequentes Offboarding. Dateirechte, ACLs, Freigaberechte, NTFS-Rechte und Linux-Rechte müssen systematisch geprüft werden, wenn Zugriffsprobleme auftreten.

Merksätze

Rechteverwaltung schützt Daten und Systeme.

Benutzerkonto = digitale Identität.

Ein Benutzer,
ein persönliches Konto.

Sammelkonten vermeiden.

Rechte möglichst über Gruppen vergeben.

Rolle beschreibt Aufgabe.

RBAC vergibt Rechte nach Rollen.

ABAC entscheidet nach Eigenschaften.

Berechtigung = erlaubte Aktion.

Rechte sind einzelne Erlaubnisse.

Rollen bündeln Rechte.

Least Privilege = minimale notwendige Rechte.

Need to Know = Zugriff nur bei Bedarf.

Adminrechte besonders schützen.

Admin- und Alltagskonto trennen.

Lokale Adminrechte vermeiden.

Root hat volle Kontrolle.

sudo ermöglicht kontrollierte Rechteerhöhung.

Dienstkonten sind technische Identitäten.

Dienstkonten rechtearm betreiben.

Gruppenrichtlinien setzen zentrale Regeln.

Verzeichnisdienste verwalten Identitäten zentral.

Active Directory ist ein Verzeichnisdienst.

LDAP ist ein Protokoll für Verzeichniszugriff.

Authentifizierung fragt:
Wer bist du?

Autorisierung fragt:
Was darfst du?

ACL legt Zugriffe fest.

Lesen sieht Daten.

Schreiben verändert Daten.

Ausführen startet Code.

Freigaberechte und NTFS-Rechte zusammen prüfen.

Linux-Rechte bestehen aus Besitzer,
Gruppe,
Andere
und r,
w,
x.

chmod ändert Rechte.

chown ändert Besitzer oder Gruppe.

SetUID ist sicherheitskritisch.

Deny-Regeln bewusst einsetzen.

Zugriff verweigert hat viele mögliche Ursachen.

Onboarding vergibt Rechte kontrolliert.

Rollenwechsel braucht Rechteentzug.

Offboarding entfernt alte Zugänge.

Rechte regelmäßig prüfen.

Rezertifizierung verhindert Altberechtigungen.

Privilegieneskalation erhöht Rechte.

Gute Rechteverwaltung ist nachvollziehbar,
dokumentiert

und regelmäßig geprüft.