17.2 SSH, RDP und sichere Fernadministration im Detail

Fernadministration bedeutet, dass Systeme aus der Ferne verwaltet werden.

Dafür werden Protokolle, Zugriffswege, Benutzerkonten, Berechtigungen und Sicherheitsmaßnahmen kombiniert.

Typische Fernadministration:

```
- Linux-Server per SSH verwalten
- Windows-Server per RDP verwalten
- Firewall über Weboberfläche konfigurieren
- Switch per SSH administrieren
- NAS oder Router über Managementoberfläche verwalten
- Cloud-Systeme über Portal oder API steuern
- Zugriff über VPN oder Bastion Host absichern
```

Merksatz:

```
Fernadministration ist praktisch,
aber sicherheitskritisch.
```

Warum Fernadministration abgesichert werden muss

Remotezugänge sind besonders interessant für Angreifer, weil sie direkten Zugriff auf Systeme ermöglichen können.

Risiken:

```
Passwortangriffe
Brute Force
Credential Stuffing
Phishing
gestohlene SSH-Schlüssel
offene RDP-Ports
veraltete Remote-Dienste
fehlende MFA
```

schwache Firewall-Regeln
kompromittierte Admin-Konten

Merksatz:

Ein unsicherer Fernzugriff kann zum Einstiegspunkt für einen kompletten Angriff werden.

Grundregel für Fernadministration

Adminzugänge sollten nicht unnötig direkt aus dem Internet erreichbar sein.

Besser:

Zugriff über VPN

oder

Zugriff über Bastion Host

oder

Zugriff nur von festen Quell-IP-Adressen

oder

Zugriff über ein getrenntes Managementnetz

Merksatz:

Adminzugänge nicht breit öffentlich bereitstellen.

SSH

SSH steht für:

Secure Shell

SSH ist ein verschlüsseltes Protokoll für entfernten Zugriff auf eine Kommandozeile.

SSH wird häufig genutzt für:

```
Linux-Server
Unix-Systeme
Netzwerkgeräte
Firewalls
Switches
Router
NAS-Systeme
Git-Zugriffe
Dateiübertragung mit SFTP oder SCP
```

Standardport:

```
TCP 22
```

Merksatz:

```
SSH ist sicherer Fernzugriff auf die Kommandozeile.
```

Was SSH schützt

SSH schützt die Verbindung durch Verschlüsselung.

Dadurch werden geschützt:

```
Zugangsdaten bei der Übertragung
Befehle
Ausgaben
Dateiübertragungen
Sitzungsdaten
```

Wichtig:

```
SSH verschlüsselt die Verbindung,
aber schützt nicht automatisch vor schwachen Passwörtern,
gestohlenen Schlüsseln
oder falsch vergebenen Rechten.
```

Merksatz:

```
SSH schützt den Transport,
aber nicht jede Fehlkonfiguration.
```

SSH-Authentifizierung

SSH kann verschiedene Anmeldearten nutzen.

Typisch sind:

```
Passwortauthentifizierung

oder

Schlüsselbasierte Authentifizierung
```

Zusätzlich möglich:

```
MFA
Zertifikatsbasierte SSH-Anmeldung
zentrale Benutzerverwaltung
Zugriff über Bastion Host
```

Merksatz:

```
SSH braucht starke Authentifizierung,
nicht nur Verschlüsselung.
```

SSH mit Passwort

Bei SSH mit Passwort meldet sich ein Benutzer mit Benutzername und Passwort an.

Vorteil:

```
einfach einzurichten
```

Nachteile:

```
anfällig gegen Brute Force
anfällig gegen gestohlene Passwörter
Risiko durch Passwortwiederverwendung
schwierig bei vielen Servern sicher zu verwalten
```

Merksatz:

```
SSH-Passwortlogin ist einfach,
aber sicherheitlich schwächer als Schlüsselanmeldung.
```

SSH mit Schlüsselpaar

Bei SSH mit Schlüsselpaar gibt es zwei Schlüssel:

```
privater Schlüssel

und

öffentlicher Schlüssel
```

Der öffentliche Schlüssel liegt auf dem Server.

Der private Schlüssel bleibt beim Administrator.

Nur wer den passenden privaten Schlüssel besitzt, kann sich authentifizieren.

Merksatz:

```
Öffentlicher Schlüssel auf den Server,
privater Schlüssel bleibt geheim beim Benutzer.
```

Privater Schlüssel

Der private SSH-Schlüssel ist geheim.

Er darf nicht:

```
weitergegeben werden
in Git-Repositories liegen
in Tickets kopiert werden
in Chatnachrichten stehen
unverschlüsselt auf fremden Systemen liegen
auf gemeinsam genutzten Netzlaufwerken liegen
```

Merksatz:

```
Der private SSH-Schlüssel ist wie ein besonders kritisches Passwort.
```

Öffentlicher Schlüssel

Der öffentliche SSH-Schlüssel ist nicht geheim.

Er wird auf Systemen hinterlegt, auf die der Benutzer zugreifen darf.

Trotzdem muss er verwaltet werden.

Wichtig:

```
alte Schlüssel entfernen
Schlüssel Personen zuordnen
Schlüssel bei Austritt löschen
keine unbekannten Schlüssel erlauben
Berechtigungen regelmäßig prüfen
```

Merksatz:

```
Öffentliche Schlüssel sind nicht geheim,
aber ihre Zuordnung ist sicherheitsrelevant.
```

Passphrase für SSH-Schlüssel

Ein privater SSH-Schlüssel sollte mit einer Passphrase geschützt werden.

Vorteil:

```
Wenn der Schlüssel gestohlen wird,
kann er nicht sofort genutzt werden.
```

Nachteil:

```
Benutzer muss Passphrase eingeben
oder sicher mit einem Agent arbeiten.
```

Merksatz:

```
Passphrase schützt den privaten Schlüssel zusätzlich.
```

SSH-Agent

Ein SSH-Agent kann private Schlüssel nach Eingabe der Passphrase temporär bereithalten.

Vorteil:

```
Benutzer muss Passphrase nicht bei jeder Verbindung neu eingeben.
```

Risiko:

```
ein kompromittiertes System kann unter Umständen den Agent missbrauchen.
```

Merksatz:

```
SSH-Agent ist praktisch,
muss aber bewusst genutzt werden.
```

SSH-Agent-Forwarding

SSH-Agent-Forwarding leitet die Authentifizierung über ein Zwischensystem weiter.

Beispiel:

```
Administrator
→
```

```
Bastion Host
→
Zielserver
```

Risiko:

```
Wenn der Bastion Host kompromittiert ist,
kann Agent-Forwarding missbraucht werden.
```

Besser je nach Umgebung:

```
ProxyJump
kurzlebige Zertifikate
Bastion-Lösung mit Protokollierung
gezielte Schlüsselverwaltung
```

Merksatz:

```
Agent-Forwarding nur nutzen,
wenn Risiko und Bedarf klar sind.
```

SSH über Bastion Host

Ein Bastion Host ist ein geschützter Sprungserver.

Vorteile:

```
Zielserver müssen nicht öffentlich erreichbar sein.
Zugriff kann zentral protokolliert werden.
Firewall-Regeln werden einfacher.
Adminzugriffe laufen kontrollierter.
```

Beispielprinzip:

```
Administrator verbindet sich zuerst zum Bastion Host.
Danach erfolgt der Zugriff auf interne Server.
```

Merksatz:

Bastion Host schützt interne Systeme vor direktem Adminzugriff aus dem Internet.

SSH ProxyJump

ProxyJump ist eine SSH-Funktion, um über einen Zwischensystem zu einem Zielserver zu gelangen.

Prinzip:

```
lokaler Rechner
→
Bastion Host
→
Zielserver
```

Vorteil:

Zielserver muss nur vom Bastion Host erreichbar sein.

Merksatz:

ProxyJump ermöglicht SSH-Zugriff über einen Sprungserver.

Root-Login per SSH

Direkter Root-Login über SSH ist riskant.

Nachteile:

Angreifer kennt den Benutzernamen root.
Aktionen sind schwerer einzelnen Personen zuzuordnen.
Root hat sofort volle Rechte.
Brute-Force-Ziel ist eindeutig.

Besser:

Anmeldung mit persönlichem Benutzerkonto

danach

Rechteerhöhung mit sudo

Merksatz:

Direkten Root-Login vermeiden,
persönliche Konten und sudo nutzen.

---

sudo bei Fernadministration

sudo erlaubt, einzelne Befehle mit erhöhten Rechten auszuführen.

Vorteile:

Adminrechte nur bei Bedarf
bessere Nachvollziehbarkeit
weniger Risiko durch Alltagssitzungen
gezieltere Rechtevergabe
Protokollierung möglich

Merksatz:

sudo trennt normale Anmeldung und administrative Rechte.

SSH sicher konfigurieren

Wichtige Maßnahmen:

Passwortlogin deaktivieren,
wenn Schlüsselbetrieb sauber eingerichtet ist

direkte Root-Anmeldung deaktivieren

nur benötigte Benutzer erlauben

SSH-Zugriff per Firewall begrenzen

```
MFA prüfen

starke Schlüssel verwenden

alte Schlüssel entfernen

Loginversuche überwachen

Systeme aktuell halten

Logs auswerten
```

Merksatz:

```
SSH-Sicherheit entsteht durch Schlüssel,
Rechte,
Regeln
und Überwachung.
```

SSH-Logs

SSH-Logs zeigen Anmeldeversuche und Fehler.

Typische Hinweise:

```
erfolgreiche Anmeldung
fehlgeschlagene Anmeldung
ungültiger Benutzer
falscher Schlüssel
falsches Passwort
viele Versuche von einer IP
Verbindungsabbrüche
```

Merksatz:

```
SSH-Logs helfen,
Angriffe und Fehlkonfigurationen zu erkennen.
```

Typische SSH-Fehlerbilder

Häufige Ursachen:

```
falscher Benutzername
falscher Hostname
falsche IP-Adresse
Firewall blockiert TCP 22
SSH-Dienst läuft nicht
Schlüssel nicht hinterlegt
falsche Dateirechte beim Schlüssel
privater Schlüssel passt nicht
Root-Login deaktiviert
Passwortlogin deaktiviert
Server-Key hat sich geändert
```

Merksatz:

```
SSH-Fehler können an Netzwerk,
Dienst,
Benutzer,
Schlüssel
oder Berechtigung liegen.
```

Host Key bei SSH

Ein SSH-Server besitzt einen Host Key.

Dieser dient dazu, den Server wiederzuerkennen.

Beim ersten Verbinden wird der Host Key gespeichert.

Wenn sich der Host Key später unerwartet ändert, erscheint eine Warnung.

Mögliche Ursachen:

```
Server wurde neu installiert
Host Key wurde erneuert
falscher Server wird erreicht
Man-in-the-Middle-Angriff möglich
```

Merksatz:

```
SSH-Host-Key-Warnungen nicht blind ignorieren.
```

RDP

RDP steht für:

```
Remote Desktop Protocol
```

RDP ermöglicht grafischen Fernzugriff auf Windows-Systeme.

Standardport:

```
TCP 3389
```

Typische Nutzung:

```
Windows-Server administrieren
Remote-Desktop-Sitzung
Zugriff auf grafische Anwendungen
Fernwartung interner Systeme
```

Merksatz:

```
RDP ist grafischer Fernzugriff auf Windows-Systeme.
```

RDP-Risiken

RDP ist ein häufiges Angriffsziel.

Risiken:

```
Brute Force
Credential Stuffing
Exploits gegen ungepatchte Systeme
Ransomware-Einstieg
```

Passwortdiebstahl
fehlende MFA
unsichere Freigabe ins Internet
schwache Kontosperrregeln

Merksatz:

RDP offen im Internet ist ein hohes Risiko.

RDP sicher betreiben

Schutzmaßnahmen:

RDP nur über VPN
RDP Gateway nutzen
MFA nutzen
Network Level Authentication aktivieren
Quell-IP einschränken
starke Passwörter
Account Lockout
Updates einspielen
Adminrechte begrenzen
Logs überwachen

Merksatz:

RDP gehört hinter VPN,
Gateway
oder andere Zugriffsschutzmaßnahmen.

Network Level Authentication

Network Level Authentication wird abgekürzt:

NLA

NLA verlangt Authentifizierung, bevor eine vollständige RDP-Sitzung aufgebaut wird.

Vorteile:

```
reduziert Angriffsfläche
spart Serverressourcen
erschwert bestimmte Angriffe
Anmeldung erfolgt früher im Verbindungsaufbau
```

Merksatz:

```
NLA schützt RDP,
bevor die vollständige Sitzung entsteht.
```

RDP Gateway

Ein RDP Gateway bündelt RDP-Zugriffe.

Statt viele Systeme direkt per RDP erreichbar zu machen, verbinden sich Benutzer zum Gateway.

Vorteile:

```
zentraler Zugriffspunkt
bessere Protokollierung
MFA möglich
interne Systeme bleiben privater
Regeln zentral steuerbar
```

Merksatz:

```
RDP Gateway reduziert direkte RDP-Erreichbarkeit interner Systeme.
```

RDP und VPN

RDP sollte häufig nur nach VPN-Verbindung erreichbar sein.

Vorteile:

```
RDP-Port nicht öffentlich
Benutzer vorher authentifiziert
```

Zugriff kann auf interne Netze begrenzt werden
MFA kann am VPN greifen
Firewall-Regeln einfacher

Wichtig:

VPN ersetzt keine RDP-Sicherheit.
Benutzerrechte und Logs bleiben wichtig.

Merksatz:

VPN schützt den Zugang,
aber nicht automatisch das Zielsystem.

RDP-Logs

RDP-Zugriffe sollten protokolliert werden.

Zu prüfen:

erfolgreiche Anmeldungen
fehlgeschlagene Anmeldungen
ungewöhnliche Uhrzeiten
unbekannte Quelladressen
viele Fehlversuche
neue Benutzer
Admin-Anmeldungen
Sitzungsabbrüche

Merksatz:

RDP-Logs sind wichtig für Angriffserkennung und Nachvollziehbarkeit.

Webbasierte Administrationsoberflächen

Viele Systeme haben Weboberflächen.

Beispiele:

```
Firewall
Router
Switch
NAS
Drucker
Hypervisor
Backup-System
Monitoring-System
Cloud-Dienst
Container-Plattform
```

Merksatz:

```
Weboberflächen sind bequem,
aber oft sehr mächtig.
```

Risiken von Admin-Weboberflächen

Typische Risiken:

```
öffentlich erreichbar
Standardpasswort aktiv
fehlende MFA
veraltete Firmware
unsicheres TLS
schwache Rollenverwaltung
XSS oder CSRF
unsichere Sessionverwaltung
fehlende Logs
```

Merksatz:

```
Admin-Weboberflächen nicht ungeschützt bereitstellen.
```

Admin-Weboberflächen absichern

Maßnahmen:

nur intern erreichbar machen
Zugriff über VPN
Quell-IP beschränken
MFA aktivieren
HTTPS nutzen
gültige Zertifikate einsetzen
Standardkonten deaktivieren
starke Passwörter
Updates einspielen
Rollen begrenzen
Logs aktivieren

Merksatz:

Admin-Weboberflächen brauchen Zugriffsschutz,
Verschlüsselung,
Updates
und Logging.

VPN als Zugriffsschutz

VPN kann Fernzugriff absichern, indem interne Dienste nicht direkt öffentlich erreichbar sind.

Vorteile:

zentrale Authentifizierung
MFA möglich
verschlüsselter Tunnel
interne Adressen bleiben privat
Zugriff über Gruppen steuerbar
Logs zentral auswertbar

Merksatz:

VPN ist ein sicherer Zugangspfad,
aber kein Ersatz für Berechtigungen.

Bastion Host und Jump Server

Bastion Host oder Jump Server dienen als kontrollierter Sprungpunkt.

Typische Eigenschaften:

```
besonders gehärtet
stark überwacht
nur für Administration gedacht
MFA geschützt
protokolliert Zugriffe
hat eingeschränkten Zugriff auf Zielsysteme
```

Merksatz:

```
Bastion Host und Jump Server bündeln administrative Zugriffe.
```

Managementnetz

Ein Managementnetz trennt administrative Zugriffe vom normalen Benutzernetz.

Darin liegen:

```
Switch-Management
Firewall-Management
Server-Management
Hypervisor-Management
Storage-Management
Backup-Management
Out-of-Band-Management
```

Merksatz:

```
Managementschnittstellen gehören nicht ins normale Clientnetz.
```

Out-of-Band-Management

Out-of-Band-Management nutzt einen getrennten Verwaltungsweg.

Beispiele:

```
iLO
iDRAC
IPMI
serielle Konsole
separates Managementinterface
```

Vorteil:

```
Zugriff auch bei Störungen des normalen Systems möglich
```

Risiko:

```
extrem mächtiger Zugriff
```

Merksatz:

```
Out-of-Band-Management besonders stark schützen.
```

API-Zugriff für Administration

Viele moderne Systeme werden über APIs verwaltet.

Beispiele:

```
Cloud-APIs
Firewall-APIs
Monitoring-APIs
Backup-APIs
Container-APIs
Automatisierungsschnittstellen
```

Risiken:

```
API-Schlüssel gestohlen
Token zu viele Rechte
fehlendes Rate Limiting
unsichere Speicherung
```

keine Protokollierung
Schlüssel nicht rotiert

Merksatz:

API-Zugänge sind Adminzugänge und müssen genauso geschützt werden.

API-Schlüssel und Tokens

API-Schlüssel und Tokens erlauben Zugriff auf Systeme.

Sie dürfen nicht:

im Quellcode stehen
in öffentlichen Repositories liegen
in Tickets kopiert werden
in Chatverläufen stehen
unverschlüsselt gespeichert werden
dauerhaft ohne Prüfung aktiv bleiben

Merksatz:

API-Schlüssel wie Passwörter behandeln.

Sichere Speicherung von Zugangsdaten

Zugangsdaten sollten sicher verwaltet werden.

Geeignete Mittel:

Passwortmanager
Secret Manager
PAM-System
verschlüsselte Tresore
zentrale Identitätsverwaltung
Zugriff nach Rollen
regelmäßige Prüfung

Ungeeignet:

```
Textdateien
Excel-Listen ohne Schutz
Screenshots
Chatnachrichten
Notizzettel
Quellcode
gemeinsam genutzte Klartextdokumente
```

Merksatz:

```
Zugangsdaten gehören in sichere Verwaltung,
nicht in Klartextablagen.
```

Least Privilege bei Fernadministration

Fernzugriffe sollten nur die Rechte erhalten, die für die Aufgabe nötig sind.

Beispiele:

```
Support darf Dienste neu starten,
aber keine Firewall löschen.

Backup-Operator darf Sicherungen prüfen,
aber keine Benutzer anlegen.

Netzwerkadmin darf Switches verwalten,
aber keine Personaldaten lesen.
```

Merksatz:

```
Remotezugriff immer nach Aufgabe und Rolle begrenzen.
```

Zeitlich begrenzte Zugriffe

Administrative Zugriffe sollten möglichst nicht dauerhaft bestehen.

Möglichkeiten:

Just-in-Time-Rechte
temporäre VPN-Gruppen
befristete Dienstleisterzugänge
zeitlich begrenzte Adminrollen
Ablaufdatum für Konten
automatische Deaktivierung

Merksatz:

Dauerhafte Adminrechte erhöhen das Risiko.

Dienstleisterzugänge

Externe Dienstleisterzugänge sind besonders kritisch.

Zu beachten:

klare Freigabe
begrenzte Rechte
zeitliche Befristung
MFA
Protokollierung
keine Sammelkonten
Zugriff nur auf benötigte Systeme
Abschaltung nach Auftrag
vertragliche Regelungen

Merksatz:

Dienstleisterzugänge müssen begrenzt,
überwacht
und wieder entfernt werden.

MFA für Fernadministration

MFA sollte besonders für Remotezugriffe genutzt werden.

Wichtig für:

VPN
Cloud-Portal
RDP Gateway
Admin-Weboberflächen
PAM
Bastion Host
Identitätsdienst
kritische APIs

Merksatz:

Fernadministration ohne MFA ist ein erhöhtes Risiko.

Logging bei Fernadministration

Protokolliert werden sollten:

wer sich angemeldet hat
wann Zugriff erfolgte
von welcher Quelle
auf welches Ziel
welche Aktion ausgeführt wurde
ob Zugriff erfolgreich war
ob Fehler auftraten
welche Rechte genutzt wurden

Merksatz:

Fernadministration muss nachvollziehbar sein.

Session Recording

Session Recording bedeutet:

administrative Sitzungen werden aufgezeichnet.

Möglich bei:

```
Bastion Hosts
PAM-Systemen
RDP Gateways
SSH-Proxies
Remote-Support-Systemen
```

Nutzen:

```
Nachvollziehbarkeit
forensische Analyse
Schulung
Missbrauchserkennung
```

Merksatz:

```
Session Recording erhöht Nachvollziehbarkeit bei kritischen Zugriffen.
```

Timeouts und automatische Trennung

Remote-Sitzungen sollten bei Inaktivität getrennt werden.

Vorteile:

```
weniger Risiko bei vergessenen Sitzungen
geringere Angriffsfläche
weniger Missbrauch offener Sessions
```

Beispiele:

```
SSH-Timeout
RDP-Session-Timeout
VPN-Timeout
Websession-Timeout
Adminportal-Timeout
```

Merksatz:

Offene Admin-Sitzungen nicht unbegrenzt laufen lassen.

Sperren nach Fehlversuchen

Viele fehlgeschlagene Loginversuche können auf Angriffe hinweisen.

Schutz:

Account Lockout
Rate Limiting
IP-Blockierung
Fail2ban oder vergleichbare Mechanismen
MFA
Alarmierung

Wichtig:

Sperrregeln sinnvoll einstellen,
damit kein einfacher Denial-of-Service gegen Benutzerkonten entsteht.

Merksatz:

Fehlversuche begrenzen,
aber Sperrlogik kontrolliert planen.

Fail2ban als Prinzip

Fail2ban ist ein Beispiel für einen Schutzmechanismus, der Logs auswertet und bei vielen Fehlversuchen Quellen blockieren kann.

Prinzip:

Log prüfen
Muster erkennen
IP-Adresse temporär sperren

Merksatz:

```
Logbasierte Sperren können Brute-Force-Angriffe bremsen.
```

Fernadministration und Change Management

Fernadministration sollte nicht bedeuten, dass Änderungen unkontrolliert erfolgen.

Auch remote gilt:

```
Änderung planen
Risiko bewerten
Wartungsfenster prüfen
Backup oder Snapshot prüfen
Rollback vorbereiten
Änderung dokumentieren
Funktion testen
Logs kontrollieren
```

Merksatz:

```
Remoteänderungen brauchen denselben Prozess wie lokale Änderungen.
```

Fernadministration und Notfallzugriff

Für Notfälle braucht es definierte Zugriffswege.

Beispiele:

```
Break-Glass-Konto
Out-of-Band-Zugang
Ersatz-VPN
lokale Konsole
Notfallpasswort im Tresor
dokumentierte Wiederherstellung
```

Wichtig:

```
Notfallzugang stark schützen
Nutzung protokollieren
```

regelmäßig testen
nicht im Alltag verwenden

Merksatz:

Notfallzugriff vorbereiten,
aber streng kontrollieren.

Typische Fehler bei Fernadministration

Häufige Fehler:

RDP direkt im Internet offen
SSH mit schwachem Passwort
Root-Login erlaubt
private Schlüssel ungeschützt
alte SSH-Schlüssel nicht entfernt
Admin-Weboberfläche öffentlich
keine MFA
keine Logs
Sammelkonten
Adminrechte dauerhaft vergeben
Dienstleisterzugang nicht deaktiviert
Managementnetz fehlt
API-Schlüssel im Code

Merksatz:

Viele Remoteprobleme entstehen durch Bequemlichkeit und fehlende Kontrolle.

Checkliste: sichere Fernadministration

Remotezugänge erfassen.
Öffentliche Erreichbarkeit prüfen.
Nur notwendige Ports freigeben.
VPN oder Bastion Host nutzen.
MFA aktivieren.

Persönliche Konten verwenden.
Sammelkonten vermeiden.
Admin- und Benutzerkonto trennen.
Least Privilege umsetzen.
SSH-Schlüssel verwalten.
Root-Login vermeiden.
RDP nicht direkt öffentlich machen.
Weboberflächen absichern.
Logs aktivieren.
Fehlversuche überwachen.
Notfallzugänge schützen.
Dienstleisterzugänge befristen.
Änderungen dokumentieren.
Zugriffe regelmäßig prüfen.

Merksatz:

Sichere Fernadministration braucht Technik,
Rechtekonzept,
Protokollierung
und klare Prozesse.

Typische IHK-Fragen

In AP1 und AP2 kann zum Beispiel gefragt werden:

- Was ist SSH?
- Wofür wird SSH verwendet?
- Welchen Standardport nutzt SSH?
- Warum sind SSH-Schlüssel sicherer als reine Passwortanmeldung?
- Was ist der Unterschied zwischen privatem und öffentlichem SSH-Schlüssel?
- Warum sollte der private SSH-Schlüssel geschützt werden?
- Was ist ein SSH Host Key?
- Warum sollte Root-Login per SSH vermieden werden?
- Was ist sudo?
- Was ist RDP?
- Welchen Standardport nutzt RDP?
- Warum sollte RDP nicht direkt aus dem Internet erreichbar sein?

- Was ist Network Level Authentication?
- Was ist ein RDP Gateway?
- Was ist ein Bastion Host?
- Was ist ein Jump Server?
- Was ist ein Managementnetz?
- Was ist Out-of-Band-Management?
- Warum sind API-Schlüssel wie Passwörter zu behandeln?
- Warum ist MFA bei Fernadministration wichtig?
- Warum ist Logging bei Remotezugriffen wichtig?

Typische Prüfungsfallen

SSH ist verschlüsselt,
aber nicht automatisch sicher konfiguriert.

SSH nutzt standardmäßig TCP 22.

RDP nutzt standardmäßig TCP 3389.

RDP offen im Internet ist riskant.

VPN ersetzt keine Benutzerrechte.

MFA ersetzt kein Rechtekonzept.

SSH-Schlüssel brauchen Schutz.

Privater Schlüssel bleibt geheim.

Öffentlicher Schlüssel gehört auf Zielsysteme.

Host-Key-Warnungen nicht blind ignorieren.

Root-Login vermeiden.

sudo kontrolliert Rechteerhöhung.

Bastion Host ist ein geschützter Sprungserver.

Managementnetz vom Clientnetz trennen.

Out-of-Band-Zugriff ist besonders kritisch.

Admin-Weboberflächen nicht öffentlich bereitstellen.

API-Tokens wie Zugangsdaten behandeln.

Dienstleisterzugänge befristen.

Logs müssen aktiv sein,
bevor ein Vorfall passiert.

Session Recording kann Nachvollziehbarkeit erhöhen.

Notfallzugang ist kein Alltagszugang.

Wichtige Begriffe kurz erklärt

Begriff	Kurze Erklärung
Fernadministration	Verwaltung von Systemen aus der Ferne
SSH	verschlüsselter Kommandozeilenzugriff
SSH-Port	TCP 22
Passwortauthentifizierung	Anmeldung mit Passwort
Schlüsselbasierte Authentifizierung	Anmeldung mit Schlüsselpaar
privater Schlüssel	geheimer Schlüsselteil beim Benutzer
öffentlicher Schlüssel	Schlüsselteil auf dem Server
Passphrase	Schutzwort für privaten Schlüssel
SSH-Agent	hält Schlüssel temporär bereit
Agent-Forwarding	Weiterleitung der SSH-Authentifizierung
Bastion Host	geschützter Sprungserver
ProxyJump	SSH-Verbindung über Zwischensystem
Root-Login	direkte Anmeldung als Root
sudo	kontrollierte Rechteerhöhung
Host Key	Identität des SSH-Servers
RDP	grafischer Windows-Fernzugriff

Begriff	Kurze Erklärung
RDP-Port	TCP 3389
NLA	Network Level Authentication
RDP Gateway	zentraler RDP-Zugriffspunkt
Admin-Weboberfläche	Verwaltungssystem im Browser
VPN	verschlüsselter Zugang ins interne Netz
Jump Server	Zwischensystem für Administration
Managementnetz	getrenntes Verwaltungsnetz
Out-of-Band-Management	Verwaltung über separaten Weg
API-Zugriff	Administration über Programmierschnittstelle
API-Schlüssel	Zugangsschlüssel für API
Token	Zugriffsnachweis für Dienste
Secret Manager	sichere Verwaltung geheimer Werte
PAM	Verwaltung privilegierter Zugriffe
Just-in-Time	zeitlich begrenzte Rechte
Dienstleisterzugang	externer Administrationszugang
Session Recording	Aufzeichnung administrativer Sitzungen
Timeout	automatische Trennung bei Inaktivität
Account Lockout	Kontosperre nach Fehlversuchen
Rate Limiting	Begrenzung von Anfragen
Fail2ban	logbasierte Sperre bei Fehlversuchen
Break-Glass-Konto	Notfallzugang

IHK-sichere Kurzformulierung

Fernadministration ermöglicht die Verwaltung von Systemen aus der Ferne, ist aber sicherheitskritisch. SSH dient dem verschlüsselten Kommandozeilenzugriff und nutzt standardmäßig TCP-Port 22. Eine schlüsselbasierte SSH-Anmeldung ist sicherer als reine Passwortanmeldung, wenn der private Schlüssel geschützt und alte Schlüssel entfernt werden. RDP dient dem grafischen Fernzugriff auf Windows-Systeme und nutzt standardmäßig TCP-Port 3389. RDP sollte nicht direkt aus dem Internet erreichbar sein, sondern über VPN, RDP Gateway oder andere Zugriffsschutzmaßnahmen abgesichert werden. Für sichere Fernadministration sind MFA, Least Privilege, persönliche Konten, Logging, Bastion Hosts, Managementnetze, abgesicherte Weboberflächen, geschützte API-Schlüssel und dokumentierte Notfallzugänge wichtig.

Merksätze

Fernadministration ist praktisch,
aber sicherheitskritisch.

Adminzugänge nicht breit öffentlich bereitstellen.

SSH = Secure Shell.

SSH nutzt TCP 22.

SSH verschlüsselt den Transport.

SSH braucht trotzdem sichere Authentifizierung.

Passwortlogin ist einfacher,
aber schwächer.

SSH-Schlüssel bestehen aus privatem und öffentlichem Schlüssel.

Privater Schlüssel bleibt geheim.

Öffentlicher Schlüssel liegt auf dem Server.

Passphrase schützt privaten Schlüssel.

Agent-Forwarding bewusst nutzen.

Bastion Host schützt interne Systeme.

ProxyJump verbindet über Sprungserver.

Root-Login vermeiden.

sudo für kontrollierte Rechteerhöhung nutzen.

SSH-Logs prüfen.

Host-Key-Warnungen ernst nehmen.

RDP = Remote Desktop Protocol.

RDP nutzt TCP 3389.

RDP nicht direkt ins Internet öffnen.

RDP über VPN oder Gateway absichern.

NLA schützt RDP vor vollständigem Sitzungsaufbau.

Admin-Weboberflächen nicht öffentlich bereitstellen.

VPN schützt Zugang,
aber ersetzt keine Rechte.

Managementnetz trennt Verwaltungszugriffe.

Out-of-Band-Management besonders schützen.

API-Schlüssel wie Passwörter behandeln.

Secrets sicher speichern.

Least Privilege auch remote anwenden.

Dienstleisterzugänge befristen.

MFA für Fernadministration nutzen.

Remotezugriffe protokollieren.

Session Recording kann Nachvollziehbarkeit erhöhen.

Timeouts für Admin-Sitzungen setzen.

Fehlversuche begrenzen.

Notfallzugänge streng kontrollieren.

Sichere Fernadministration braucht Technik,
Prozesse
und Kontrolle.