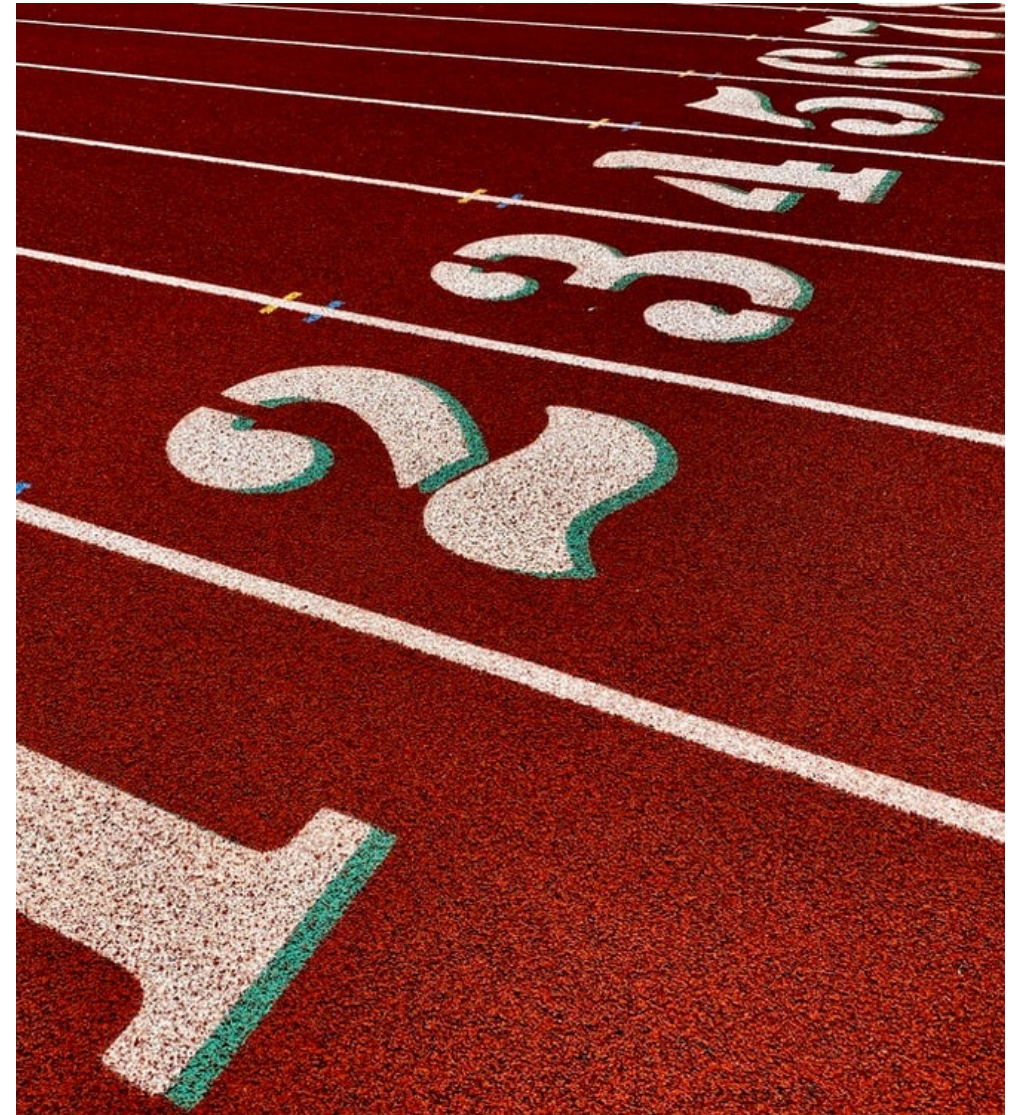


THE SECURE API SDLC

OWASP API Security Top 10

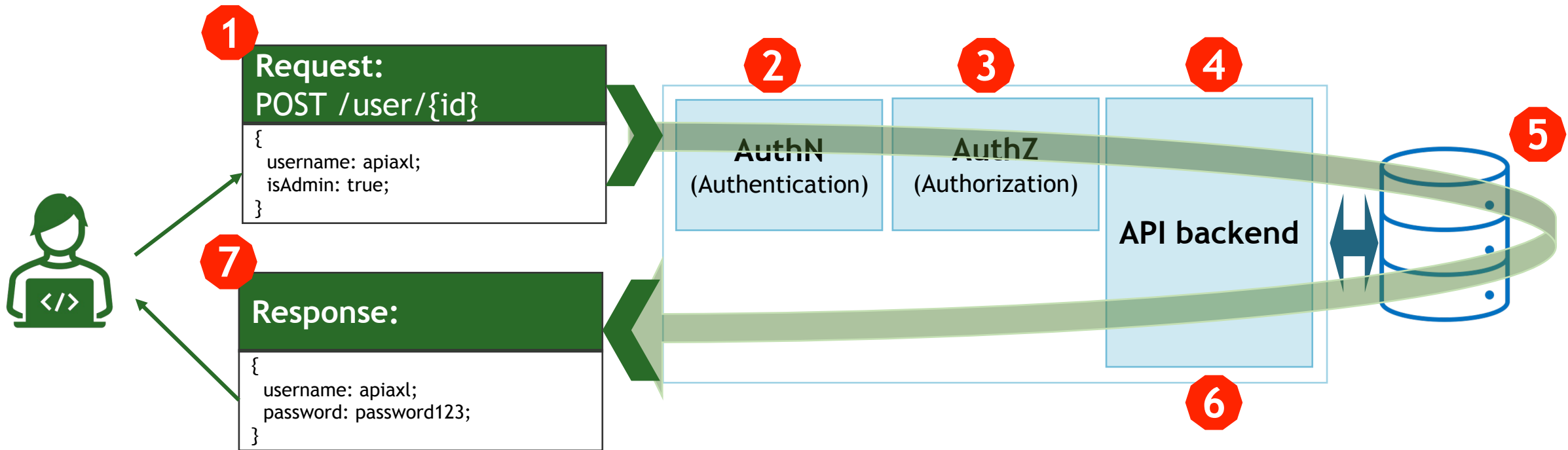
Modul 2



API security is different to web security

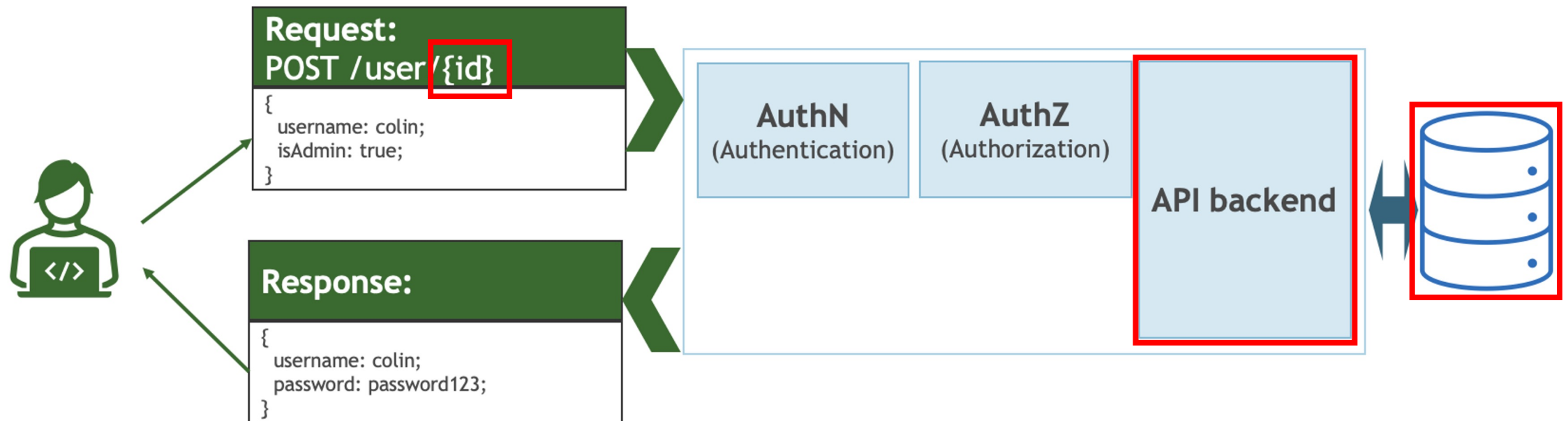
OWASP API Security Top 10	OWASP Top 10
API1:2019 Broken Object Level Authorization	A01:2021-Broken Access Control
API2:2019 Broken User Authentication	A02:2021-Cryptographic Failures
API3:2019 Excessive Data Exposure	A03:2021-Injection
API4:2019 Lack of Resources & Rate Limiting	A04:2021-Insecure Design
API5:2019 Broken Function Level Authorization	A05:2021-Security Misconfiguration
API6:2019 Mass Assignment	A06:2021-Vulnerable and Outdated Components
API7:2019 Security Misconfiguration	A07:2021-Identification and Authentication Failures
API8:2019 Injection	A08:2021-Software and Data Integrity Failures
API9:2019 Improper Assets Management	A09:2021-Security Logging and Monitoring Failures
API10:2019 Insufficient Logging & Monitoring	A10:2021-Server-Side Request Forgery

How does a REST API work?



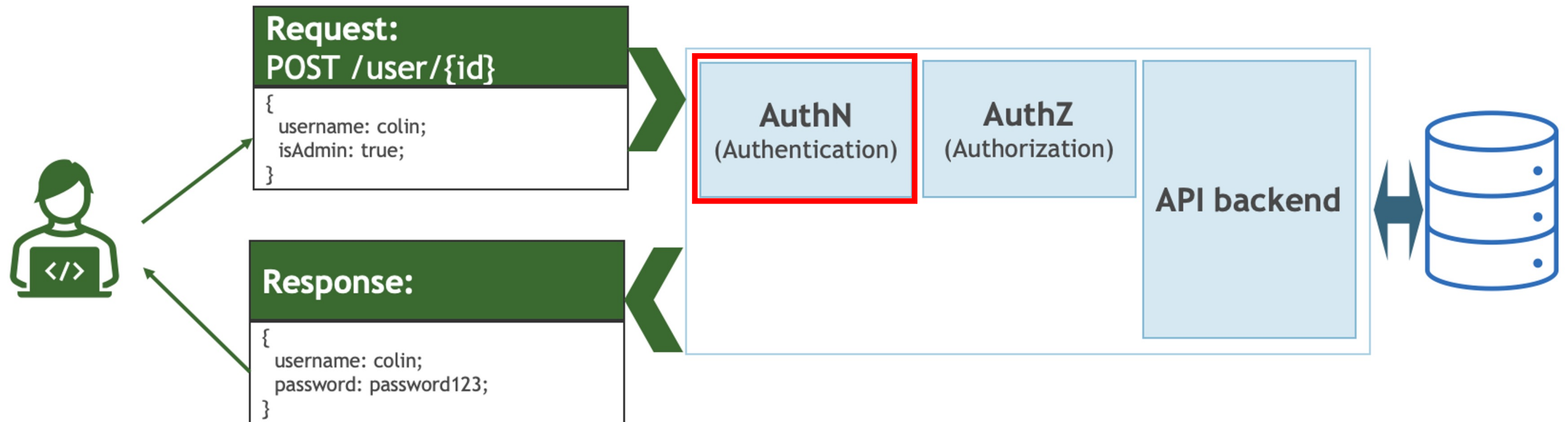
API1 — BROKEN OBJECT LEVEL AUTHORIZATION

Angreifer ersetzt die ID in der URL durch eine eigene um an die Daten eines anderen Nutzers zu kommen



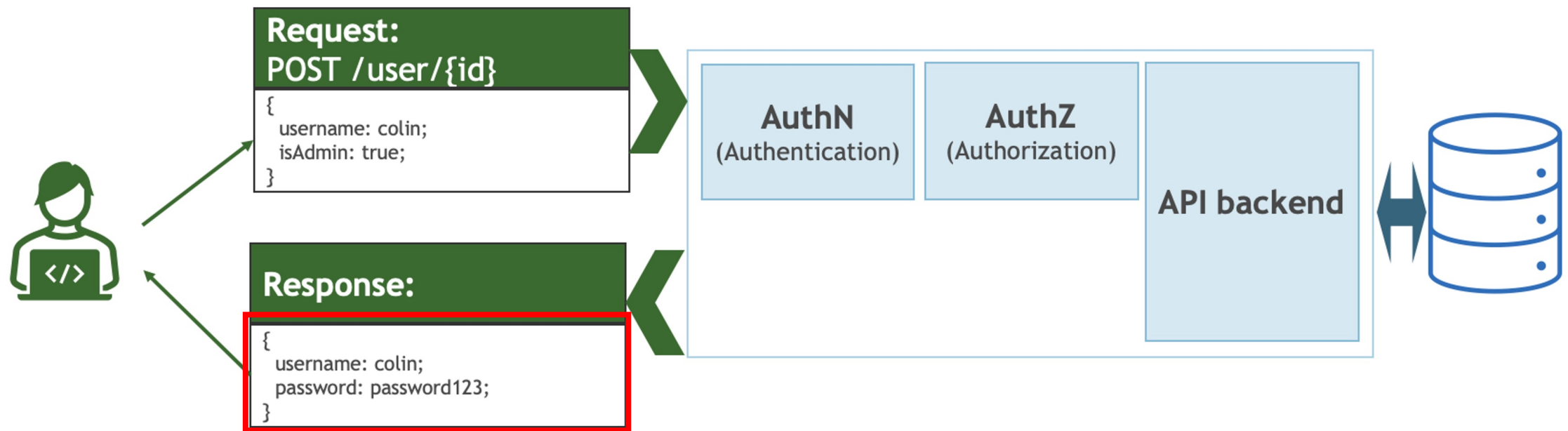
API 2: BROKEN AUTHENTICATION

Schlecht implementierte API Authentifizierung erlaubt Identitätsdiebstahl



API 3: DATA/EXCEPTION LEAKAGE

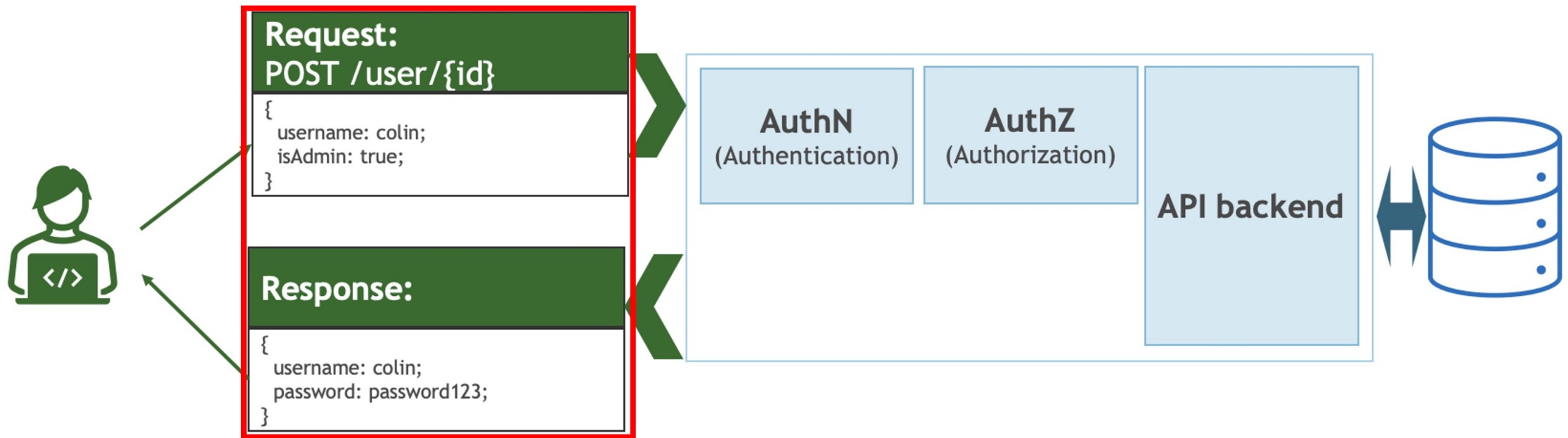
Die API exponiert mehr Daten als für den aktuellen Konsumenten nötig sind



<https://apisecurity.io/encyclopedia/content/owasp/api3-excessive-data-exposure>

API 4: RESOURCES PROTECTION/RATE LIMITING

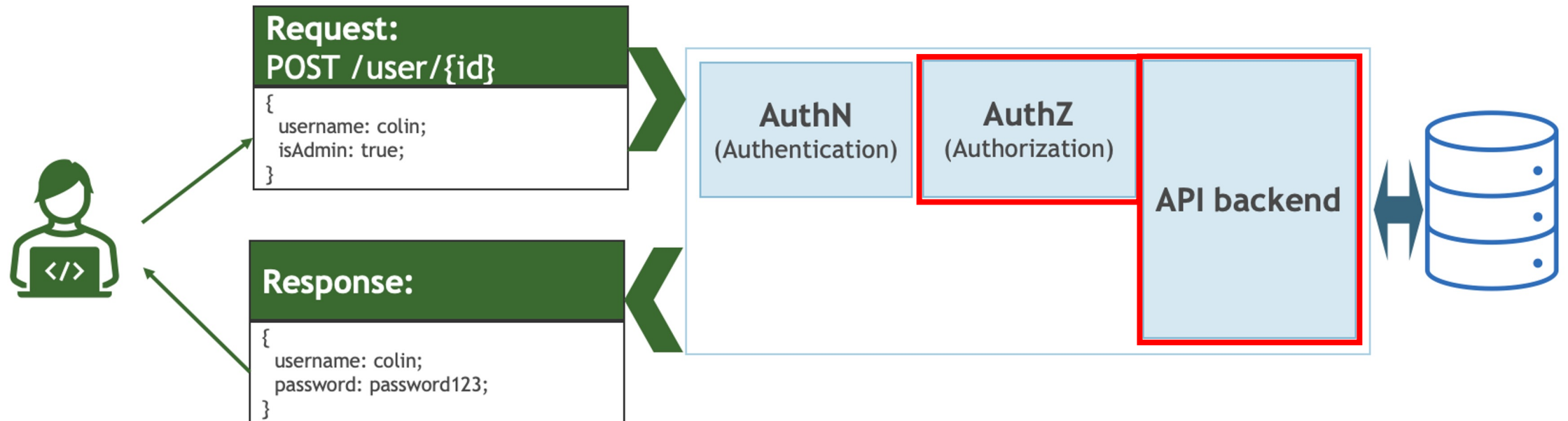
Die API hat kein Aufruflimit und/oder nicht genug Ressourcen



<https://apisecurity.io/encyclopedia/content/owasp/api4-lack-of-resources-and-rate-limiting>

API 5: BROKEN FUNCTION LEVEL AUTH

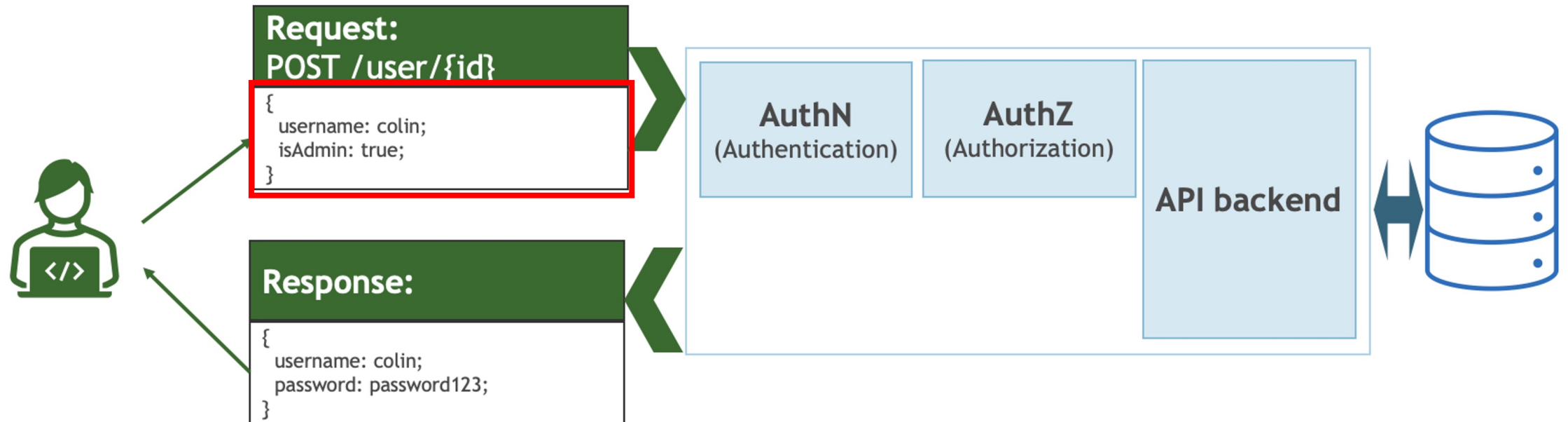
Die API verlässt sich auf die Client App zwischen ADMIN und NON-ADMIN zu unterscheiden
Der Angreifer identifiziert die versteckte ADMIN Funktionalität.



<https://apisecurity.io/encyclopedia/content/owasp/api5-broken-function-level-authorization>

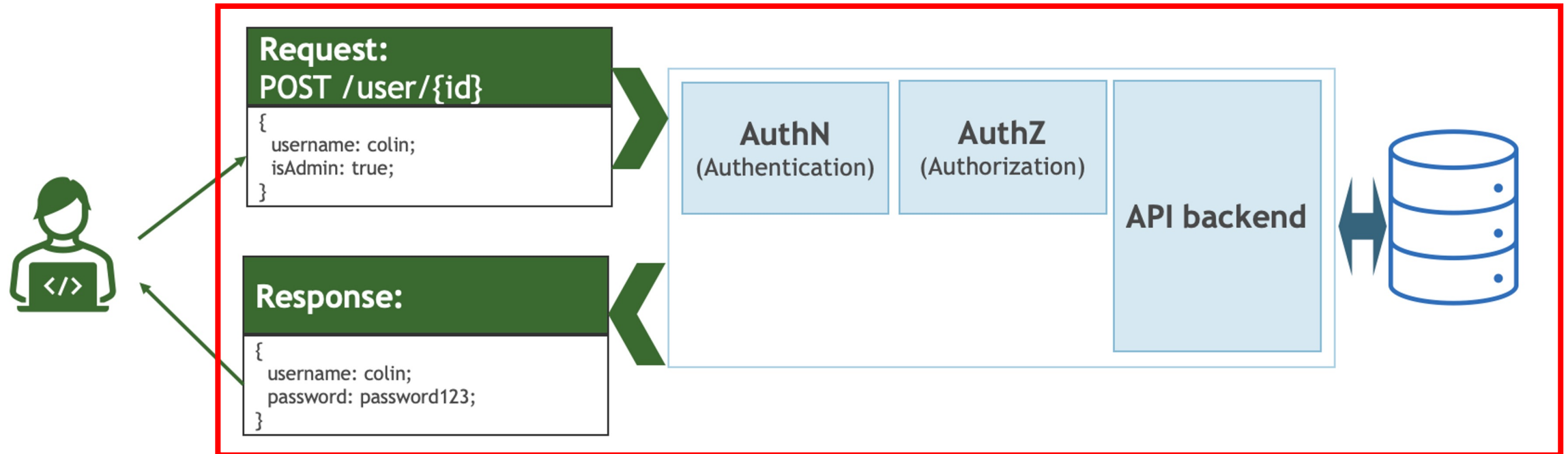
API 6: MASS ASSIGNMENT

Die API akzeptiert Daten ungeprüft und speichert sie direkt.
Angreifer kann anhand der Daten weitere Attribute errahnen und nutzen



API 7: SECURITY MISCONFIGURATION

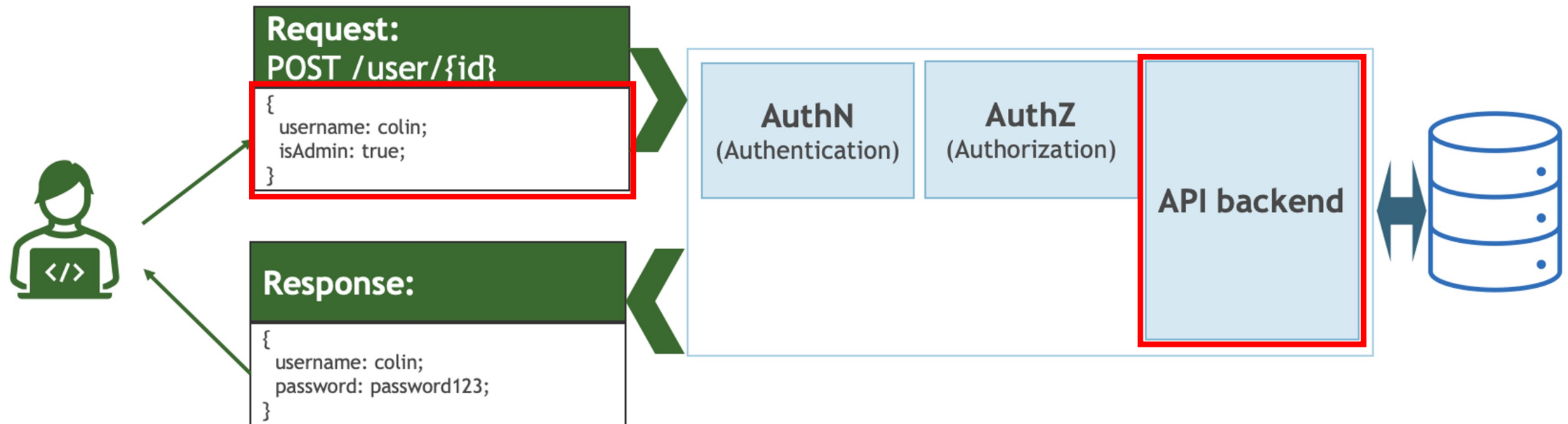
Schlecht konfigurierte API Server exponieren Daten und Funktionalitäten die nicht extern erreichbar sein sollen



<https://apisecurity.io/encyclopedia/content/owasp/api7-security-misconfiguration>

API 8: INJECTION

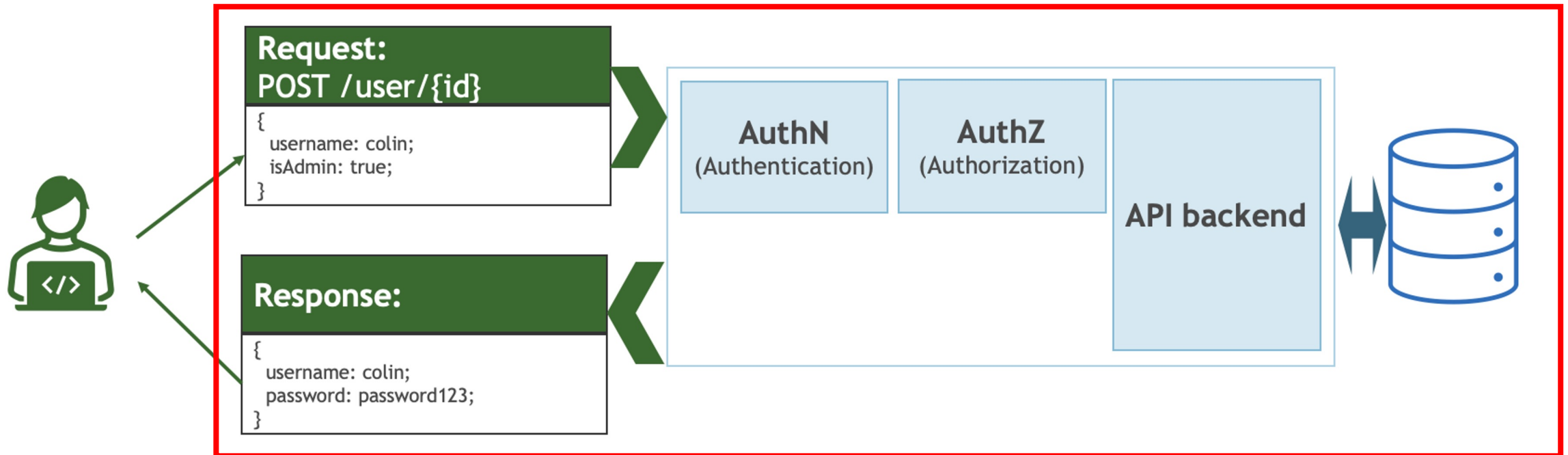
Angreifer nutzen Parameter um schädliche SQL, NoSQL, LDAP oder OS befehle einfügen



<https://apisecurity.io/encyclopedia/content/owasp/api8-injection>

API 9: IMPROPER ASSETS MANAGEMENT

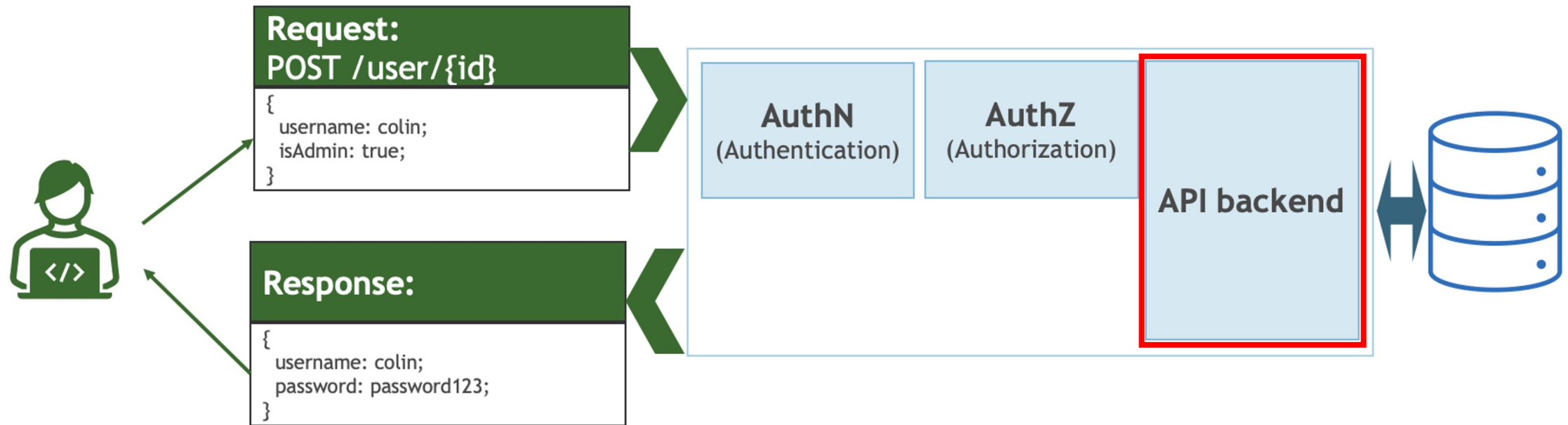
Angreifer finden Non-production oder alte API Versionen
mit niederschwelliger Sicherheitsvorgaben



<https://apisecurity.io/encyclopedia/content/owasp/api9-improper-assets-management>






API 10: LOGGING / MONITORING

Schlechtes Logging, Monitoring erlaubt Angreifer unerkannt zu bleiben.



<https://apisecurity.io/encyclopedia/content/owasp/api10-insufficient-logging-and-monitoring>

Wo bin als Entwickler involviert

OWASP API Security Top 10	
	API1:2019 Broken Object Level Authorization
	API2:2019 Broken User Authentication
	API3:2019 Excessive Data Exposure
	API4:2019 Lack of Resources & Rate Limiting
	API5:2019 Broken Function Level Authorization
	API6:2019 Mass Assignment
	API7:2019 Security Misconfiguration
	API8:2019 Injection
	API9:2019 Improper Assets Management
	API10:2019 Insufficient Logging & Monitoring