

30 Praktijkuitdagingen

- | | | | |
|----|---|----|--|
| 01 | Netwerken en Adressering | 11 | Versiebeheer en DevOps |
| 02 | Webtechnologie en Protocollen | 12 | Container Orchestration & Cloud |
| 03 | Databases en Datastructuren | 13 | DevOps & Automatisering |
| 04 | Systeemonderdelen en Geheugen | 14 | Infrastructuur & Webserver |
| 05 | Besturingssystemen en Beveiliging | 15 | Systeembeveiliging & Toegangsbeheer |
| 06 | Concurrency en Threading | 16 | Applicatiebeveiliging & DevSecOps |
| 07 | API-architectuur en Integratie | 17 | Datastructuren & Performance |
| 08 | Cryptografie en Authenticatie | 18 | Monitoring & Observability |
| 09 | Algoritmen en Complexiteit | 19 | Databases & Databeheer |
| 10 | Containerisatie en Cloud Infrastructure | 20 | Disaster Recovery & Bedrijfscontinuïteit |

Netwerken en Adressering

1 Uitdaging 01: Binaire handberekening van IP-subnets

Scenario: Een host heeft IP 192.168.10.77/27. Bereken zonder rekenmachine het netwerkadres, het broadcastadres en het aantal bruikbare hostadressen. Tijdsdoel: 5 minuten. Verificatiecriterium: Het netwerkadres is exact 192.168.10.64, broadcast is 192.168.10.95 en het aantal bruikbare hosts is 30.

| Zelfstandige vaardigheidstest voor subnetcalculatie onder tijdsdruk.

2 Uitdaging 02: DNS-foutopsporing via de terminal

Scenario: Gebruikers melden dat 'api.voorbeeld.nl' niet bereikbaar is. Onderzoek via 'dig' of 'nslookup' of het A-record, CNAME of de gezaghebbende nameserver faalt. Tijdsdoel: 7 minuten. Verificatiecriterium: Correcte identificatie van de ontbrekende DNS-recordwaarde en de responderende DNS-server.

| Diagnostische checklist voor netwerk- en applicatiebeheerders.

3 Uitdaging 03: TCP three-way handshake traceren

Scenario: Analyseer een PCAP-bestand in Wireshark of tshark en vind de SYN, SYN-ACK en ACK packets tussen client en webserver. Tijdsdoel: 6 minuten. Verificatiecriterium: Rapporteer de exacte timestamp, bronpoort en initiële sequence numbers (ISN) van beide endpoints.

| Validatieoefening voor netwerkanalyse en protocolkennis.

Webtechnologie en Protocollen

4 Uitdaging 04: HTTP-statuscodes en headers diagnosticeren

Scenario: Een endpoint resulteert in ERR_TOO_MANY_REDIRECTS. Analyseer met curl de HTTP-headers en statuscodes (301, 302, 307, 308). Tijdsdoel: 5 minuten. Verificatiecriterium: Wijs de exacte header-loop aan inclusief de Location-parameters die de oneindige cyclus veroorzaken.

| Snelle triage bij webrouting- en configuratiefouten.

5 Uitdaging 05: TLS-certificaatketen valideren

Scenario: Verifieer via 'openssl s_client' de certificaatketen van een externe server. Controleer verloopdatum, tussenliggende CA's en Subject Alternative Names. Tijdsdoel: 6 minuten.

Verificatiecriterium: Bevestig geldigheidstatus, resterende dagen en correcte CA-vertrouwensketen.

| Toegangscontrole en beveiligingsauditing voor webserver.

Databases en Datastructuren

6 Uitdaging 06: B-Tree vs. Hash-index selectie

Scenario: Een tabel met 5 miljoen rijen voert intensieve bereikqueries (BETWEEN) en exacte zoekopdrachten uit op twee kolommen. Bepaal het optimale indextype per kolom. Tijdsdoel: 4 minuten. Verificatiecriterium: Correcte toewijzing van B-Tree voor bereikselecties en Hash voor gelijkheid, onderbouwd met O-notatie.

| Database-optimalisatietest voor data engineers en DBA's.

7 Uitdaging 07: SQL JOIN-strategie en EXPLAIN-analyse

Scenario: Analyseer de uitvoeringsplannen van INNER, LEFT en FULL JOIN over twee gerelateerde tabellen. Voorspel vooraf het type join en de scanmethode, en verifieer de daadwerkelijke uitvoer via EXPLAIN. Tijdsdoel: 8 minuten. Verificatiecriterium: De voorspelde en daadwerkelijke plannen voor alle drie de joins zijn correct gedocumenteerd en sequential scans zijn geïdentificeerd.

| Praktijktoets voor database query tuning.

8 Uitdaging 08: ACID-transactie-isolatie-niveau bepalen

Scenario: Twee gelijktijdige transacties veroorzaken inconsistenties in rekeningsaldi. Selecteer het minimale isolatieniveau om non-repeatable reads te voorkomen. Tijdsdoel: 5 minuten.

Verificatiecriterium: Correcte keuze voor 'Repeatable Read' met toelichting waarom 'Read Committed' ontoereikend is.

| Architectuurcheck voor financiële en bedrijfskritische gegevensopslag.

Systeemonderdelen en Geheugen

9 Uitdaging 09: Stack versus Heap geheugeninspectie

Scenario: Analyseer een C/Rust-codefragment. Bepaal voor zes variabelen of deze op de stack of heap leven en identificeer een potentiële dangling pointer. Tijdsdoel: 7 minuten. Verificatiecriterium: Foutloze categorisering van alle zes variabelen en exacte markering van de out-of-scope pointer.

| Systeemprogrammeertest voor code reviews en debugging.

10 Uitdaging 10: Processen en CPU context switches traceren

Scenario: Een server vertoont hoge CPU-load terwijl applicatiethreads wachten. Gebruik 'vmstat' of 'pidstat' om context switches en runqueue-lengte te analyseren. Tijdsdoel: 6 minuten.

Verificatiecriterium: Bepaal of het knelpunt ligt bij voluntary context switches (I/O) of CPU-saturatie.

| Prestatiemeting en OS-level resource troubleshooting.

Besturingssystemen en Beveiliging

11 Uitdaging 11: Linux bestandsrechten en SUID herstellen

Scenario: Een serviceaccount kan een configuratiebestand niet lezen en een onbevoegd script heeft 4755-rechten. Pas rechten aan naar strikt noodzakelijk (least privilege). Tijdsdoel: 5 minuten.

Verificatiecriterium: Geen SUID-bit op het script, bestand ingesteld op 0640 met juiste chown-eigenaar.

| Beveiligingscontrole voor Linux systeembeheerders.

Concurrency en Threading

12 Uitdaging 12: Deadlock-scenario diagnosticeren

Scenario: Twee threads lopen vast. Onderzoek een thread stackdump waarin Thread A wacht op Lock 2 (in bezit van Thread B) en vice versa. Tijdsdoel: 6 minuten. Verificatiecriterium: Teken de wait-for graph en noteer de specifieke lock-volgorde die de deadlock definitief opheft.

| Probleemanalyse bij multithreaded backend-diensten.

13 Uitdaging 13: Race condition oplossen met atomaire operaties

Scenario: Een webapp verliest factuurnummers bij gelijktijdige requests door 'counter++'. Vervang de logica door een atomaire primitief. Tijdsdoel: 5 minuten. Verificatiecriterium: Correct gebruik van AtomicInteger / Compare-And-Swap (CAS) zonder gedeelde ongebalanceerde mutaties.

| Codeverbetering voor schaalbare gelijktijdige verwerking.

API-architectuur en Integratie

14 Uitdaging 14: REST API foutafhandeling en idempotentie

Scenario: Een betalings-API ontvangt dubbele POST-aanroepen bij netwerkuitval. Ontwerp een idempotente afhandeling en kies statuscodes voor dubbele transacties. Tijdsdoel: 7 minuten. Verificatiecriterium: Implementatie van Idempotency-Key header en correcte toekenning van 200/201 versus 409 Conflict.

| Architectuurtoets voor veilige financiële API-interfaces.

Cryptografie en Authenticatie

15 Uitdaging 15: Asymmetrische cryptografie en RSA-handtekening

Scenario: Gegeven een publieke en privésleutel. Toon aan welke sleutel wordt gebruikt voor authenticiteit (ondertekening) en welke voor vertrouwelijkheid (encryptie). Tijdsdoel: 4 minuten. Verificatiecriterium: Foutloze demonstratie dat handtekening = encryptie met privésleutel en verificatie = decryptie met publieke sleutel.

| Kennisvalidatie voor cryptografische implementaties.

16 Uitdaging 16: JWT payload decoding en handtekeningverificatie

Scenario: Analyseer een verdacht JWT-token. Controleer header, payload en handtekening op 'alg: none' manipulatie en de claim 'exp'. Tijdsdoel: 6 minuten. Verificatiecriterium: Identificeer tokenmanipulatie en verifieer dat de handtekening afgewezen wordt bij algoritme-mismatch.

| Beveiligingstoets voor authenticatietokens en API-gateways.

Algoritmen en Complexiteit

17 Uitdaging 17: Big-O complexiteit van sorteeralgoritmen

Scenario: Vergelijk QuickSort, MergeSort en HeapSort op een reeds nagenoeg gesorteerde dataset. Bepaal de worst-case looptijden en geheugenbeslag. Tijdsdoel: 5 minuten.

Verificatiecriterium: Benoem correct $O(n^2)$ worst-case voor naïeve Quicksort en $O(n \log n)$ met $O(n)$ hulpgeheugen voor Mergesort.

| Selectiemodel voor algoritmekeuze in datapijplijnen.

Containerisatie en Cloud Infrastructure

18 Uitdaging 18: Dockerfile layer caching en omvang minimaliseren

Scenario: Een build duurt 12 minuten omdat broncode vóór package-installatie wordt gekopieerd.

Herschrijf de Dockerfile met multi-stage build. Tijdsdoel: 8 minuten. Verificatiecriterium: Dependency manifests gekopieerd vóór 'RUN install', resulterend in hergebruik van de cache bij codewijzigingen.

| Optimalisatietest voor CI/CD-pijplijnen en cloud builds.

19 Uitdaging 19: Container CPU- en geheugenlimieten instellen

Scenario: Een microservice wordt onverwacht beëindigd door de Linux OOM Killer. Analyseer de 'requests' en 'limits' van het pod-manifest. Tijdsdoel: 6 minuten. Verificatiecriterium: Stel de geheugenlimiet in op 1.5x piekverbruik en configureer geschikte CPU-quota zonder agressieve throttling.

| Stabiliteitscontrole voor cluster- en microservicebeheerders.

Versiebeheer en DevOps

20 Uitdaging 20: Git Merge Conflict en Rebase strategie

Scenario: Een feature branch loopt 14 commits achter op 'main' en heeft conflicterende regels in een configuratiebestand. Voer een rebase uit en los het conflict op. Tijdsdoel: 7 minuten.

Verificatiecriterium: Een schone, lineaire git-historie zonder merge commit en met behoud van de geteste configuratie.

| Praktijkvaardigheidstest voor teamontwikkeling en CI-hygiëne.

Container Orchestration & Cloud

21 Uitdaging 21: Kubernetes Pod CrashLoopBackOff diagnosticeren

Scenario: Een backend-pod bevindt zich in de status CrashLoopBackOff na een recente deployment. Gebruik CLI-tools zoals `kubectl describe pod` en `kubectl logs --previous` om de root cause te achterhalen. Tijdsdoel: 7 minuten. Verificatiecriterium: De ontbrekende configuratieparameter is geïdentificeerd en de pod bereikt na herstel de status Running zonder crashes.

| Toetsing van snelle diagnostiek en probleemoplossing in containers.

DevOps & Automatisering

22 Uitdaging 22: CI/CD Pipeline Build Failure isoleren

Scenario: Een CI/CD-pipeline faalt tijdens de build-stage door een ontbrekende afhankelijkheid en foutieve caching. Analyseer het logbestand, identificeer de mislukte stap en herstel de caching-configuratie. Tijdsdoel: 8 minuten. Verificatiecriterium: De pipeline draait volledig succesvol door en levert een gevalideerd release-artefact op.

| Praktijkoefening voor build- en pipeline-herstel onder tijdsdruk.

Infrastructuur & Webserver

23 Uitdaging 23: Reverse Proxy en Nginx Caching configureren

Scenario: Een webserver raakt overbelast door asset-aanvragen. Configureer Nginx als reverse proxy met `proxy_cache` regels en `Cache-Control` headers om static files efficiënt af te handelen. Tijdsdoel: 10 minuten. Verificatiecriterium: `curl -I` toont `X-Cache-Status: HIT` bij herhaalde aanvragen en headers tonen de juiste TTL.

| Evaluatie van webserverconfiguratie en caching-implementatie.

Systeembeveiliging & Toegangsbeheer

24 Uitdaging 24: SSH Key-Based authenticatie en Hardening

Scenario: Een Linux-server moet worden beveiligd tegen brute-force aanvallen. Genereer een Ed25519-sleutelpaar, plaats de publieke sleutel en configureer sshd_config om wachtwoordinlog en root-login te blokkeren. Tijdsdoel: 6 minuten. Verificatiecriterium: Inloggen zonder sleutel wordt afgewezen; inloggen met Ed25519 slaagt zonder wachtwoordprompts.

| Test voor veilige servertoegang en basis-systeembeveiliging.

Applicatiebeveiliging & DevSecOps

25 Uitdaging 25: SQL Injection kwetsbaarheid detecteren en dichten

Scenario: Een zoekfunctionaliteit bouwt queries op via string-concatenatie, wat data-exfiltratie mogelijk maakt. Detecteer de kwetsbaarheid en vervang de dynamische query door geparametriseerde invoer. Tijdsdoel: 8 minuten. Verificatiecriterium: Een ' OR '1'='1 payload faalt zonder foutmelding en reguliere invoer retourneert uitsluitend gevalideerde records.

| Praktijkvaardigheid in veilige backend-ontwikkeling en kwetsbaarheidsherstel.

26 Uitdaging 27: Webhook Payload signature verificatie (HMAC)

Scenario: Een betalingsprovider verstuurt status-updates via webhooks. Implementeer signature-validatie om replay-attacks en gemanipuleerde payloads te blokkeren.

| Tijdsdoel: 7 minuten. Verificatie: Vervalste handtekeningen genereren HTTP 401 Unauthorized; authentieke payloads worden succesvol verwerkt met HTTP 200.

Datastructuren & Performance

27 Uitdaging 26: Redis In-Memory Cache implementeren voor database-ontlasting

Scenario: Een veelbezochte productpagina veroorzaakt te hoge databaselatency. Implementeer Redis caching met een TTL van 60 seconden en cache-invalidation bij updates.

| Tijdsdoel: 9 minuten. Verificatie: Responstijd daalt onder 20ms voor gecachte sleutels en cache-invalidatie functioneert correct na bewerking.

Monitoring & Observability

28 Uitdaging 28: Microservice Distributed Tracing analyseren

Scenario: Een checkout-flow vertoont onverklaarbare vertraging. Analyseer Jaeger/OpenTelemetry traces, inspecteer span-tijden en identificeer de blokkerende synchrone service-aanroep.

Tijdsdoel: 8 minuten. Verificatie: De trage downstream dependency is exact aangewezen inclusief de specifieke span-latency.

Databases & Databeheer

29 Uitdaging 29: Database Migratie zonder Downtime uitvoeren

Scenario: Een kolomnaam moet worden gewijzigd in een productietabel met miljoenen rijen zonder applicatieonderbreking. Pas het Expand-and-Contract schema toe met dual-writing.

Tijdsdoel: 10 minuten. Verificatie: Zowel oude als nieuwe codeversies kunnen gelijktijdig data lezen en schrijven zonder integriteitsfouten.

Disaster Recovery & Bedrijfscontinuïteit

30 Uitdaging 30: Disaster Recovery & Point-in-Time Database Restore

Scenario: Een onbedoelde DROP TABLE heeft plaatsgevonden om 14:02. Voer een Point-in-Time Recovery uit met base backups en Write-Ahead Logs (WAL) naar 14:01:59.

Tijdsdoel: 12 minuten. Verificatie: De data is hersteld tot 14:01:59 en de ontbrekende tabel is volledig integer beschikbaar.
