Milestone 1: Infrastruktur-Spezifikation

Hammerschmidt, Rentenberger, Schodl, Weidinger

28. November 2024

Inhaltsverzeichnis

1 Netzwerktopologie

Die Infrastruktur wird in einer **AWS VPC** implementiert und besteht aus mehreren Subnetzen:

- **Privates Subnetz**: Für interne Dienste wie LDAP, DNS und GitLab Runner.
- **Öffentliches Subnetz**: Für den GitLab-Server (mit eingeschränktem Zugriff von außen).
- **Routentabelle**: Konfiguration des Datenflusses zwischen Subnetzen und Internet-Gateway.

Graphische Darstellung: Ein Diagramm zeigt hier

- Virtuelle Maschinen (VMs) und deren IP-Adressen.
- Subnetze (privat/öffentlich) und CIDR-Blöcke.
- Routing zwischen den Subnetzen.
- DNS- und FQDN-Konventionen.

Dienst	**Subnetztyp**	**IP-Adresse**	**FQDN**
Primärer DNS	Privates Subnetz	10.0.1.10	dns1.intern.local
Sekundärer DNS	Privates Subnetz	10.0.1.11	dns2.intern.local
GitLab Server	Öffentliches Subnetz	10.0.2.10	gitlab.intern.local
GitLab Runner	Privates Subnetz	10.0.1.20	runner.intern.local
LDAP Server	Privates Subnetz	10.0.1.30	ldap.intern.local

Tabelle 1: Netzwerkdienste und IP-Zuordnung

2 Geplante Security Groups und Regeln

Die Security Groups definieren, welche Dienste und Ports erreichbar sind:

Dienst	**Eingehend (Ingress)**	**Ausgehend (Egress)**
DNS	Port 53 (UDP/TCP) von privatem Subnetz	Alle Ports ins VPC-Netz
GitLab	Port 80/443 (HTTP/HTTPS) von bestimmten IPs	Alle Ports ins VPC-Netz
GitLab Runner	Port 8093 von GitLab-Server	Alle Ports ins VPC-Netz
LDAP	Port 389 (LDAP) nur aus privatem Subnetz	Alle Ports ins VPC-Netz

Tabelle 2: Geplante Security Groups und Regeln

Inbound SGRs	DNS	Bastion	LDAP/GitLab-R	GitLab-Server
HTTP	Nein	Nein	Nein	Ja (0.0.0.0/0)
HTTPS	Nein	Nein	Ja (10.0.0.0/24)	Ja (0.0.0.0/0)
SSH	Ja (10.0.0.0/24)	Ja (0.0.0.0/0)	Ja (0.0.0.0/0)	Ja (0.0.0.0/0)
DNS (UDP)	Ja (10.0.0.0/24)	Nein	Nein	Nein
DNS (TCP)	Ja (10.0.0.0/24)	Nein	Nein	Nein
LDAP	Nein	Nein	Ja (0.0.0.0/0)	Ja (0.0.0.0/0)
ALL ICMP	Ja (0.0.0.0/0)	Nein	Nein	Ja (10.0.0.0/24)

Tabelle 3: Eingehende Sicherheitsgruppenregeln (SGRs) für verschiedene Dienste

Outbound SGRs	DNS	Bastion	LDAP/GitLab-R	GitLab-Server
HTTP	Nein	Nein	Ja (0.0.0.0/0)	Nein
HTTPS	Nein	Nein	Ja (0.0.0.0/0)	Nein
SSH	Nein	Nein	Ja (0.0.0.0/0)	Nein
DNS (UDP)	Nein	Nein	Ja (0.0.0.0/0)	Nein
DNS (TCP)	Nein	Nein	Ja (0.0.0.0/0)	Nein
LDAP	Nein	Nein	Ja (0.0.0.0/0)	Ja (0.0.0.0/0)
ALL ICMP	Nein	Nein	Ja (0.0.0.0/0)	Nein
ALL Traffic	Ja (0.0.0.0/0)	Ja (0.0.0.0/0)	Nein	Ja (0.0.0.0/0)

Tabelle 4: Ausgehende Sicherheitsgruppenregeln (SGRs) für verschiedene Dienste

Serverinstanz	Betriebssystem	Dienste
DNS-Server 1	Ubuntu 22.04 LTS	BIND (Primärer DNS)
DNS-Server 2	Ubuntu 22.04 LTS	BIND (Sekundärer DNS)
GitLab Server	Ubuntu 22.04 LTS	GitLab CE
GitLab Runner	Ubuntu 22.04 LTS	GitLab Runner
LDAP Server	Ubuntu 22.04 LTS	OpenLDAP

Tabelle 5: Dienste-Zuordnung zu Server-Instanzen

Server	OS	Packages	Version
Preferred DNS Server	Ubuntu Server	bind9, bind9utils	BIND 9.18.19 / Ubuntu 22.04.3 LTS
Alternative DNS Server	Ubuntu Server	bind9, bind9utils	BIND 9.18.19 / Ubuntu 22.04.3 LTS
LDAP Server	Ubuntu Server	slapd, ldap-utils	OpenLDAP 2.6.6
GitLab Runner	Ubuntu Server	-	Ubuntu 22.04.3 LTS
GitLab Server	Ubuntu Server	GitLab CE	GitLab CE / Ubuntu 22.04.3 LTS

Tabelle 6: Server-Spezifikationen: Betriebssystem, Pakete und Instanztypen

3 Dienste-Zuordnung zu Server-Instanzen

4 Spezifikationen der eingesetzten Systeme

- **Betriebssystem**: Ubuntu 22.04 LTS (64-bit)
- **DNS-Server**: BIND 9.18
- **GitLab**: GitLab CE 15.x

- **GitLab Runner**: Version kompatibel mit GitLab CE 15.x
- **LDAP**: OpenLDAP 2.5.x
- **Optionale Überwachung**: AWS CloudWatch zur Protokollierung und Überwachung.

5 Geplante Tests zur Funktionsprüfung

1. **DNS-Server**:
   - Überprüfung der Namensauflösung über `dig` und `nslookup`.
   - Test der Replikation zwischen primärem und sekundärem DNS.
2. **GitLab-Server**:
   - Überprüfung der Weboberfläche auf Port 80/443.
   - Push und Pull in ein Repository testen.
   - CI/CD-Pipeline ausführen.
3. **GitLab Runner**:
   - Test eines einfachen CI/CD-Jobs.
4. **LDAP-Server**:
   - Verbindung und Authentifizierung mit `ldapsearch` und `ldapwhoami`.
   - Integration mit GitLab testen.

6 Rollen und Verantwortlichkeiten im Team

Rolle	**Verantwortlichkeiten**
Netzwerk-Architekt	Planung und Einrichtung der VPC und Subnetze
DevOps-Ingenieur	Bereitstellung von GitLab und CI/CD
Systemadministrator	Konfiguration von DNS- und LDAP-Servern
QA-Ingenieur	Testen der Dienste und Sicherheitskonfiguration

Tabelle 7: Rollen und Verantwortlichkeiten

7 FQDN und interne Domain

Die interne Domain wird als **intern.local** festgelegt. Beispiele:

- DNS-Server: `dns1.intern.local`, `dns2.intern.local`.
- GitLab: `gitlab.intern.local`.

Optional: Monitoring

AWS CloudWatch wird zur Protokollierung und Überwachung genutzt:

- Überwachung der CPU-, Speicher- und Netzwerknutzung.
- Automatische Alarme bei Ausfällen.