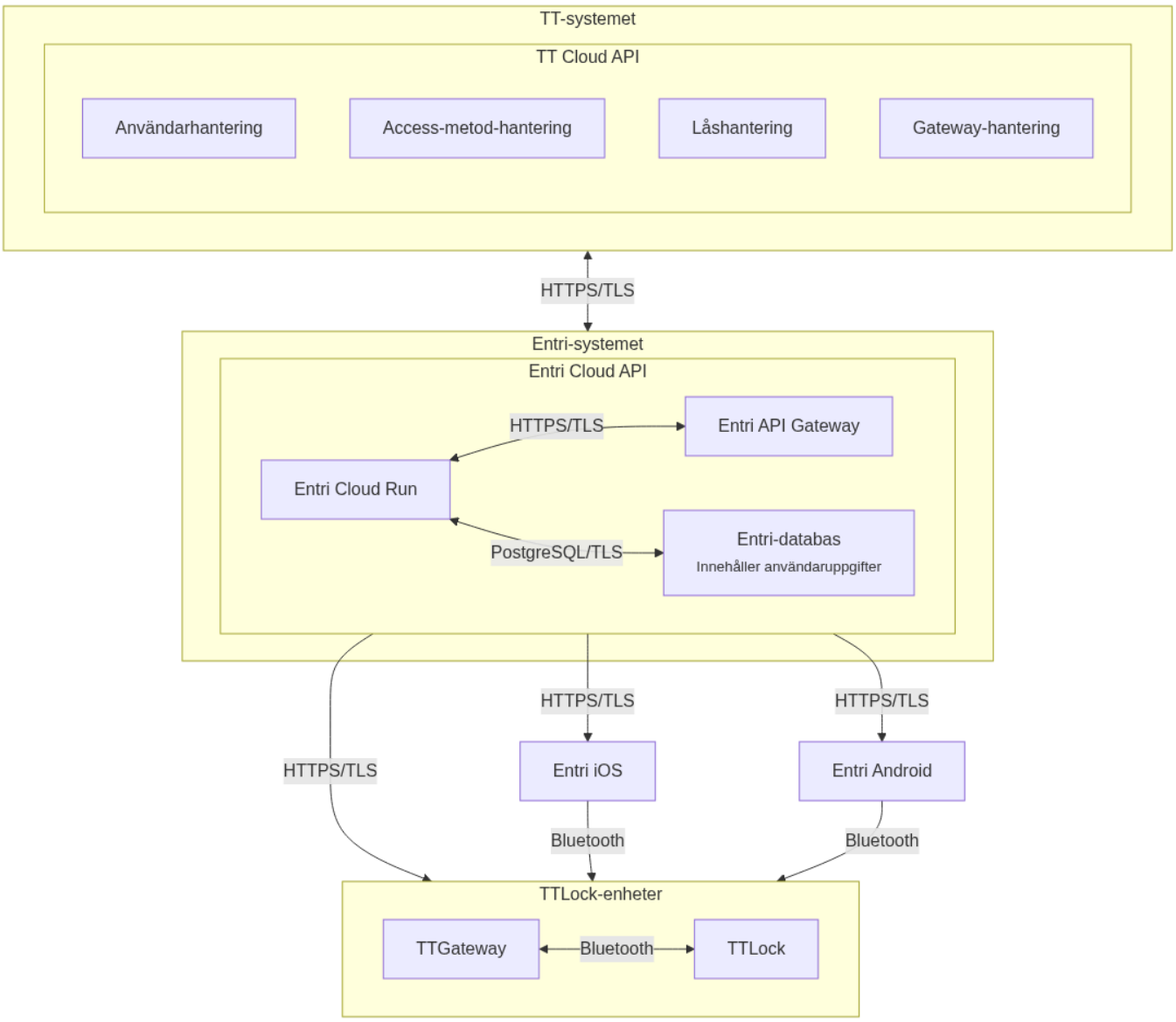
Säkerhetsrapport

I denna rapport går vi igenom olika säkerhetsaspekter gällande Entri-systemet (Beslagsgruppen i Kållered ABs varumärke för deras nya app och underliggande system) och dess användning av tredje-part TTLock.

Systemöversikt



Driftsmiljö

Vi använder följande Google Cloud-tjänster:

- Google Cloud Run (Stockholm)
- Google Cloud SQL (Belgien)
- Google Cloud API Gateway (Belgien)
- Google Artifact Registry

Dataflöde

All kommunikation med både Entris och TTLocks molntjänster sker över HTTPS med TLS, vilket innebär att all trafik är krypterad under överföring (in transit). Övrig kommunikation sker direkt mellan klienten och hårdvaran (lås och gateway) via Bluetooth-protokollet, med hjälp av de officiella SDK:erna för iOS och Android.

All kommunikation till TTLock går genom Entri-backend, som fungerar som en proxy och vidarebefordrar anrop till TTLocks molntjänst. Backend definierar endast de Entri-specifika operationer som krävs enligt Entris behov, exempelvis funktioner som inte finns hos TTLock eller som bedöms vara osäkra att exponera direkt. Övriga anrop proxas vidare oförändrat.

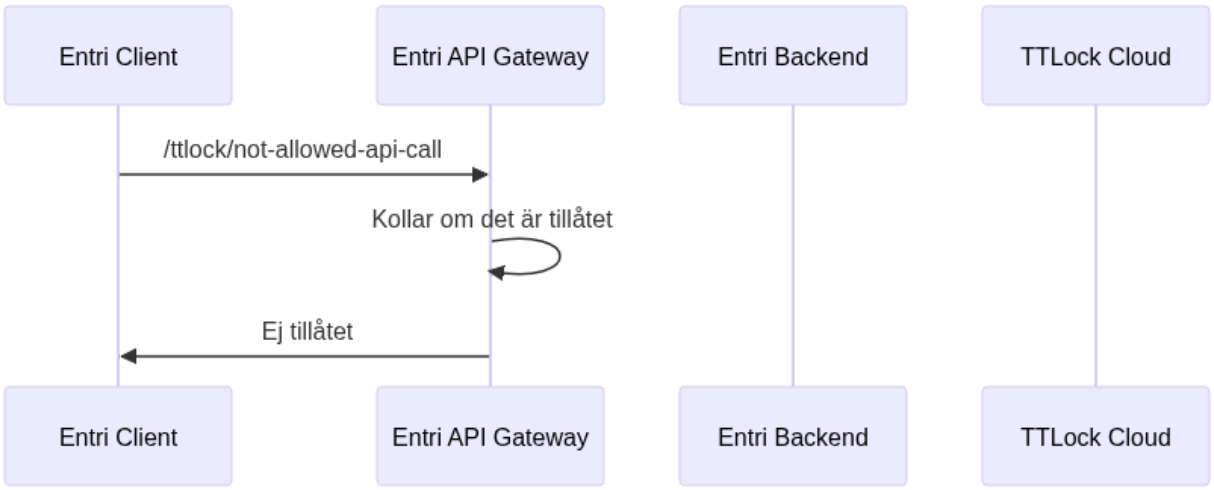
Entris-specifika operationer är följande:

- Registrering av Entri-användare
- Inloggning av Entri-användare
- Överföra lås mellan en Administratör till en annan användare

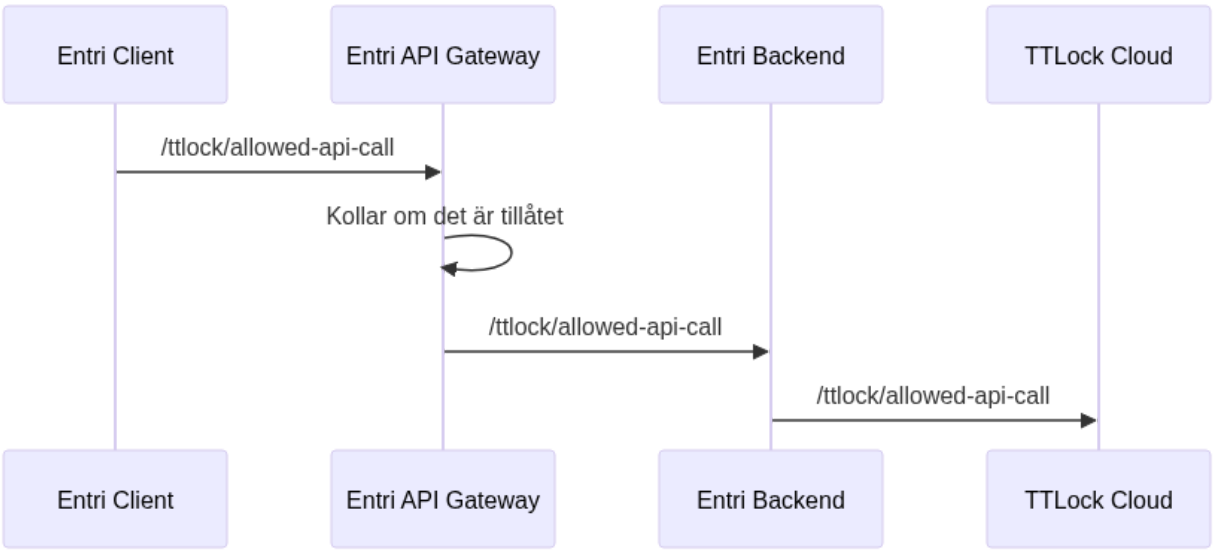
För att upprätthålla kontroll används Entris API Gateway, vars syfte är att vitlista tillåtna anrop till TTLock. Detta innebär att all kommunikation alltid passerar genom Gateway:en, som endast vidarebefordrar anrop som är uttryckligen tillåtna. På detta sätt kan Entri säkerställa vilka delar av TTLocks API:er som är tillgängliga att använda.

Det är inte möjligt att anropa Entris backend direkt. Detta förhindras genom att backend endast kan kommunicera med ett specifikt Google Cloud Service Account, vilket innebär att all trafik måste gå via Entris API Gateway. Se exemplena nedan:

Otillåtna anrop



Tillåtna anrop



Övriga beroenden

Backend

- `github.com/jackc/pgx/v5 v5.7.5` (PostgreSQL-driver)
- `github.com/golang-migrate/migrate/v4 v4.18.3` (Databasmigreringar)
- **TTLock API**

iOS

- **TTLock SDK** (Bluetooth-kommunikation mot TTLocks enheter)

Android

- `androidx-compose-bom: 2025.03.00`
- `room: 2.6.1`
- `hilt: 2.52`
- `retrofit: 3.0.0` (HTTP-klient)
- `okhttp: 4.12.0` (HTTP-klient)
- `kotlinx-coroutines: 1.7.3`
- `androidx-navigation-compose: 2.8.9`
- `androidx-datastore-preferences: 1.1.7`
- `tink-android: 1.8.0` (Kryptografi)
- `ttlock: 3.5.4` (Bluetooth-kommunikation mot TTLocks enheter)
- `kotlinx-serialization-json: 1.8.0`

Användare och roller

För varje slutanvändare finns två separata konton – ett Entri-konto och ett TTLock-konto. Kontona kopplas samman genom en relationen i Entris databas där varje Entri-användare får ett associerat TTLock-användarkonto.

Användaridentitet och integritetsskydd:

- Vid registrering genereras ett slumpmässigt 16-tecken användarnamn (alfanumeriskt) via kryptografiskt säker random-generator
- Detta slumpmässiga användarnamn skickas till TTLock som användaridentitet
- TTLock får aldrig tillgång till användarens riktiga identitetsuppgifter (e-post, namn)
- Entri lagrar kopplingen mellan användarens riktiga uppgifter (krypterat i databasen) och det slumpmässiga TTLock-användarnamnet
- All användarautentisering delegeras till TTLocks OAuth2-system - Entri gör ingen egen tokenvalidering

I Entris databas definieras inga roller, och vi delegerar all autentisering och auktorisering av tokens till TTLocks interna hantering. Detta säkerställer fullständig anonymisering av Entri-användarnas identiteter gentemot TTLock.

TTLock har tre roller:

- Administratör
    - Första användaren som kopplar upp sig till ett lås får denna roll och har full kontroll över låsets inställningar. Exempelvis är det bara denna användare som kan fabriksåterställa det.

    Denna användare kan lägga till accessmetoder (taggar, fingeravtryck och pinkoder).
- Inbjuden Administratör
    - Denna användare bjuds in av Administratören och även denna roll har möjlighet att lägga till accessmetoder.
- Vanlig Användare
    - Dessa användare kan bara låsa upp låset med appen. De kan också låsa upp låset om de har blivit tilldelade en accessmetod av en administratör, exempelvis om de äger en tagg, pinkod eller har skannat sitt fingeravtryck på låset.

Autentisering och Åtkomstkontroll

Eftersom Entri-systemet inte själv hanterar autentisering, utöver lagring av krypterade användaruppgifter, förlitar sig systemet på TTLocks OAuth2-flöde. Vid registrering i Entri anger användaren e-postadress och lösenord, vilka lagras krypterat.

I samband med registreringen genereras dessutom ett slumpmässigt användarnamn, helt oberoende av de uppgifter som användaren angett. Detta användarnamn används för att registrera kontot hos TTLock. TTLock får därmed ingen information om vem användaren är, eftersom övriga personuppgifter endast lagras, i krypterad form, hos Entri.

Under registrering så måste förövrigt lösenorden uppfylla följande krav:

- Minst 8 tecken
- Minst 3 av följande:
    - Gemena bokstäver
    - Versala bokstäver
    - Siffror
    - Specialtecken

Genom OAuth2-flödet tilldelar TTLock ett Access Token samt ett Refresh Token, där det senare används för att begära ut nya access tokens vid behov.

Sessionshantering iOS

På iOS hanteras Access Token, Refresh Token, Provider User ID och lösenord via KeychainService API med följande säkerhetsegenskaper:

- Data krypteras automatiskt av iOS med hårdvarubaserad kryptering
- Tokens är endast tillgängliga när enheten är upplåst
- Data synkas inte mellan enheter

Inför varje nätverksanrop hämtas Access Token för att autentisera anropet, medan Refresh Token används för att begära ett nytt Access Token när det gamla har gått ut.

Sessionshantering Android

På Android används för närvarande Preferences Data Store för att lagra både Access Token och Refresh Token. Krypteringen sker med hjälp av Androids inbyggda KeyStore tillsammans med det open-source-biblioteket Tink. Tokens krypteras med AES-256-GCM.

Inför varje nätverksanrop hämtas Access Token för att autentisera anropet, medan Refresh Token används för att begära ett nytt Access Token när det gamla har gått ut.

Autentisering av Backend mot TTLock

TTLock kräver att varje klient är registrerad och kan identifiera sig med Client ID och Client Secret. Client ID måste alltid skickas med i anropen till TTLock, vilket hanteras centralt av Entri-backend. Detta innebär att klienterna (iOS och Android) aldrig har direkt tillgång till Client ID, utan det hanteras enbart från backend.

Enligt TTLock finns endast två klienter: Entri Dev (backend för utvecklingsmiljö) och Entri Prod (backend för produktionsmiljö).

Client Secret hanteras också exklusivt av backend och skickas aldrig från klienterna. Både Client ID och Client Secret injiceras i backend under byggsteget och lagras som säkra miljövariabler på GitHub.

Datasäkerhet

Kryptering av data i vila

Entris backend-system lagrar användaruppgifter (e-post, förnamn, efternamn och lösenord) i en PostgreSQL 16-databas med AES-256-CTR-kryptering.

Databasinstansen kräver SSL-kryptering och är konfigurerad med `ENCRYPTED_ONLY` -läge.

Denna robusta krypteringsimplementering med AES-256-CTR uppfyller EN 18031:s strikta krav på skydd av personuppgifter i vila enligt Article 3.3(e).

Kryptering av data i transit

All kommunikation mellan klientapplikationer och backend-tjänsterna sker över TLS-krypterade anslutningar. API Gateway är konfigurerat att endast acceptera HTTPS-förfrågningar, och Cloud Run-tjänsterna kommunicerar internt via HTTP/2 med JWT-autentisering för ytterligare säkerhet.

Databasanslutningar kräver SSL (`POSTGRES_SSL_MODE=require` i produktionsmiljön). Värt att tillägga att vi aldrig har produktionsdata i utvecklingsmiljön, därav finns ingen risk att detta hämtas från utvecklingsmiljön.

Nyckelhantering

Systemet hanterar känsliga konfigurationsparametrar genom GitHub Secrets som injiceras som miljövariabler under deployment. Produktionsmiljön använder separata hemliga nycklar från utvecklingsmiljön.

Infrastruktursäkerhet

Alla våra deployments (CI/CD, kodsignering av appar) körs på byggdator in-house innan de skjuts iväg till Google Cloud och respektive app-butiker (App Store och Google Play Store).

För konfigurering av Infrastruktur så använder vi Terraform. Alla känsliga variabler är markerade som `sensitive = true` i Terraform-konfigurationen och exponeras aldrig i loggar eller versionshantering.

Cloud Run-säkerhet

Autentisering mellan API Gateway och Cloud Run sker med hjälp av ett Service Account. Som tidigare nämnts är det inte möjligt att autentisera sig direkt mot backend utan det korrekta service account, vilket endast API Gateway har tillgång till.

Database-säkerhet

Databasen är endast åtkomlig via privat IP med begränsad nätverksåtkomst, och alla anslutningar sker över SSL/TLS. Regelbundna säkerhetskopior med point-in-time recovery säkerställer att data kan återställas vid behov.

API Gateway-säkerhet

API Gateway skyddar systemet genom vitlistning av tillåtna endpoints, vilket gör det möjligt att blockera operationer hos TTLock som bedöms som osäkra. Ett exempel är

lösenordsåterställningsfunktionen, som endast krävde ett token och användarnamn; detta var en anledning till att användarnamnet obfuskeras med kryptering.

Compliance med EU:s Radio Equipment Directive (RED)

Från 1 augusti 2025 träder EU:s cybersäkerhetsförordning EN 18031 i kraft som en del av Radio Equipment Directive. Förordningen påverkar TTLock:s gateway-hårdvara (G2 WiFi & G3 Ethernet) som internetansluten radioutrustning som hanterar personuppgifter.

Entri-applikationerna och TTLock-enheterna omfattas inte av förordningen då de endast använder Bluetooth-kommunikation utan direkt internetanslutning. Våra mjukvarukomponenter uppfyller redan höga säkerhetsstandarder med HTTPS/TLS-kommunikation, OAuth2-autentisering och krypterad datalagring.