INSTALLATION - KONFIGURATION - DRIFT

# WitVia

Svensk användar- och installationsmanual

Grafisk revision: 5 november 2026

Tekniskt innehåll: 21 augusti 2026

För installatörer, systemintegratörer, systemadministratörer och driftansvariga

# Om manualen

**Manualen beskriver nuvarande WitVia Edge-plattform från fysisk installation till daglig drift, inklusive valbar WitVia-TBR på samma Raspberry Pi.**

> **Versionsprincip:** Menyer och funktioner kan bero på installerad Core-version, aktiverade komponenter och vald leveransomfattning. Kontrollera alltid Core-version och health i System Admin före arbete.

WitVia är en lokal edgeplattform. Normal protokolltrafik, styrning, larm och livevärden hanteras på Edge. Internetanslutning används för centrala kontoflöden, signerade uppdateringar och funktioner som uttryckligen kräver en extern tjänst.

## Så används manualen

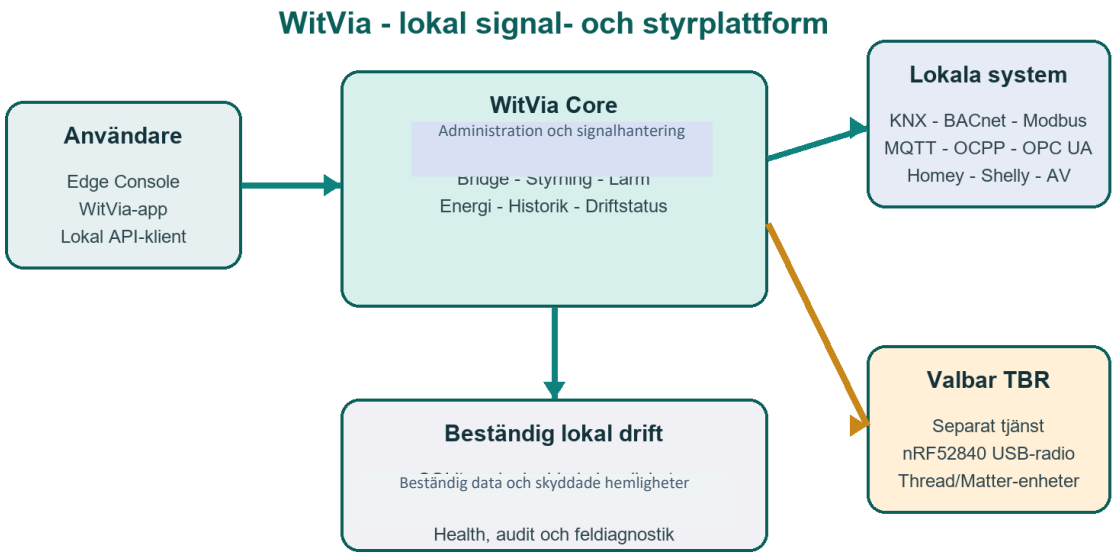
- Börja med Installation och första start när en ny Edge ska driftsättas.
- Använd Source Admin-avsnitten för respektive protokoll.
- Läs Säkerhet och regler före inkoppling mot fastighetsautomation, laddning eller annan OT-utrustning.
- Använd Felsökning och driftskontroll när en källa eller funktion inte blir grön.

# Innehåll

**Om manualen** ..... 2
**Innehåll** ..... 3
**Systemöversikt** ..... 4
**Säkerhet, regler och ansvar** ..... 5
**Leverans och förberedelser** ..... 6
**Installation och första start** ..... 7
**Registrering, konto och MFA** ..... 8
**Edge Console** ..... 9
**Source Admin - gemensamt arbetssätt** ..... 10
**KNX** ..... 11
**Modbus** ..... 12
**BACnet** ..... 13
**MQTT** ..... 14
**OCPP och OPC UA** ..... 15
**Lokala enheter och AV** ..... 16
**Matter och WitVia-TBR** ..... 17
**Bridge** ..... 18
**Styrning** ..... 19
**Larm, larmgrupper och jour** ..... 20
**Energi** ..... 21
**System Admin** ..... 22
**Larmmail** ..... 23
**Uppdatering, backup och återställning** ..... 24
**Felsökning** ..... 25
**Teknisk referens** ..... 26
**Driftsättningskontroll** ..... 27
Begrepp 28
Skärmbilder från Edge Console 29

# Systemöversikt

**WitVia samlar lokala protokoll i en gemensam signalmodell. Samma signaler kan sedan visas, larmas, bryggas och användas i Styrning.**




| Komponent | Ansvar |
|---|---|
| **WitVia Core** | Källor, mappings, signalhantering, Bridge, Styrning, larm, energi, API och Edge Console. |
| **Source adapter** | En isolerad anslutning till ett protokoll eller en lokal enhet. Varje anslutning har egen health och supervisor. |
| **WitVia-TBR** | Valbar separat tjänst för Thread-routing, Border Agent, SRP/Discovery och USB-radio. Kan köras på samma Pi som Core. |
| **WitVia-app** | Länkas till Edge och kan användas för Matter-commissioning med telefonens BLE när det behövs. |
| **Signerad updater** | Installerar verifierade Core-paket atomiskt och återställer föregående version om readiness misslyckas. |

> **Viktigt:** Thread är ett av flera protokoll i WitVia. WitVia-TBR installeras endast när anläggningen behöver en Thread Border Router.

# Säkerhet, regler och ansvar

**WitVia ska projekteras som en del av anläggningens IT- och OT-miljö. Den tekniska möjligheten att ansluta en enhet innebär inte automatiskt att installationen är tillåten, certifierad eller lämplig.**

- Fast elinstallation, laddinfrastruktur, brandfunktion, ventilation och andra reglerade system ska hanteras av behörig person enligt lokala krav.
- Följ respektive enhets leverantörsvillkor, installationsanvisningar och säkerhetsgränser.
- Placera okrypterade lokala protokoll i betrodda segment och begränsa trafik med brandvägg eller VLAN.
- Återanvänd inte lösenord. Lagra inte nycklar, setup-koder eller certifikat i projektfiler, mail eller supportärenden.
- Testa alltid skrivning, återläsning, felbeteende och återställning innan en Bridge eller Styrning aktiveras skarpt.
- Certifiering får endast påstås när den kan styrkas uttryckligen. BACnet-komponenten har en verifierad implementation PICS men beskrivs inte som BTL-certifierad.

## Nätsegmentering

| Trafik | Rekommendation |
|---|---|
| **Edge Console och API** | HTTPS på 443. Begränsa administration till betrodda klienter. |
| **KNX, BACnet och Modbus** | Separat OT-segment när anläggningen kräver det. Tillåt endast nödvändiga portar och motparter. |
| **MQTT, Shelly, Sonos och PJLink** | Betrott IoT/AV-segment. Använd TLS där protokollet stöder det. |
| **Thread/Matter** | IPv6 och multicast måste fungera mellan Edge/TBR och relevant infrastrukturlänk. |

# Leverans och förberedelser

## Kontrollera leveransen

- Raspberry Pi eller annan avtalad Linux ARM64 Edge-plattform med stabil strömförsörjning.
- Ethernetanslutning. Kabel rekommenderas för fast installation.
- Valbar nRF52840-baserad USB-radio när WitVia-TBR ingår.
- Adressplan, DNS, tidskälla, VLAN och brandväggsregler.
- Projektunderlag: ETS-export, Modbus-registerlista, BACnet Device instance, certifikat, MQTT-topics, OCPP-identiteter eller motsvarande.

## Förbered nätverket

- Reservera DHCP-adress eller planera statisk IPv4.
- Säkerställ DNS och korrekt tidszon. Fel tid påverkar certifikat, larm, loggar och scheman.
- Tillåt utgående HTTPS till WitVias centrala tjänster när registrering och uppdatering ska användas.
- Planera multicast för mDNS, Matter, Sonos och annan discovery. Routad drift kan kräva explicita unicast-adresser.
- Dokumentera vilka skrivbara signaler som ska tillåtas innan import eller discovery.

> **Kapacitet:** Core och WitVia-TBR kan köras på samma Pi när CPU, minne, USB, lagring och nätverk är dimensionerade för båda. Tjänsterna använder separata portar, statusfiler, credentials och livscykler.

# Installation och första start

1. **Montera Edge säkert.** Sörj för ventilation, dragavlastning och åtkomst till Ethernet och eventuell USB-radio.
2. **Anslut Ethernet.** Använd det projekterade nätsegmentet innan strömmen ansluts.
3. **Anslut valbar TBR-radio.** USB-radion används endast när Thread Border Router ingår i leveransen.
4. **Slå på strömmen.** Vänta tills operativsystem, proxy och Core har startat.
5. **Öppna Edge.** Gå till https://<värdnamn>.local/ eller enhetens IP-adress.

> **Lokalt certifikat:** Den lokala HTTPS-proxyn använder ett självsignerat certifikat som förnyas när värdnamn eller IP ändras. Administratören ska verifiera att adressen verkligen tillhör rätt Edge innan ett webbläsarundantag godkänns.

## Grundkontroll

- Sidopanelen visar Ansluten till Core och en versionssträng.
- Livestatus laddas utan feltoast.
- System Admin visar rätt värdnamn, operativsystem, arkitektur och nätverksinterface.
- Tidszon är korrekt, normalt Europe/Stockholm för svensk anläggning.

```
https://<vardnamn>.local/
https://<edge-ip>/
```

# Registrering, konto och MFA

**Registreringen skapar Edge-enhetens identitet, ett lokalt administratörskonto och kopplingen till användarens centrala konto.**

1. **Öppna Licens i Edge Console.** Välj Registrera Edge.
2. **Ange uppgifterna.** Fyll i namn, e-post, enhetsnamn och ett starkt lösenord.
3. **Skicka registreringen.** Edge skapar enhets-ID och lokalt konto. Enhetens hemlighet visas inte.
4. **Slutför MFA.** Följ länken till portalen och slutför TOTP-registrering eller inloggning.
5. **Kontrollera status.** Återgå till Edge och välj Kontrollera status tills registrering, konto och uppdateringsstatus är gröna.

> **Separata roller:** Det lokala kontot används för ändringar på just denna Edge. Det centrala kontot och MFA används för ägande och portalflöden. Dela inte lokala administratörsuppgifter mellan personer.

## Återställning

- Förvara återställningskoder på en skyddad plats som inte är samma enhet som autentiseringsappen.
- Vid ägarbyte ska användare, grupper, jour och enhetskoppling granskas som en sammanhållen process.
- Support ska aldrig efterfråga enhetshemlighet, TOTP-hemlighet, nätverksnyckel eller setup-kod i klartext.

# Edge Console

| Vy | Användning |
|---|---|
| **Livestatus** | Aktuella signaler, värde, kvalitet, aktualitet, källa och ursprung. |
| **Energi** | Elpris, effekt, kostnad och historik när energi är aktiverat. |
| **Source Admin** | Skapa anslutningar, mappings, discovery, import, läsning och skrivning. |
| **Bridge** | Koppla två mappings eller, i stödjande version, en KNX-mapping till en state-safe variabel. |
| **Larm** | Aktiva larm, kvittens, händelser, regler, grupper och jour. |
| **Min jour** | Personlig kalender och larm som routas till användarens aktuella jour. |
| **Styrning** | Bygg, validera, simulera och aktivera grafiska styrflöden. |
| **Licens** | Registreringsstatus, enhetsidentitet, konto och uppdateringsstatus. |
| **System Admin** | Nätverk, TLS-policy, uppdatering, app-länk, larmmail, användare och diagnostik. |

## Statusbegrepp

- Online betyder att adapterns egen health är grön - inte att varje mapping är korrekt projekterad.
- Aktuell betyder att värdet ligger inom sin aktualitetspolicy eller att protokollets verifierade anslutningsliv håller det aktuellt.
- Inaktuell betyder att värdet inte ska användas för nya styrbeslut utan uttrycklig fallback.
- Dålig kvalitet ska hanteras som ett felvärde, inte som ett normalt numeriskt värde.

# Source Admin - gemensamt arbetssätt

1. **Välj source-typ.** Skapa en ny anslutning och lämna den inaktiv tills parametrarna är kontrollerade.
2. **Ange anslutningsdata.** Host, port, transport, tidsgränser och protokollspecifika identiteter.
3. **Provisionera hemligheter separat.** Använd Source Admins särskilda åtgärd för lösenord, nycklar eller certifikat.
4. **Upptäck eller importera mappings.** Granska alltid namn, datatyp, enhet och skrivbarhet före aktivering.
5. **Aktivera och prova.** Kontrollera health, läsning och endast avsedd skrivning med återläsning.
6. **Dokumentera.** Spara adressplan, ägare, beroenden, brandväggsregler och acceptansresultat i projektets förvaltningsunderlag.

## Mappings

En mapping kopplar en protokollidentitet till WitVias gemensamma fält: namn, beskrivning, datatyp, enhet och skrivbarhet. Protokollspecifika uppgifter ligger i mappingens inställningar.

- Sätt endast writable när både protokollet och anläggningens ansvarsmatris tillåter skrivning.
- Använd deadband för numeriska källor när endast relevanta värdeförändringar ska behandlas vidare.

- Template-apply mergar normalt på signalnyckel och raderar inte andra mappings.
- Testa explicit Läs och Skriv via Source Admin innan Bridge eller Styrning använder mappingen.

> **Ändringsdisciplin:** Ändra en anslutning i taget. Efter sparning ska just den anslutningens health och livevärden verifieras innan nästa ändring görs.

# KNX

KNX-adaptern håller en Falcon-baserad KNX/IP-anslutning och hanterar grupptelegram enligt konfigurerade mappings och DPT-typer.

1. **Skapa KNX-källa.** Ange KNX/IP-interface, normalt port 3671, och välj gruppadressformat.
2. **Spara anslutningen.** Skapa först en beständig source-identitet.
3. **Importera ETS-projekt.** XML, KNXPROJ eller ZIP skapar en template som appliceras explicit.
4. **Importera keyring separat vid Secure.** En KNX Secure-keyring är inte samma sak som ETS-projektfilen.
5. **Kontrollera DPT och skrivbarhet.** Gruppadress är signalKey; DPT ligger som mappinginställning.
6. **Lyssna och prova.** Verifiera telegram, explicit gruppread och endast avsedda gruppwrite.

> **Okända telegram:** Telegram utan mapping kan visas rått men startar inte automatiskt polling eller styrning. Lägg till och typa mappingen innan signalen används operativt.

## Definierat bussvärde

I versioner som visar variabelmålet i Bridge kan en KNX-mapping kopplas till ett definierat svarsvärde. Funktionen säkerställer en entydig källa per KNX-mapping för att undvika konkurrerande svar.

- Använd funktionen för bussobjekt som behöver ett definierat svarsvärde.
- Kontrollera DPT, datatyp och riktning innan bidirektionell användning.
- Inaktivera variabeln eller Bridge-regeln före större förändring och verifiera gruppread efter aktivering.

# Modbus

| Transport | Användning |
|---|---|
| **TCP** | Klient mot Modbus TCP-enhet, normalt port 502. |
| **TCP Security** | MBAP över ömsesidig TLS, normalt port 802. |
| **RTU** | Serieport med CRC och t3.5-framing. Kräver rätt Linux-serieenhet och parametrar. |
| **ASCII** | Serieformat med LRC och CRLF. |
| **TCP Server** | Exponerar endast uttryckligt konfigurerade mappings till externa klienter. |

1. **Välj transport och endpoint.** Ange unit ID, adressbas, tidsgränser och pollintervall.
2. **Importera profil eller skapa mapping.** Signalnycklar normaliseras som coil, discrete, holding eller input.
3. **Ange rå datatyp.** Kontrollera byteordning, word order, scale, offset och antal register.
4. **Optimera varsamt.** Max registergap 0 undviker läsning av okända mellanregister.
5. **Verifiera write med read-back.** Kontrollera funktion, separat statusadress och enhetens egna skyddsgränser.

> **Klartext:** Vanlig Modbus TCP har varken autentisering eller kryptering. Använd ett betrott OT-nät eller TCP Security när hotbilden kräver det.

# BACnet

**WitVia kan exponera en lokal BACnet-enhet eller ansluta externa BACnet/IP- och BACnet/SC-enheter. Produktens BACnet-server använder Witcons registrerade BACnet Vendor ID; numret behöver normalt inte anges av installatören.**

1. **Välj roll.** Lokal enhet exponerar WitVia-punkter; extern enhet läser och skriver en fjärrenhet.
2. **Sätt Device instance.** Varje lokal eller extern identitet måste vara unik och korrekt projekterad.
3. **Välj transport.** BACnet/IP använder UDP, normalt 47808. BACnet/SC kräver certifikat och hubbadress.
4. **Upptäck enhet och objekt.** Granska objekt, property, array-index, skrivbarhet och eventuell prioritet.
5. **Aktivera avancerade roller explicit.** Routing, BBMD och Foreign Device ska bara aktiveras när nätets design kräver det.
6. **Verifiera interop.** Testa ReadProperty, WriteProperty, RPM/RP-fallback och COV mot den verkliga motparten.

## Utökat stöd

- Direkta och routade mål, flerportsrouting samt BBMD/Foreign Device som valbara roller.
- ReadPropertyMultiple med ReadProperty-fallback och hanterad COV/COV Multiple.
- Who-Is/I-Am, Who-Has/I-Have och katalog/discovery.
- Valbar tidsynk, larm/events, ReadRange och loggobjekt.
- Calendar och Schedule med versions- och konfigurationsberoende användning.

> **Konformitet:** Implementation PICS är verifierad för den fristående komponenten. Det är inte samma sak som BTL-certifiering eller garanti för alla roller i varje installation. Aktivera bara den profil som har projekterats och verifierats på plats.

# MQTT

## Hosted broker

- Körs inuti Core och tillåter aldrig anonym anslutning.
- Varje klient får eget användarnamn, engångsvisat lösenord och separata publish/subscribe-ACL.
- En anslutningsägd allMappings-destination kan publicera valda state- och eventtopics.
- Hosted broker övervakas och återställs automatiskt vid driftfel.

## Extern broker

- Ange brokeradress, port, klientidentitet, TLS och credentials.
- Mappings använder topicfilter, datatyp och valfri JSON Pointer.
- Skrivbara mappings använder separat command topic och kan bygga JSON-payload.
- Sparkplug B väljs per mapping och använder samma anslutning, inte en extra broker.

> Topic-ACL: Ge en klient endast de topics och riktningar som behövs. Ge inte bred publish-rätt till reserverade topics.

# OCPP och OPC UA

## OCPP

- Stöd för OCPP-J 1.6, 2.0.1 och 2.1 via en gemensam CSMS-source.
- Laddaren ansluter normalt med WSS till wss://<edge-namn>/ocpp/<charge-point-id>.
- Basic Auth provisioneras per laddpunkt och lagras hashat; lösenordet ska inte ligga i URL eller dokumentation.
- Välj versionsspecifika actions och response templates endast när anläggningens affärslogik är definierad.
- Testa BootNotification, Heartbeat, StatusNotification och avsedda skrivkommandon före överlämning.

## OPC UA

- Klientläge kan använda discovery, browse, batch-read, typad write och subscriptions.
- Serverläge exponerar endast konfigurerade variabler och roller.
- Produktionsdrift ska använda betrodd PKI och krypterad endpoint, exempelvis Basic256Sha256 eller modern AES-baserad policy.
- SecurityPolicy None ska endast användas i ett uttryckligt loopback-test.

> **Certifikat:** Kontrollera DNS-namn, giltighet, kedja och tillit på båda sidor. Automatisk tillit till ett okänt produktionscertifikat är inte en acceptabel driftsättningsmetod.

# Lokala enheter och AV

| Adapter | Första kontroll |
|---|---|
| **Homey Energy** | Aktivera Local API i Homey-appen och kontrollera dongelns lokala WebSocket. |
| **Shelly** | Gen1-4 stöds lokalt. Kör discovery, sätt auth separat och verifiera read-back. |
| **SiGN** | QR-onboarding hämtar nödvändig device sharing-data; löpande drift sker lokalt över LAN. |
| **Sonos** | Installera i Sonos-appen, aktivera UPnP och tillåt SSDP/TCP 1400 på lokalt AV-nät. |
| **PJLink** | Upptäck projektorn eller ange host, normalt TCP 4352. Provisionera lösenord separat. |

## Homey Energy

Välj host, port och WebSocket-path. Standardmappings omfattar bland annat total effekt, fasvärden, spänning, ström och mätarställning. Telegrammens CRC valideras innan värden publiceras.

## Shelly

Discovery identifierar generation och föreslår booleska, numeriska och textbaserade statusfält. Endast säkerhetsvaliderade relä-, ljus-, färg- och coverkommandon ska göras skrivbara.

## SiGN

SiGN/Tuya kan onboardas via Smart Life device sharing utan eget Tuya Developer Cloud-projekt. Local key lagras separat och får aldrig visas i logg eller supportunderlag.

## Sonos och PJLink

Sonos använder lokalt UPnP/SOAP utan TLS; PJLink-trafik är också okrypterad även när kommandon autentiseras. Båda ska därför ligga i ett betrott AV-/IoT-segment. Använd discovery eller routad skanning och reservera stabila IP-adresser.

# Matter och WitVia-TBR

**WitVia är Matter-controller för enheter från Matter 1.0 till 1.6. Commissioning kan använda lokal BLE, telefonens distribuerade BLE-länk eller en redan nåbar enhet på IPv6-nätet.**

1. **Förbered nätet.** Säkerställ IPv6, mDNS och korrekt tid. För Thread behövs en kompatibel Border Router.
2. **Öppna Source Admin - Matter.** Välj Lägg till Matter-enhet och commissioning-transport.
3. **Ange setup-koden.** Använd QR eller manuell Matter-kod. Koden ska behandlas som en hemlighet under commissioning.
4. **Välj nätcredential när det krävs.** För första Matter-over-Thread-enheten kan Android ge ett användarauktoriserat Thread-dataset.
5. **Vänta på komplett commissioning.** Avbryt inte när BLE, PASE, nätprovisionering, CASE och capability-läsning pågår.
6. **Verifiera mappings.** Kontrollera enhetens faktiska endpoints, kluster, attribut och kommandon - inte bara modellnamnet.

## WitVia-TBR på samma Pi

- Core och TBR körs som separata systemd-tjänster och kan dela Raspberry Pi när kapaciteten räcker.
- TBR använder en nRF52840-baserad USB-radio för tidskritisk IEEE 802.15.4-radiohantering.
- Pi:n äger routing, Border Agent, SRP/Advertising/Discovery Proxy och Linux-nätintegration.
- Thread-credentials ska komma från ett officiellt auktoriserat onboardingflöde eller från explicit provisionering av ett nytt nät.
- TBR-readiness är mer än att tjänsten kör: radio, attach, child/parent, Matter UDP 5540 och discoveryvägar ska verifieras.

> **IKEA:** Matter-over-Thread-produkter från IKEA kan onboardas via Matter och nås genom en kompatibel TBR, exempelvis WitVia-TBR. Äldre IKEA-produkter som endast använder Zigbee blir inte automatiskt Matter/Thread-enheter och kräver en separat kompatibel brygga. Stöd ska alltid kontrolleras per modell och firmware.

## Länka WitVia-appen

1. **Öppna System Admin.** Välj Länka en telefon och skapa en 8-siffrig engångskod.
2. **Öppna appen.** Välj Lägg till Edge och ange koden innan den löper ut.

3. **Kontrollera länken.** Telefonen får en permanent länknyckel och kan därefter starta Matter-flöden enligt sin behörighet.

## Bridge

**Bridge vidarebefordrar färska signaler med god kvalitet mellan två mappings. Varje regel har riktning, enable-status, valfri boolesk invertering, räknare och begränsad audit.**

1. **Välj Källa A.** Utgå från den signal som ska starta vidarebefordringen.
2. **Välj mål.** Normalt Källa B. I stödjande version kan en KNX-mapping använda en state-safe variabel.
3. **Välj riktning.** A till B, B till A eller dubbelriktat. Mottagande sida måste vara skrivbar.
4. **Välj eventuell invertering.** Använd endast för booleska eller uttryckligen kompatibla värden.
5. **Spara inaktivt och prova.** Kontrollera write, read-back, loopskydd och audit innan aktivering.

• Inbyggt loopskydd hindrar att ett fysiskt eko skapar en oändlig kedja.

- En Bridge ersätter inte anläggningens säkerhetslogik eller gränsvärden.
- Riktningsbyte eller byte av skrivbart mål ska behandlas som en funktionsändring med ny acceptanstest.

# Styrning

**Styrning är WitVias grafiska, lokala styrmotor. Katalogen innehåller 83 körbara block med typade portar, validering, simulering och livekörning i Core.**

## Arbetsflöde

1. **Skapa eller klona en styrning.** Ge den ett tydligt namn, syfte och ansvarig.
2. **Bygg grafen.** Välj trigger, läsning, villkor, tidsblock, säkerhetsblock och kommando.
3. **Validera.** Åtgärda fel för portar, datatyper, cykler, onåbara noder och saknade mappings.
4. **Simulera.** Simulering skriver aldrig till verkliga källor. Granska spårning och grenval.
5. **Spara och aktivera.** Livekörning kräver att styrningen är sparad, aktiv och har status Redo.
6. **Följ körhistorik.** Verifiera trigger, beslut, skrivning, återkopplingsskydd och eventuella fel.

## Viktiga blockgrupper

- Trigger: manuell, källhändelse, schema, intervall och signalövervakning.
- Logik och matematik: villkor, NOT, boolesk logik, väljare, min/max, prioriterat val och tillståndsmaskin.
- Tid och OT: delay, debounce, separata till-/frånslagstider, cooldown, min/max gångtid, driftlägesfördröjning och pulståg.
- Flöde: separata flödes- och dataportar samt explicit förening av oberoende flöden.
- Energi: pris, effekt, prognos, termisk modell, lastfördelning och effekttoppsbegränsning.
- Säkerhet: kvalitetsgrind, förregling, återkopplingskontroll, dödband, ramp och kommando-timeout.

> **Livekommando:** Värdet till Sätt värde, Sätt variabel, puls eller annat kommando måste vara härlett från en Core-mapping eller en tillåten intern signalväg. Ett ensamt fast testvärde får inte bli ett skarpt kommando.

# Larm, larmgrupper och jour

## Skapa en larmregel

1. **Välj source mapping.** Regeln ska ha en tydlig signalägare och rätt enhet.
2. **Välj villkor.** Exempel: sant/falskt, lika/inte lika, över/under, mellan/utanför, dålig kvalitet, heartbeat eller intervallavvikelse.
3. **Sätt fördröjning.** Aktiv- och återställningsfördröjning minskar flapp.
4. **Välj nivå och kvittenspolicy.** Bestäm om larmet får stänga utan kvittens.
5. **Koppla larmgrupp eller mottagare.** Gruppens nivåpolicy kan styra mail och jour.
6. **Testa regeln.** Kontrollera villkor, text, mottagare och återställning innan aktivering.

## Heartbeat och liveness

WitVia använder protokollmedveten heartbeat/liveness. Ett oförändrat processvärde ska inte automatiskt bli inaktuellt när protokollet separat visar att anslutningen lever. Automatiska heartbeat-regler kan pausas men ska inte raderas utan analys.

## Jour

- Skapa lokala användare och grupper i System Admin.
- Skapa larmgrupp och nivåpolicy i Larmadmin.
- Koppla ett jourschema till bemannande användargrupp.
- Kontrollera vardag, helgdagar, undantag och aktuell ansvarig.
- Verifiera i Min jour att rätt person ser rätt aktiva larm.

# Energi

**Energi visas när funktionen är aktiverad. Core hämtar kompletta svenska 15-minutersscheman och kombinerar dem med lokala effektmätningar för kostnad och uppföljning.**

1. **Välj elområde.** SE1, SE2, SE3 eller SE4.
2. **Välj priskälla.** Välj tillgänglig källa enligt leveransomfattningen.
3. **Välj effektmapping.** Använd en verifierad total effekt i W eller kW med korrekt skala.
4. **Kontrollera historik.** Jämför effekt, kostnad och tidsperioder före styrning mot pris.
5. **Bygg energistyrning.** Använd kvalitetsgrind, effektreserv och anläggningens hårda gränser.

> **Kompletta dygn:** Sommartidsdygn kan innehålla 92, 96 eller 100 kvartal. Ofullständiga eller överlappande scheman ska behandlas som fel eller stale data - inte fyllas ut med antaganden.

# System Admin

## Enhet och uppdatering

- Värdnamn och tidszon.
- Stable eller beta som uppdateringskanal.
- Manuellt, timvis, dagligt eller veckovis schema.
- Automatiska uppdateringar och separat tillstånd för huvudversionsbyte.
- TLS 1.3 som standard; TLS 1.2 endast som avsiktligt kompatibilitetsläge.
- Valbar beständig signalhistorik med konfigurerbar retention.
- Kapacitetsgränser för HTTP, importfiler och Matter-commissioning.

## Nätverksändring med rollback

1. **Välj interface och IPv4/IPv6-läge.** Kontrollera adress, prefix, gateway och DNS.
2. **Välj Tillämpa med rollback.** Edge provar den nya NetworkManager-konfigurationen.
3. **Öppna kandidatadressen.** Använd länken i bannern från en betrodd klient.
4. **Bekräfta.** Välj Behåll inställningen. Annars återställs ändringen efter vald timeout.

## Lokala användare och grupper

| Roll | Huvudansvar |
|---|---|
| **Systemadministratör** | Nätverk, systeminställningar, larmmail samt lokala användare och grupper. |
| **Larmadministratör** | Larmregler, grupper, nivåpolicy och jourscheman. |
| **Jouranvändare** | Egna jourpass, tilldelade larm och kvittens enligt policy. |

# Larmmail

Alla larmregler använder en global mailtransport: central WitVia-mailtransport eller egen SMTP-server på Edge. Varje regel eller larmgrupp styr fortfarande mottagare och policy.

1. **Välj transport.** Central transport kräver inga lokala SMTP-hemligheter. Egen SMTP konfigureras på Edge.
2. **Ange SMTP-parametrar.** Server, port, avsändare, krypteringsläge och valfritt användarnamn.
3. **Spara lösenord via formuläret.** Tomt lösenord behåller befintlig hemlighet; radering kräver explicit val.
4. **Skicka testmail.** Testa samma transport innan skarpa regler aktiveras.
5. **Kontrollera outbox och fel.** Mail ligger kvar beständigt vid tillfälligt nätfel och återförsöks.

> **SMTP-säkerhet:** Autentisering över okrypterad transport avvisas. Alternativet Ingen kryptering är endast för en betrodd lokal relay utan autentisering.

# Uppdatering, backup och återställning

## Signerad uppdatering

- Updatern kör separat från Core och kontrollerar signerade manifest.
- Paketets signatur, storlek och SHA-256 verifieras före aktivering.
- Ny version packas i en egen versionskatalog och aktiveras atomiskt.
- Readiness måste bli grön inom tidsgränsen; annars återställs föregående version.
- Använd Uppdatera nu för en uttrycklig kontroll efter att driftfönster och backup har säkerställts.

## Backup

• Säkerhetskopiera installationens beständiga data enligt anläggningens backupplan när Core är stoppad eller genom en godkänd konsistent snapshotmetod.

- Skyddade hemlighetsfiler ska ingå endast i krypterad backup med strikt åtkomstkontroll.
- Dokumentera installerad version, nätverksinställning, source-exporter, ETS/templateunderlag och certifikatkedjor.
- TBR har egna beständiga data och credentials; backup och återställning ska behandlas separat från Core.

> **Återställning:** Kopiera inte en databas eller hemlighetsfil mellan två aktiva Edge-enheter utan en planerad identitets- och credentialprocess. Dubbletter kan skapa felaktigt ägande eller konkurrerande protokollidentiteter.

# Felsökning

| Symtom | Kontroll | Åtgärd |
|---|---|---|
| **Edge öppnas inte** | Ström, Ethernet, DHCP, DNS och HTTPS 443. | Prova IP-adress. Kontrollera lokal proxy och Core-service. |
| **Core svarar men källa är offline** | Source health, host/port, brandvägg och credentials. | Prova anslutningen, granska Feldiagnostik och motpartens logg. |
| **Livevärde saknas** | Mapping, signalKey, pollning/event och datatyp. | Kör explicit Läs eller discovery; kontrollera att mappingen är aktiv. |
| **Skrivning misslyckas** | Writable, måltyp, protokollbehörighet och read-back. | Testa i Source Admin före Bridge/Styrning. |
| **Värde blir inaktuellt** | Heartbeat/liveness, pollintervall och aktualitetspolicy. | Återställ anslutning eller justera policy med verifierat protokollunderlag. |
| **Matter commissioning stannar** | BLE-länk, setup-kod, IPv6, mDNS, tid och TBR. | Väck enheten, avbryt kontrollerat och försök med en transport i taget. |
| **Thread-enhet syns inte** | TBR service, radio, attach, dataset, UDP 5540 och discovery. | Verifiera hela TBR-readiness, inte bara processstatus. |
| **Mail uteblir** | Transport, testmail, mottagare, DNS och outbox. | Åtgärda SMTP/TLS eller central anslutning och invänta retry. |

## Avancerad servicestatus

```
systemctl status witvia-core.service
systemctl status witvia-updater.timer
curl -k https://localhost/health
curl -k https://localhost/health/ready
```

När WitVia-TBR ingår kan en behörig tekniker även kontrollera witthread-tbr.service och TBR-status. Ändra eller starta inte om tjänster enbart för att samla information; dokumentera först aktuell status och felbild.

# Teknisk referens

| Funktion | Normal port/transport | Kommentar |
|---|---|---|
| **Edge Console/API** | TCP 443 HTTPS | TCP 80 omdirigerar till HTTPS. |
| Lokala systemtjänster | Ej exponerade | Hanteras av installationen. |
| Mobil onboarding | Säker anslutning | Hanteras via WitVia-appen. |
| OCPP | WSS /ocpp/ | Charge point ID i sökvägen. |
| **Matter** | UDP 5540 | IPv6 och lokalt Matter-dataflöde. |
| **KNXnet/IP** | UDP/TCP 3671 | Beror på gateway och transport. |
| **BACnet/IP** | UDP 47808 | Projekt kan använda annan port. |
| **Modbus TCP** | TCP 502 | TCP Security normalt 802. |
| **MQTT** | TCP 1883/8883 | Plain/TLS enligt konfiguration. |
| **Sonos** | UDP 1900, TCP 1400 | SSDP och lokal UPnP/SOAP. |
| **PJLink** | TCP 4352, UDP discovery | Ingen transportkryptering. |
| **mDNS** | UDP 5353 | Discovery för lokala tjänster. |

## Health-endpoints

```
GET /health
GET /health/ready
```

/health visar övergripande driftstatus. /health/ready används för att kontrollera att tjänsten är redo vid start och uppdatering.

## Driftsättningskontroll

- Edge är märkt med värdnamn, IP, fysisk plats och ansvarig.
- HTTPS-adress, lokal administratör och MFA är verifierade.
- Tidszon, DNS, NTP, VLAN och brandvägg är dokumenterade.
- Core health och readiness är gröna efter omstart.
- Varje source connection har testad health, läsning och avsedd skrivning.
- Skrivbara mappings har ägare, gränser och read-back-test.
- Bridge-regler har testad riktning, loopskydd och audit.
- Styrningar är validerade, simulerade och liveprovade under kontrollerade förhållanden.
- Larm har testad aktivering, återställning, kvittens, mottagare och jourrouting.
- Uppdateringskanal, schema, backup och återställningsansvar är beslutade.
- När TBR ingår är radio, attach, child/parent, Matter UDP 5540 och discovery verifierade.

> **Överlämning:** Lämna aldrig anläggningen med tillfälliga credentials, öppna brandväggsregler, aktiva testflöden eller okvitterade säkerhetsavvikelser.

# Begrepp

| Begrepp | Betydelse |
|---|---|
| **Source connection** | En protokollanslutning eller fysisk/logisk enhet med egen lifecycle. |
| **Mapping** | Koppling mellan en protokollidentitet och WitVias gemensamma signalmodell. |
| Signalmodell | Gemensam beskrivning av datapunkter oavsett protokoll. |
| Aktuellt värde | Senast kända signalstatus som visas i gränssnittet. |
| **Bridge** | Regel som vidarebefordrar värden mellan mappings eller till en state-safe variabel. |
| **Styrning** | Grafisk lokal automationsmotor med validering, simulering och livekörning. |
| **Commissioning** | Säker process som lägger till en Matter-enhet i WitVias fabric. |
| **TBR** | Thread Border Router som routar mellan Thread och infrastrukturlänken. |
| **PICS** | Dokumenterad lista över implementerade BACnet-funktioner och roller. |
| **Readiness** | Kontroll att tjänsten är redo att användas, inte bara att processen kör. |

## Supportunderlag

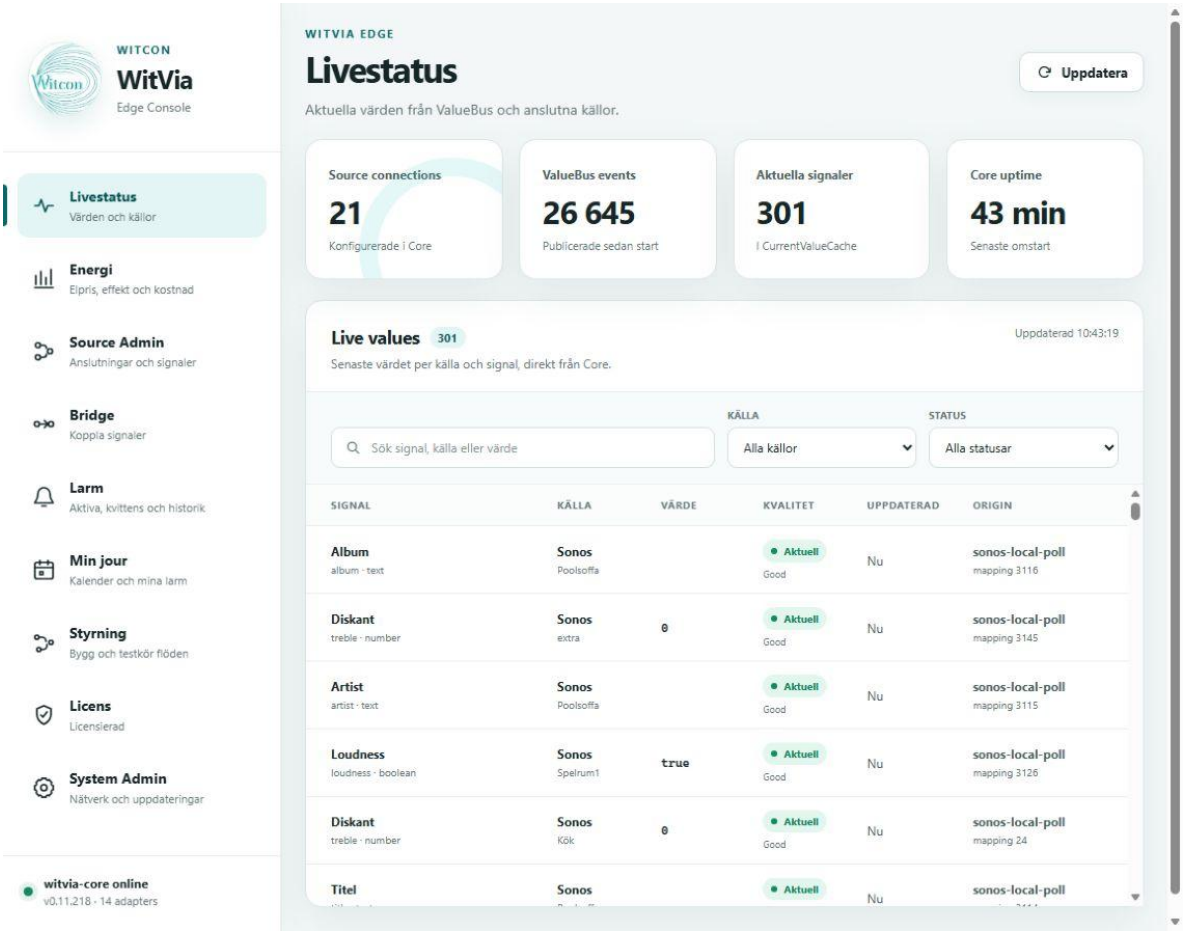
- Edge-värdnamn och Core-version.
- Tidpunkt med tidszon och exakt åtgärd som utfördes.
- Source-typ, anslutningsnamn och mappingnamn - inte lösenord eller nycklar.
- Felkod och meddelande från Feldiagnostik.
- Reproducerbara steg och om felet kvarstår efter kontrollerad omstart.

> **Sekretess:** Skicka aldrig setup-koder, local keys, nätverksnycklar, privata certifikatnycklar, TOTP-hemligheter eller återställningskoder i supportunderlag.

# Skärmbilder från Edge Console

Skärmbilderna är tagna från en driftsatt WitVia Edge med Core 0.11.218 den 20 augusti 2026. Tillgängliga vyer, värden och status kan variera med installerad version, aktiverade adaptrar, behörighet och leveransomfattning.

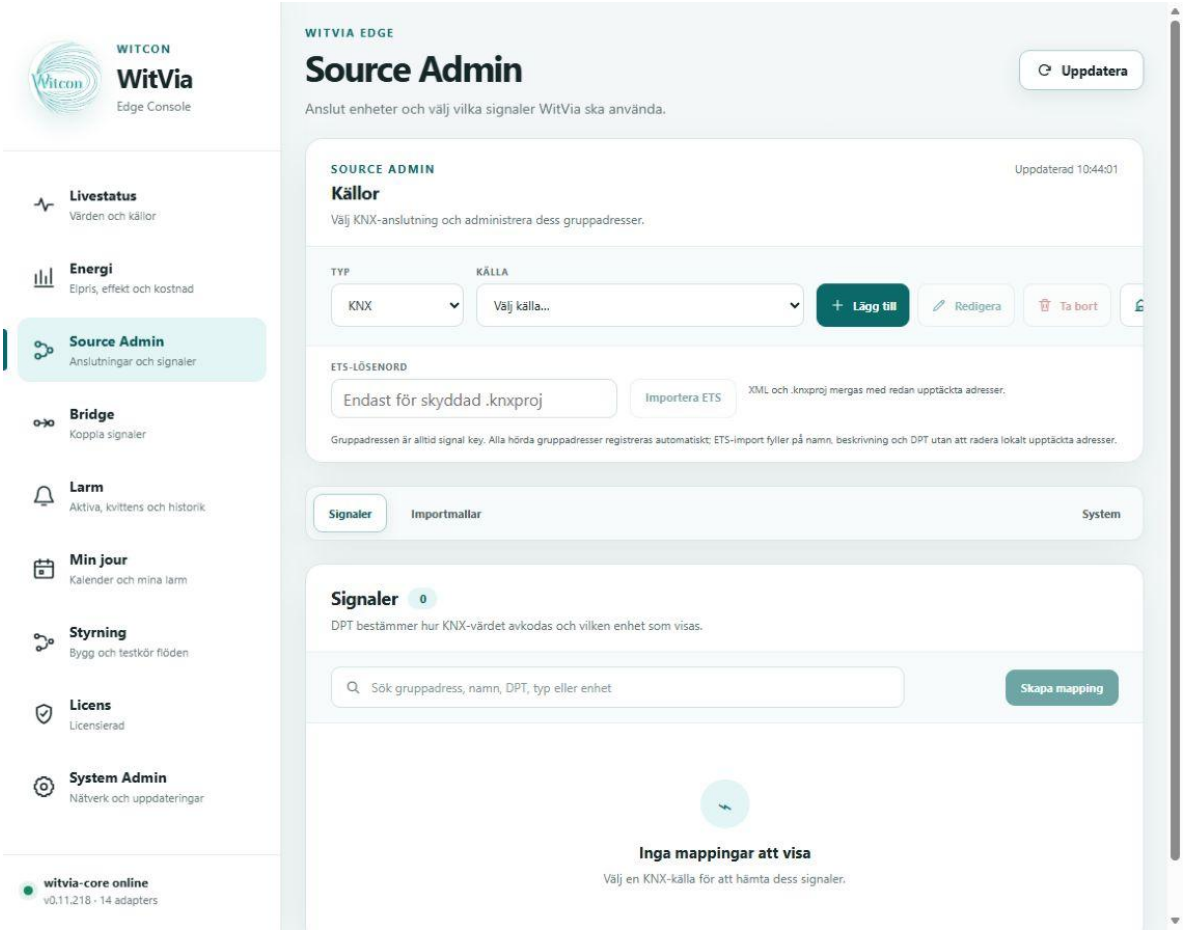
## Livestatus




**Figur 1. Livestatus samlar anslutningshälsa, signalmängd och aktuella värden.**

Kontrollera först statuskorten och filtrera sedan tabellen för att hitta en källa eller signal. Kvalitet och tidsstämpel hjälper dig avgöra om ett värde är användbart.

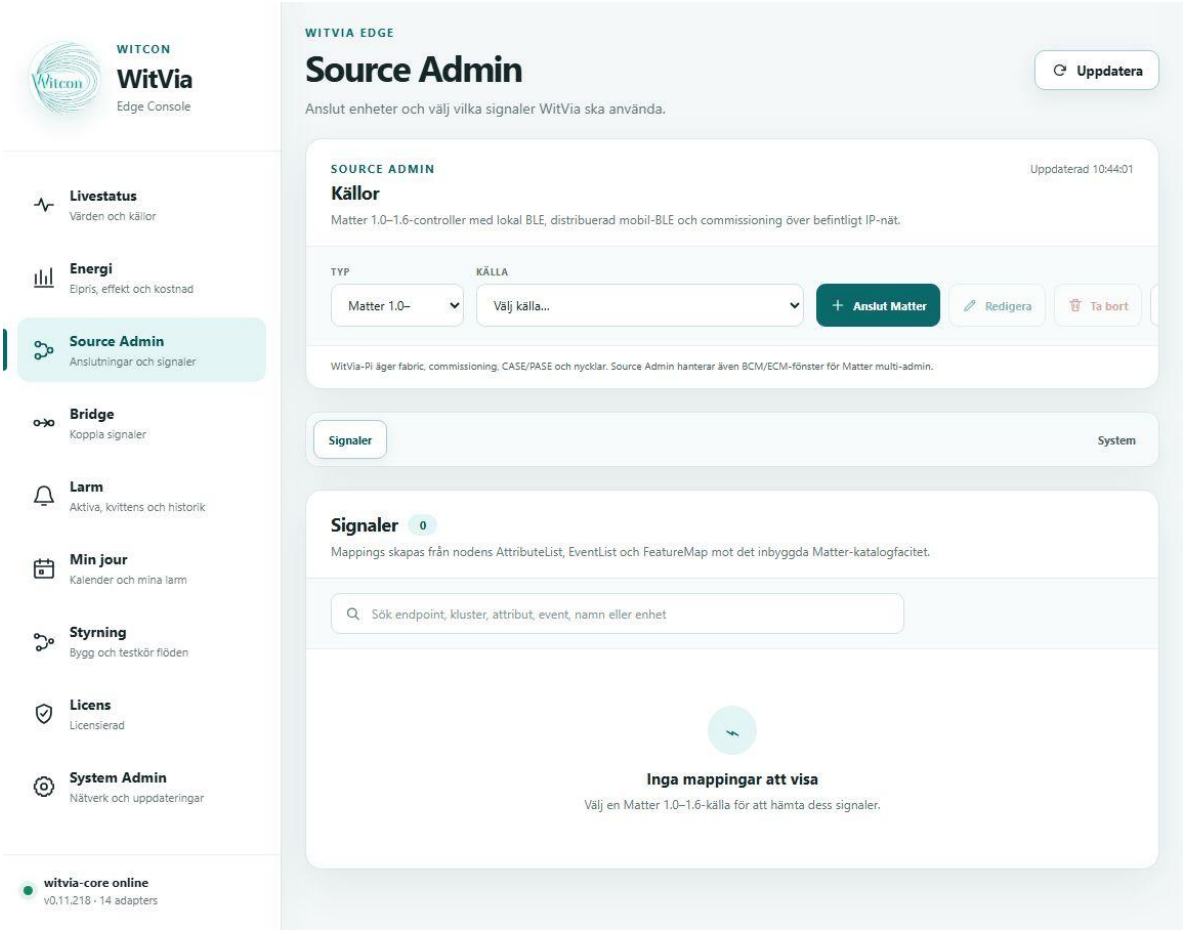
## Source Admin




**Figur 2. Source Admin används för att lägga till anslutningar och administrera signaler.**

Välj protokoll och källa i vänsterspalten. Därifrån kan du skapa anslutningar, importera underlag, upptäcka enheter och kontrollera källans signaler.

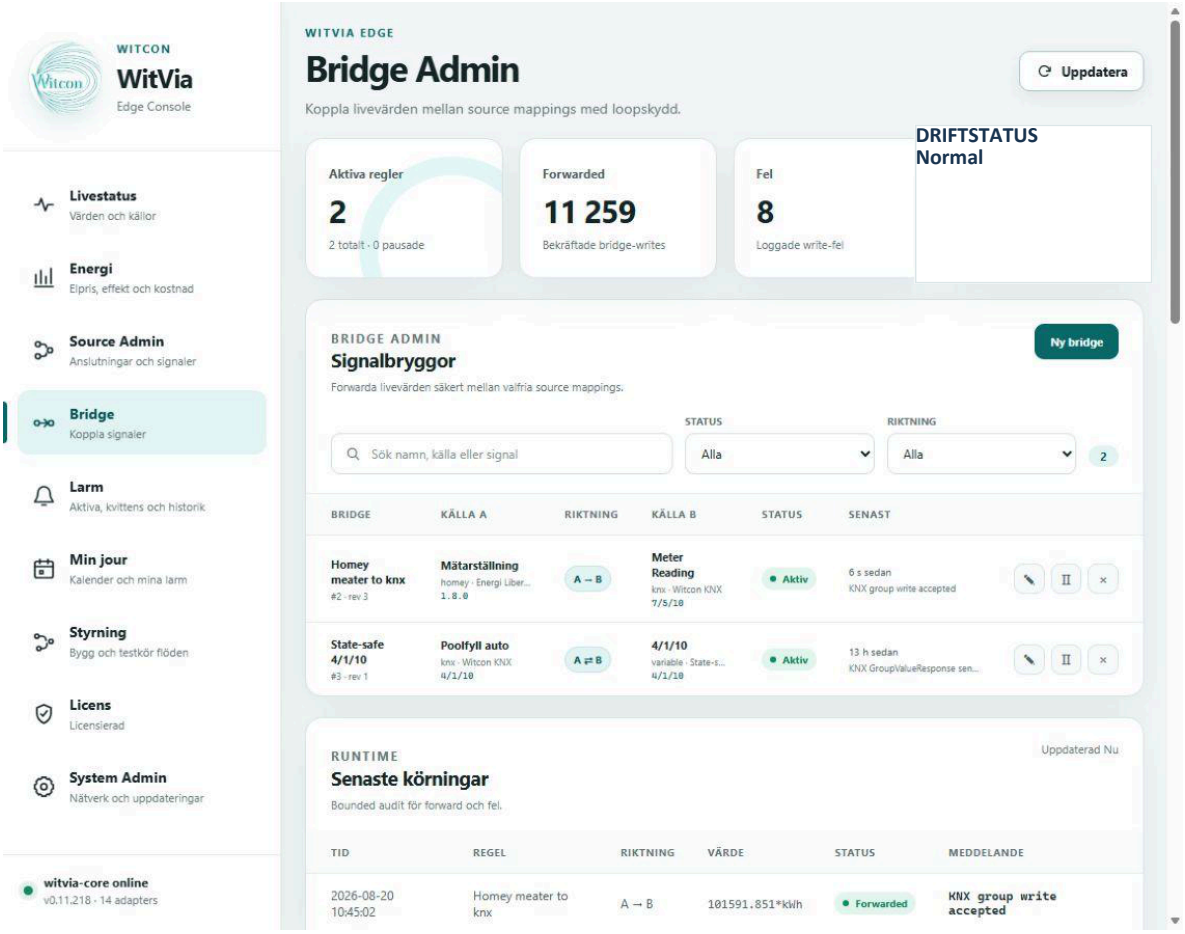
## Matter




**Figur 3. Matter-vyn visar commissioning och anslutna Matter-källor.**

Använd Anslut Matter när en ny enhet ska commissioneras. Enheten behöver vara i parningsläge och telefonen eller Edge-miljön måste ha rätt nätverksåtkomst.

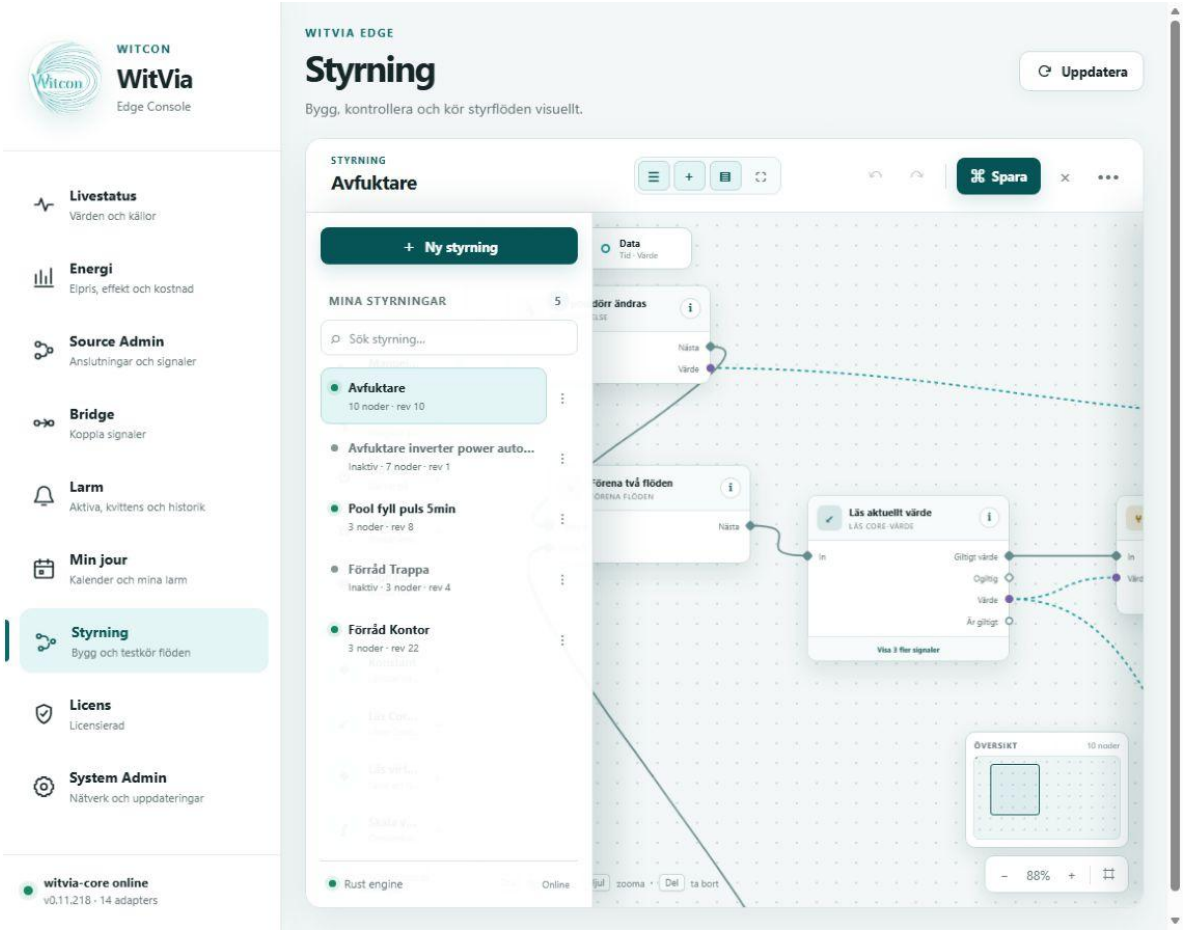
## Bridge




**Figur 4. Bridge visar aktiva regler, senaste körning och vidarebefordrade värden.**

Varje regel kopplar en källsignal till ett mål. Kontrollera status och senaste körning innan en regel lämnas i drift.

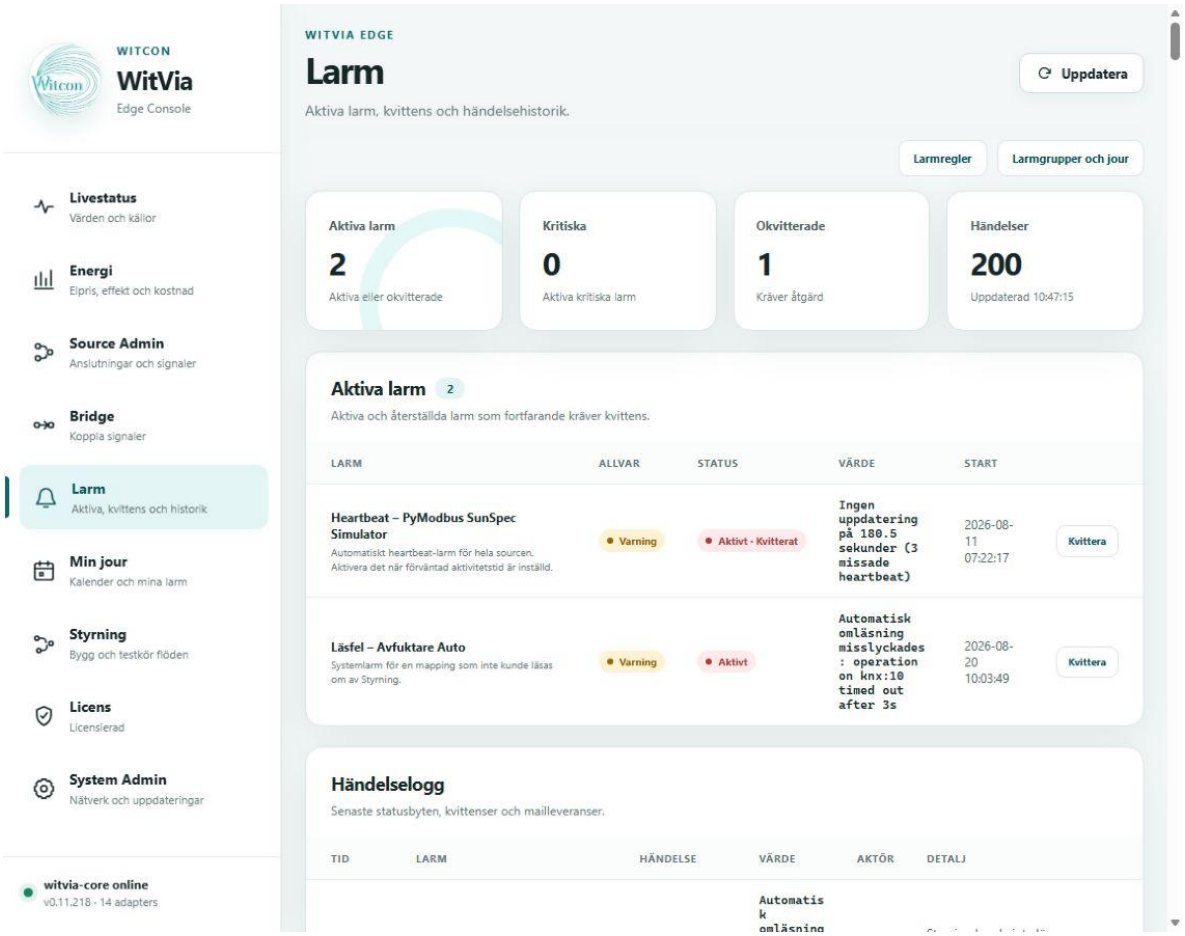
## Styrning




**Figur 5. Styrning bygger lokala flöden visuellt med noder och signalanslutningar.**

Öppna ett flöde, följ signalvägarna och kontrollera nodernas status. Testa alltid en förändring kontrollerat innan flödet används mot skarp OT-utrustning.

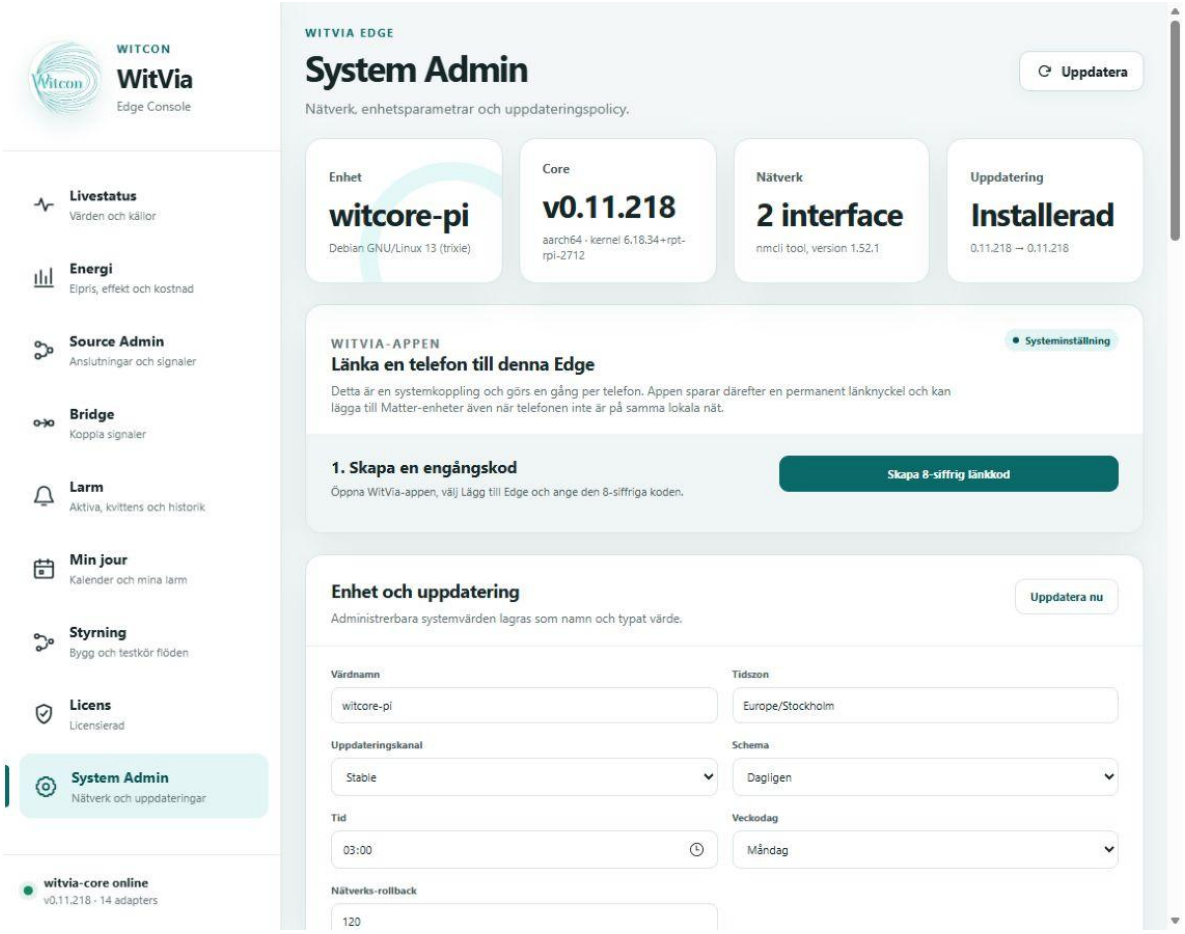
## Larm




**Figur 6. Larmöversikten sammanfattar aktiva, kritiska och okvitterade larm.**

Öppna ett larm för att bedöma värde, starttid och allvar. Kvittering bekräftar hantering men ändrar inte den bakomliggande signalen eller orsaken.

## System Admin




**Figur 7. System Admin visar Edge-version, nätverksstatus och uppdateringspolicy.**

Här kontrollerar du Core-version, enhetsstatus och uppdateringskanal. Nätverksändringar ska göras med rollback och verifieras från en separat klient.