Privitty

Secure Edge Access & Industrial Data Integration

Whitepaper -- Architecture, Security, and Deployment

Version: Draft 0.2
Prepared by: Alanring Technologies -- “Privitty”
https://privitty.com

“Machine data flows freely -- access never does”

Table of Contents

Executive Summary

Industrial operators and OEM machine builders face a common challenge: engineers, service technicians, and monitoring systems need reliable access to factory-floor equipment -- programmable controllers, HMIs, and SCADA nodes -- from locations outside the plant. Traditional approaches such as VPN, open firewall ports, or jump-server arrangements introduce unacceptable risk to operational technology (OT) networks, create compliance gaps, and impose administrative overhead that growing fleets of equipment cannot absorb.

Privitty is an edge-native, end-to-end encrypted (E2EE) platform designed specifically for this problem. It gives remote engineers and operators secure, governed access to factory systems without opening a single inbound firewall port on the plant network. Encrypted file transfer with cryptographic access control replaces untracked USB sticks and FTP sessions for program distribution. A dedicated management console gives administrators full visibility over every device, session, and transfer -- with instant revocation capability.

This whitepaper describes Privitty's architecture, security model, and two validated deployment patterns for industrial environments:

- Secure remote HMI access and industrial data integration -- an E2EE gateway on the factory floor provides encrypted VNC/RDP access to HMIs and continuous OPC UA data collection without exposing the plant network.
- Secure program transfer -- remote engineers deliver controller and HMI project files to the edge over E2EE, with per-file access control, time limits, and true revocation capability.

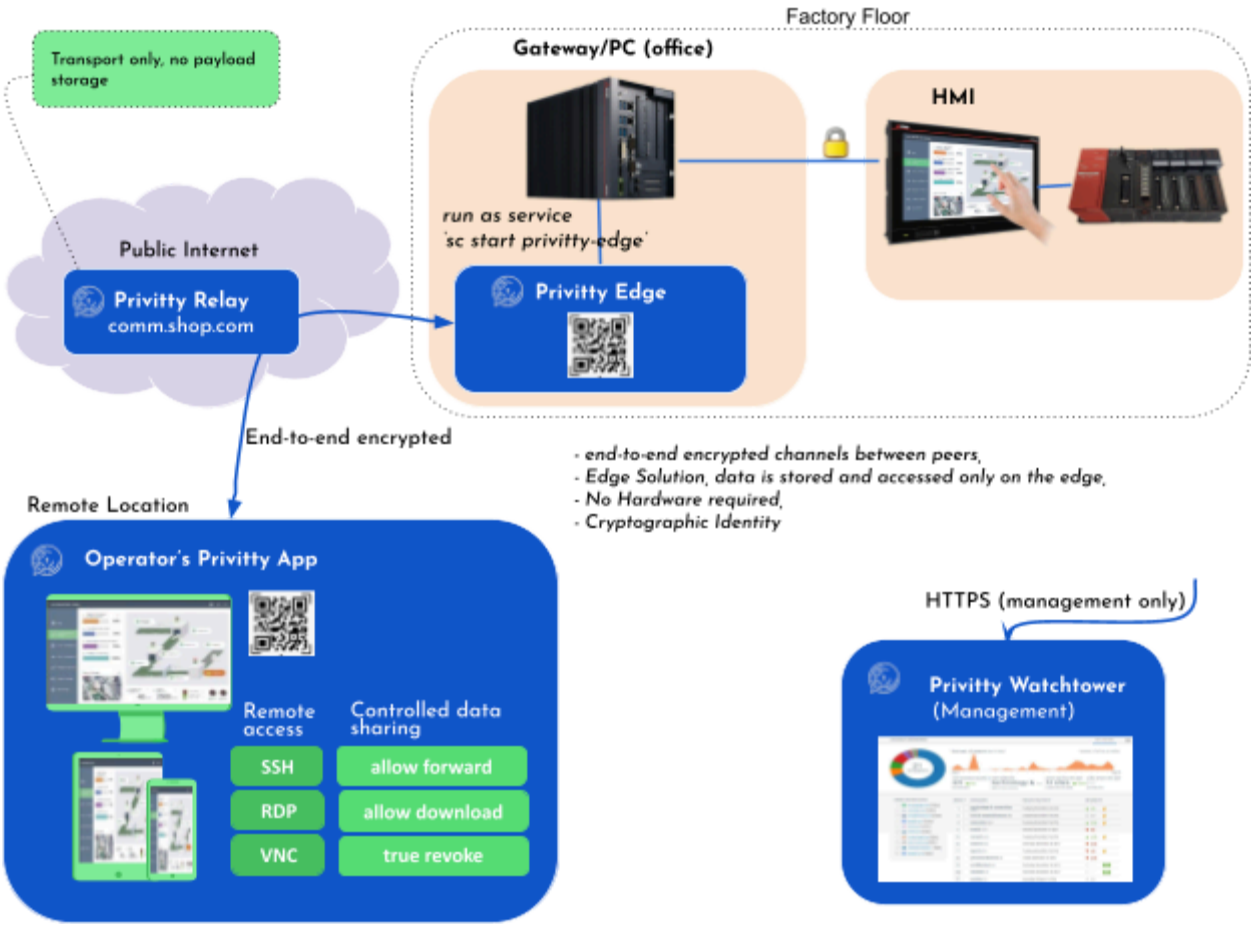


Fig 1: Platform Overview

The Challenge: Secure Remote Access in OT Environments

Operational technology networks were designed for isolation. As remote operations, predictive maintenance, and global engineering teams have become the norm, the gaps in conventional remote-access approaches have grown untenable:

- Open inbound ports: VPN concentrators and port-forwarded RDP/VNC expose the plant network directly to the internet. A single misconfiguration or compromised credential is all an attacker needs.
- No true access control: VPN grants network access, not per-device, per-session access. Operators retain access indefinitely -- revocation is manual and often missed.
- Uncontrolled file transfer: USB drives and email attachments carry intellectual property (IP) without encryption, access logging, or expiry. There is no way to withdraw a file once delivered.
- Compliance gaps: IEC 62443, NIS2, and NIST 800-82 require defence-in-depth, least-privilege access, MFA, and full audit trails. Most field-deployed VPN arrangements do not satisfy these requirements.
- Operational visibility: Administrators cannot see which sessions are active, which files have been transferred, or whether access policies are being observed -- until an incident occurs.

Key insight: Every industrial remote-access security incident analysed by leading OT security researchers traces back to one of three root causes -- an exposed network service, an over-privileged credential, or an uncontrolled file. Privitty eliminates all three by design.

The Privitty Platform

Privitty is a three-component software platform. No new hardware is required. The three components work together to give operators the access they need while giving administrators the visibility and control the plant demands.

Component	Where it runs	Who uses it	Core function
Privitty Edge	Factory-floor gateway PC	Factory asset (automated)	E2EE gateway -- tunnels, file I/O, OPC UA, local API
Privitty App	Remote engineer's phone/device	Remote operator/engineer	Encrypted access -- file transfer, RDP/VNC/SSH tunnels
Privitty Watchtower	Web browser (any location)	Administrator / IT-OT team	Governance -- fleet management, sessions, telemetry, revoke

Privitty Edge

Privitty Edge is a lightweight software service deployed on the industrial gateway PC on the factory floor -- the same PC that already hosts remote-HMI client software, engineering tools, or OPC UA integration. No dedicated hardware purchase is required.

Once installed, Privitty Edge establishes an outbound-only encrypted connection to a partner-private relay. Operators and data consumers on any internet-connected device can reach factory systems through that channel -- without any inbound firewall rule on the plant network, and without the Edge PC needing a public IP address.

Remote access tunnels

Privitty Edge acts as the protocol bridge on the factory side. When an operator opens a tunnel from Privitty App, Edge connects to the local LAN target (HMI, controller, edge PC desktop) and carries the session over the E2EE channel. Three tunnel types are supported out of the box:

Tunnel	Typical target	Use case
RDP	Edge PC remote desktop	Full engineering workstation session; run vendor tools, apply programs

Tunnel	Typical target	Use case
VNC	HMI built-in VNC server	Real-time HMI screen view and interaction without physical presence
SSH	Edge PC or gateway OS	Scripting, automation, or port-forwarding for advanced workflows

All three protocols traverse the E2EE channel. The cleartext protocol never reaches the internet -- it is terminated at the Edge on the LAN side and at the operator's device on the remote side. There is no VPN, no TLS terminating proxy, and no cloud component that can read session content.

Encrypted file transfer

Program packages, recipe files, firmware, and other engineering artefacts are delivered to the Edge over the same E2EE channel. Every file transfer carries configurable access policies:

- Download permission -- the administrator can prevent the recipient from saving a local copy.
- Forward restriction -- the file cannot be forwarded outside the intended recipient.
- Time-bound access -- access expires automatically at a configurable time.
- True revocation -- access can be withdrawn after delivery, including files already received.

Files are encrypted end-to-end before leaving the operator's device. The relay never stores or reads file content. Decrypted programs are only available on the Edge, preserving IP on the factory floor.

OPC UA industrial data collection

For environments where continuous monitoring is required, Privitty Edge includes an integrated OPC UA client. It connects to the HMI's built-in OPC UA server on the shop-floor LAN, subscribes to controller tags using the OPC UA MonitoredItems mechanism, and delivers structured tag data, threshold alarms, and OPC UA Alarms & Conditions events to Privitty Watchtower -- without any inbound connection to the plant.

Data type	How Privitty Edge delivers it
Controller tag values	OPC UA subscription; configurable polling / on-change
Threshold alarms	Per-tag rules; alarm event generated and forwarded when value crosses threshold
OPC UA Alarms & Conditions	Part 9 subscription; severity, active state, acknowledgement forwarded
Session & transfer metadata	Per-session log: tunnel protocol, timestamps, operator identity, file reference
Gateway system health	CPU, RAM, storage, heartbeat -- forwarded to Watchtower

Resource footprint

Privitty Edge is engineered to co-exist with existing software on constrained industrial hardware. Typical footprint on a Raspberry Pi 4-class or equivalent gateway:

Condition	RAM	CPU (Pi 4-class)
Idle	~30-40 MB	< 1%
Active tunnel	~50-70 MB	2-15% (protocol-dependent)
Active tunnel disk I/O	Zero	Tunnel is a pure in-memory byte bridge

Privitty App

Privitty App is the remote operator's client. It is available as a white-label Android application -- partner-branded for distribution through partner channels or the Google Play Store. Desktop and iOS clients are on the partner roadmap.

The App is the single tool an operator needs to:

- Establish a verified, encrypted connection to a specific Privitty Edge unit on the factory floor.
- Send program files and engineering artefacts to the Edge with configurable access policies.
- Open RDP, VNC, or SSH tunnels to reach HMI screens and engineering tools remotely.
- View and interact with access-controlled files inside the App's secure viewer.
- Respond to access invitations and time-bounded sessions created by the administrator.

Pairing and Identity

Before an operator can reach any Edge device, they must be explicitly provisioned by an administrator. Pairing uses Privitty's SecureJoin protocol -- a QR-code based handshake that establishes a cryptographically verified, mutually authenticated E2EE relationship between the operator's App and a specific Edge unit. No shared passwords, no VPN credentials, no static IP addresses are exchanged.

Zero implicit trust: An operator who has access to one Edge device has no access to any other Edge device unless the administrator explicitly grants it. Access is per-operator, per-device.

Access-controlled file viewer

Files received in Privitty App are opened in a built-in secure viewer that enforces the access policies set at transfer time. If the administrator has disabled download and forwarding, the

operator can review the file contents but cannot extract it from the App or share it to other applications -- even if device backup or screen capture is attempted.

Privitty Watchtower

Privitty Watchtower is the administrator's web console. Each customer or OEM partner receives a dedicated Watchtower instance -- an isolated deployment with its own data store, accessible at a unique URL. It is the single pane of glass for fleet governance, operator provisioning, telemetry review, and incident response.

What administrators can do in Watchtower

Function	What it enables
Fleet dashboard	Live view of all Edge devices, connection status, licence utilisation, active pairings
Edge device management	Register new Edge units, view last-seen status, revoke devices from the fleet
Operator provisioning	Add operators by email; send licence and pairing invite; set access windows (time-bounded)
Telemetry & OPC UA	Per-edge tag browser, live sample view, alarm history, configurable threshold rules
Session audit	Every tunnel session and file transfer logged with operator identity, timestamps, protocol
Revocation	Instantly revoke file access or terminate a pairing -- takes effect immediately on the Edge
Multi-factor authentication	Admin login protected by TOTP authenticator app and email one-time password (OTP)
Licence management	View seat allocation, push licence QR to operators, track activated devices vs. quota

Operator provisioning workflow

Setting up a new remote operator takes four steps -- no IT involvement required:

1. Administrator logs into Watchtower and selects the target Edge device.
2. Administrator sends a pairing invite to the operator's email address. The email contains a QR code.
3. Operator opens Privitty App, scans the QR code. A verified E2EE relationship is established with that Edge unit.
4. Administrator optionally sets an access window (e.g. valid for one business day) and the operator's permitted actions.

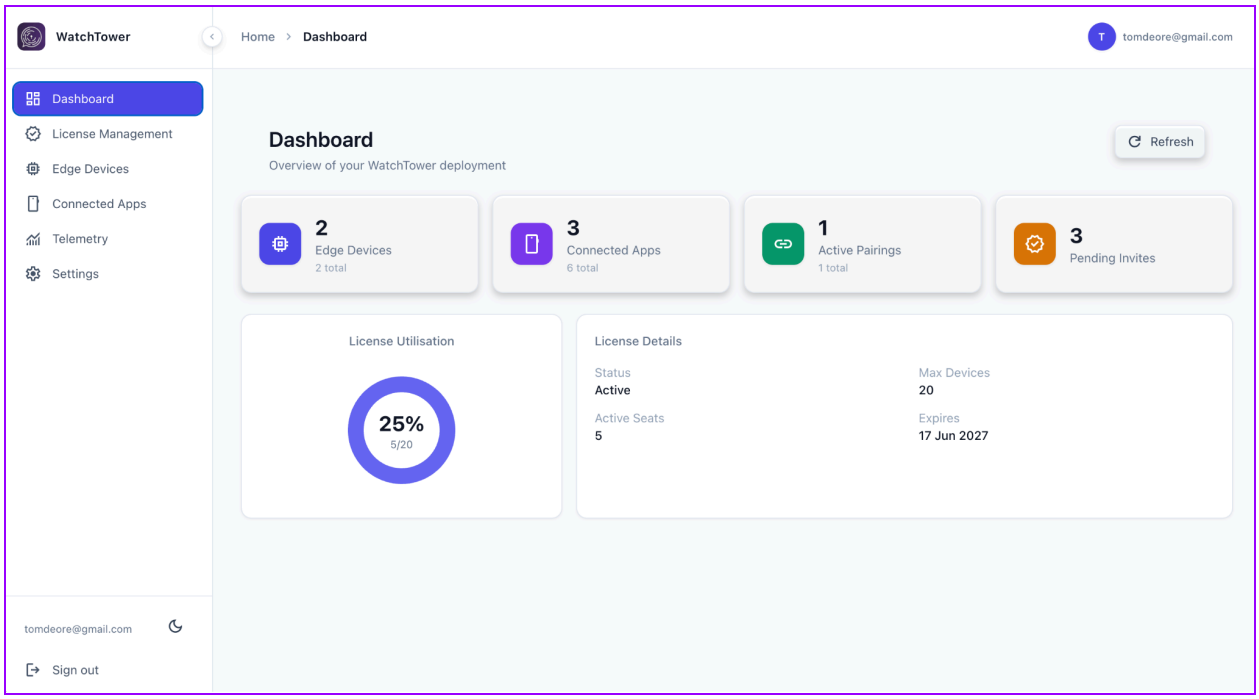


Fig 2: Privitty Watchtower – Dashboard

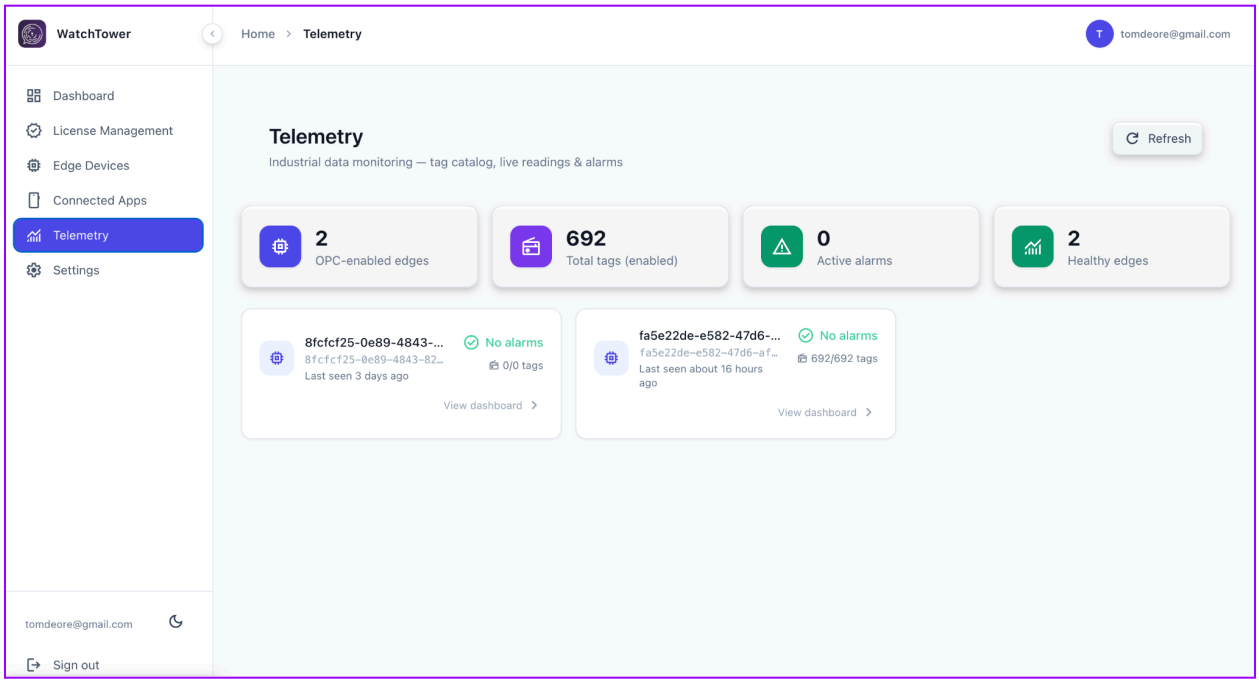
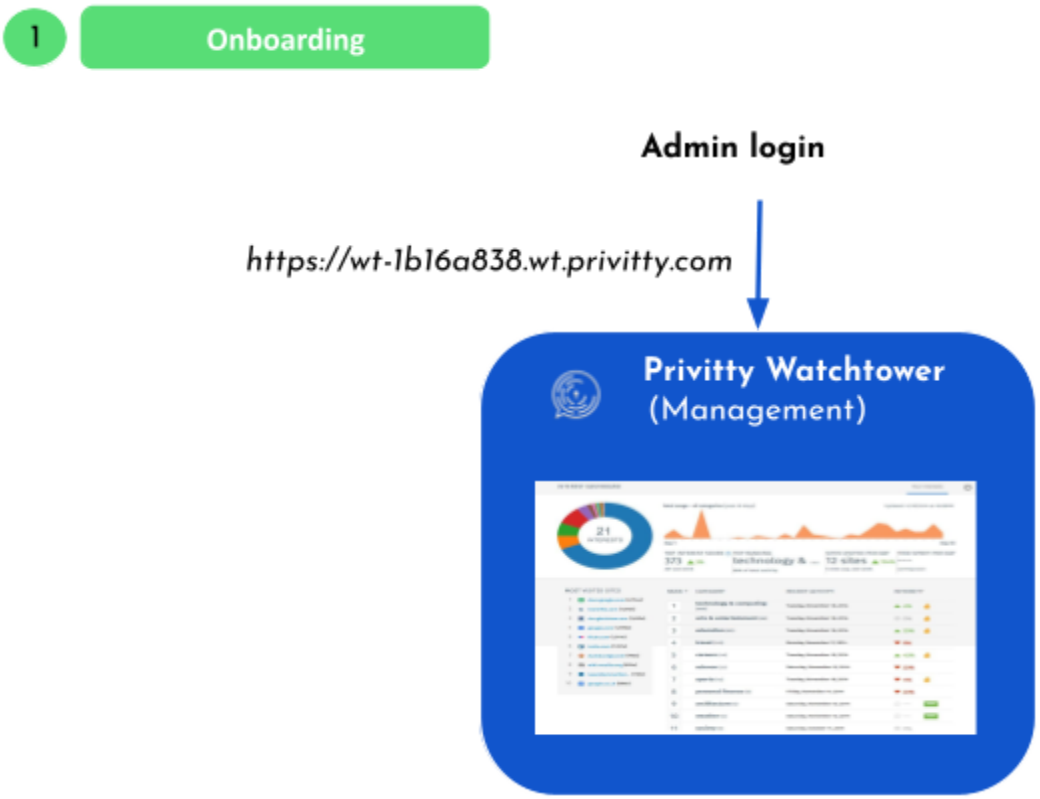


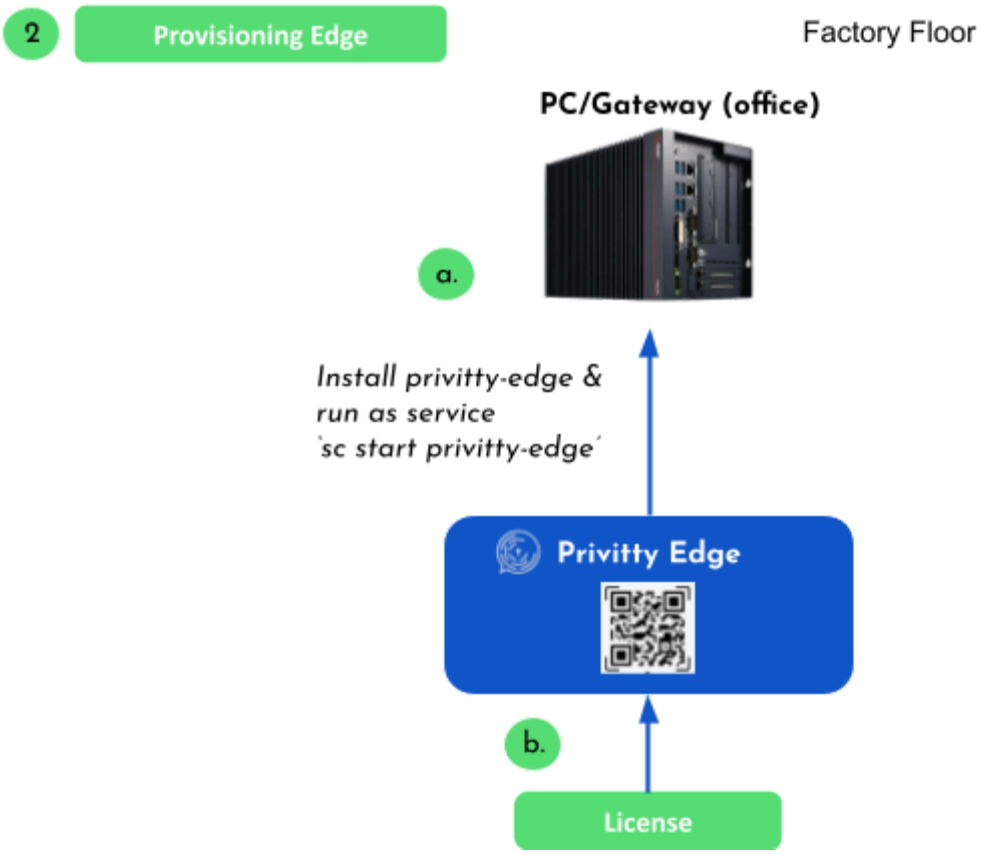
Fig 3: Watchtower – Telemetry dashboard

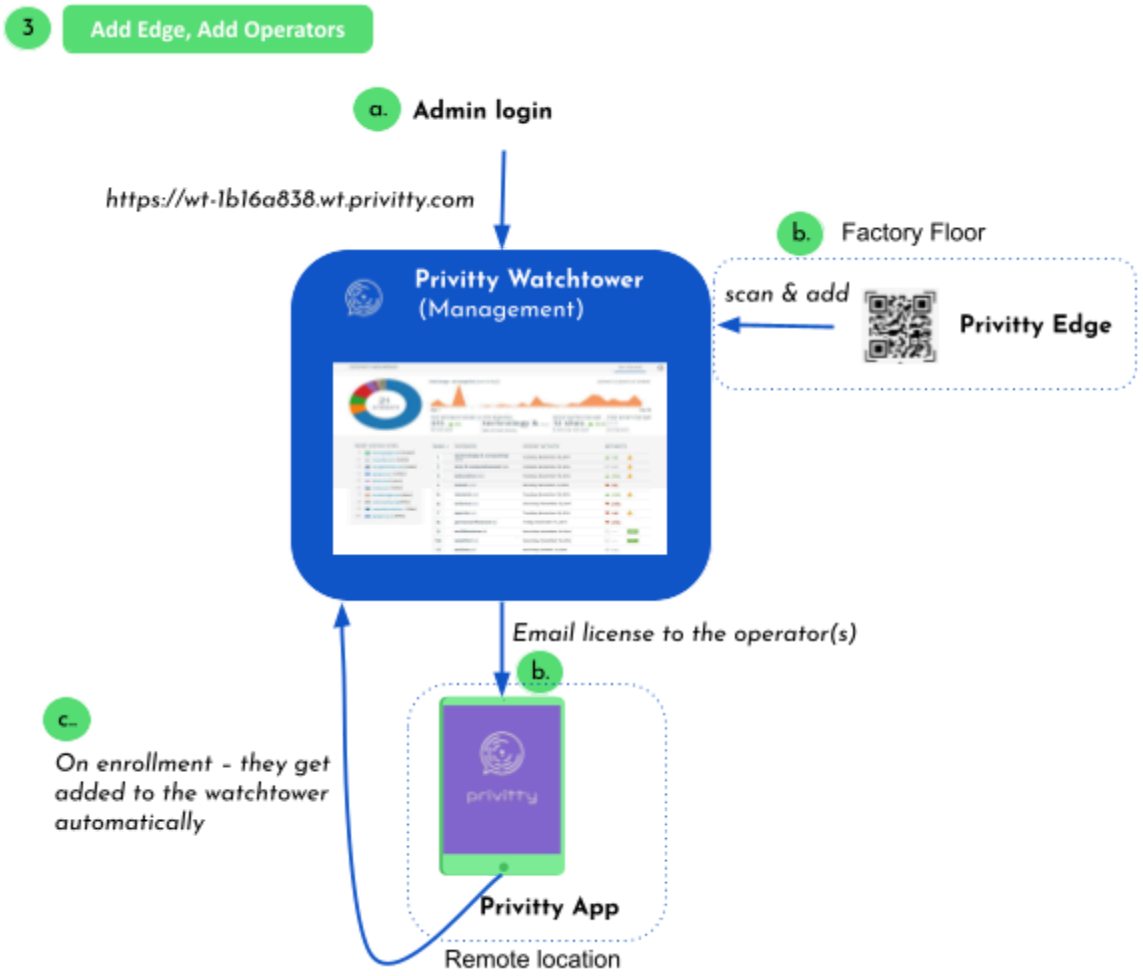
End-to-End: How a Remote Session Works

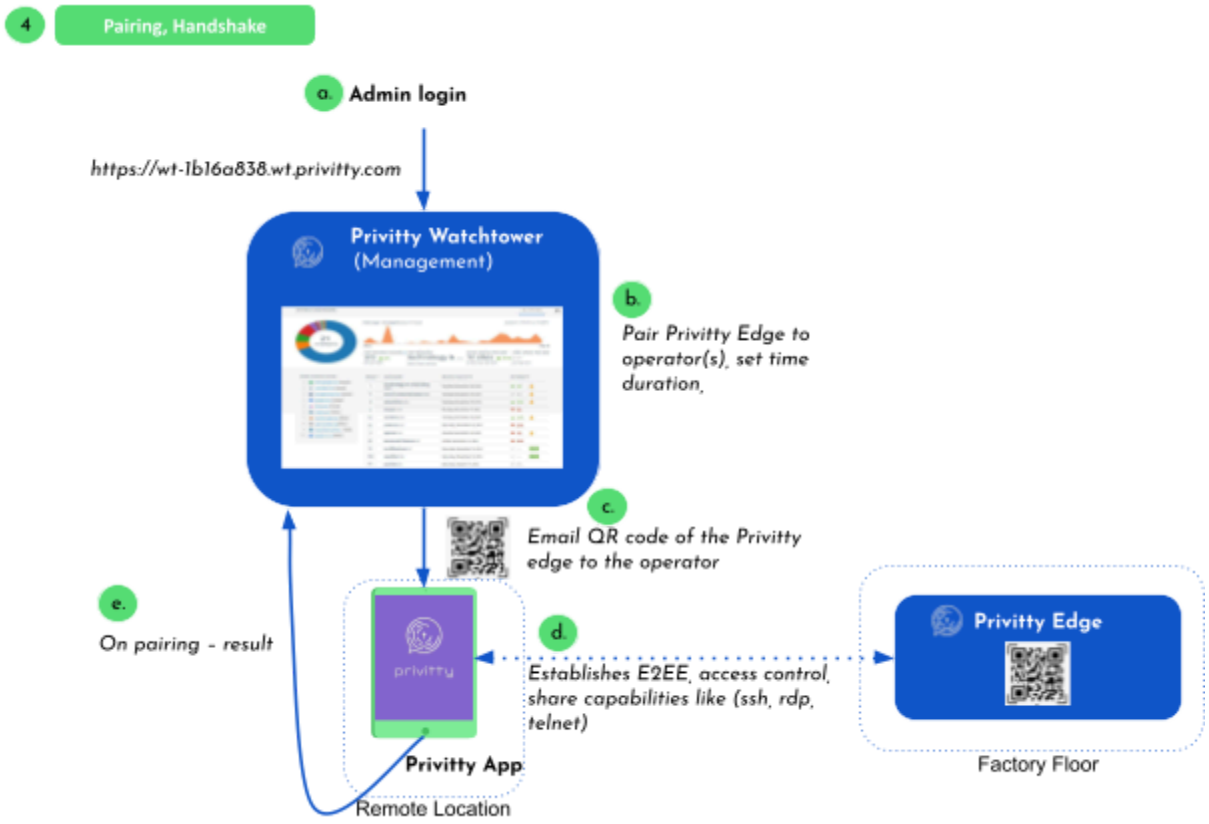
The following describes a complete remote program transfer and verification session. All steps use only Privitty App, Privitty Watchtower, and vendor engineering tools -- no VPN client, no firewall change, no jump host.

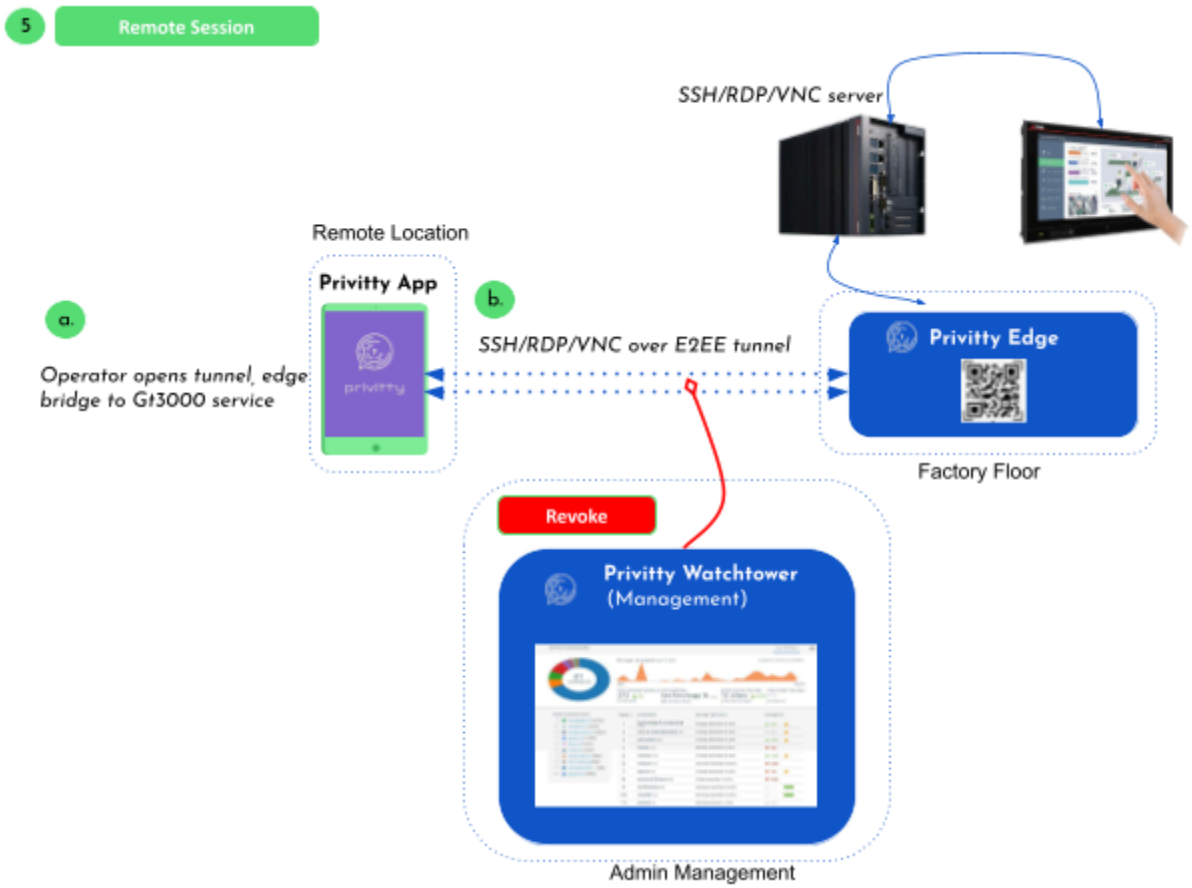
1. **Administrator provisions**. Admin registers the Edge unit in Watchtower and sends a pairing invite to the engineer's email.
2. **Engineer activates**. Engineer opens Privitty App, scans the QR from the invite email. E2EE pairing established.
3. **Engineer transfers file**. From Privitty App, engineer selects the controller/HMI program package and sends it to the Edge with chosen access policies (e.g. no forward, valid for 4 hours).
4. **Edge receives and stages**. Privitty Edge receives the E2EE package, decrypts it locally, and stages it per access policy. No copy exists in the cloud or on the relay.
5. **Engineer opens RDP tunnel**. Engineer taps RDP in Privitty App. An E2EE tunnel opens to the Edge PC. The full engineering desktop appears.
6. **Apply and verify**. Engineer opens vendor engineering tool on the Edge PC, loads the program from the staged location, and applies it to the controller/HMI over the LAN. Verification uses the remote-HMI client or VNC tunnel.
7. **Administrator monitors**. Throughout, Watchtower shows the active tunnel, file transfer status, and any OPC UA tag changes confirming successful program apply.
8. **Revoke and close**. Engineer closes the tunnel. Administrator (or engineer) revokes file access in App. The session is logged. The access window expires automatically.

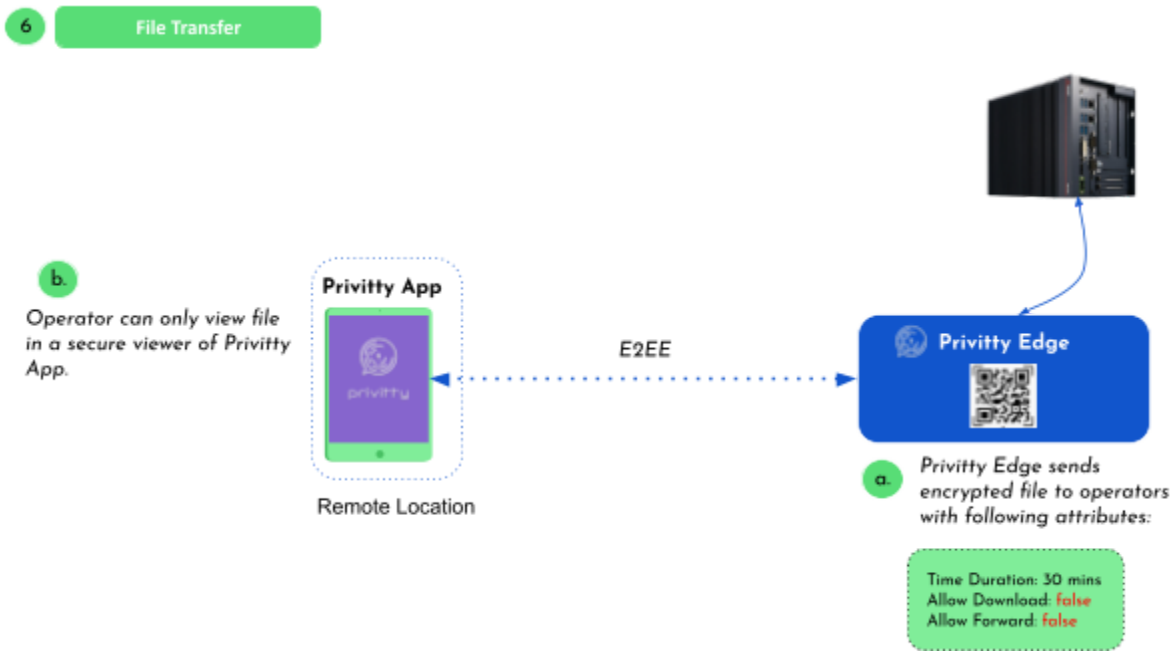



2
Provisioning Edge
Factory Floor
PC/Gateway (office)
a.
Install privitty-edge &
run as service
'sc start privitty-edge'
Privitty Edge
b.
License


3
Add Edge, Add Operators
a.
Admin login
https://wt-1b16a838.wt.privitty.com
b.
Factory Floor
Privitty Watchtower
(Management)
scan & add
Privitty Edge
Email license to the operator(s)
b.
c.
On enrollment - they get added to the watchtower automatically
privitty
Privitty App
Remote location


4
Pairing, Handshake
a. Admin login
https://wt-1b16a838.wt.privitty.com
Privitty Watchtower
(Management)
b.
Pair Privitty Edge to operator(s), set time duration,
c.
Email QR code of the Privitty edge to the operator
e.
On pairing - result
privitty
Privitty App
Remote Location
d.
Establishes E2EE, access control, share capabilities like (ssh, rdp, telnet)
Privitty Edge
Factory Floor


5
Remote Session
SSH/RDP/VNC server
Remote Location
Privitty App
b.
a.
Operator opens tunnel, edge bridge to Gt3000 service
SSH/RDP/VNC over E2EE tunnel
Privitty Edge
Factory Floor
Revoke
Privitty Watchtower
(Management)
Admin Management


6
File Transfer
b.
Operator can only view file in a secure viewer of Privitty App.
Privitty App
E2EE
Privitty Edge
Remote Location
a.
Privitty Edge sends encrypted file to operators with following attributes:
Time Duration: 30 mins
Allow Download: false
Allow Forward: false

Security Architecture

Privitty's security model is built on the principle that no single compromised component should be sufficient to gain access to plant equipment. The platform implements defence in depth across identity, transport, access control, governance, and network exposure.

Security layers

Layer	Mechanism	What it prevents
Identity & device binding	Cryptographic device identity (UUID + key pair) established at activation; SecureJoin for operator pairing	Impersonation; rogue Edge units; unauthorised operators
End-to-end encryption	ChaCha20-Poly1305 + BLAKE3 + X25519; operator-to-edge; relay cannot decrypt	Network eavesdropping; relay compromise; man-in-the-middle
Per-file access control	Download, forward, expiry, and true revoke enforced cryptographically	IP leakage; unauthorised redistribution; stale access
Operator access windows	Administrator sets time-bounded pairing validity per operator per Edge	Persistent access after engagement ends; role creep
Session audit & logging	Every tunnel and transfer recorded in Watchtower; operator identity, timestamps, protocol	Undetected lateral movement; deniability; post-incident analysis gaps
Outbound-only connectivity	Edge initiates all connections; no inbound ports on plant firewall or NAT	Internet-exposed OT services; direct attack surface on plant LAN
Admin MFA	Watchtower login requires TOTP + email OTP	Admin credential theft; portal takeover
Relay isolation	Each customer operates on a private relay; transport-only; no payload retention	Cross-tenant data leakage; relay operator access to content

Data residency

Data type	Where it lives
Controller/HMI program files (decrypted)	Edge device only -- never in cloud
Encrypted payloads in transit	Ephemeral on relay -- not stored, not logged
OPC UA tag telemetry	Customer's Watchtower instance (customer-controlled retention)
Session & audit metadata	Customer's Watchtower instance

Data type	Where it lives
Operator cryptographic keys	Operator device only
Edge cryptographic keys	Edge device only

Alignment with IEC 62443 and industry standards

Privitty's design is intentionally aligned with the security requirements of IEC 62443 for industrial automation and control systems. The table below maps key platform capabilities to IEC 62443 requirements and related standards.

Privitty Capability	IEC 62443 requirement	Related standard
Cryptographic device identity	IAC-2: Authenticator management	IEC 62443-3-3
SecureJoin operator pairing	IAC-3: Unique identification + authentication	IEC 62443-3-3
Admin MFA	IAC-7: Strength of password-based authentication	NIS2 Article 21; IEC 62443-2-4
Least-privilege access windows	AC-3: Use control -- authorised access only	IEC 62443-3-3; NIST 800-82
True file revocation	AC-4: Information flow enforcement	IEC 62443-3-3
Outbound-only / no inbound ports	AC-17: Remote session establishment	IEC 62443-3-2 zone/conduit model
Full session audit in Watchtower	AU-6: Audit review / DM-3: information management	IEC 62443-3-3; NIST 800-82
Private relay -- no data retention	DM-2: Information preservation	IEC 62443-3-3

Deployment Patterns

Pattern A -- Secure HMI Access and OPC UA Data Integration

Privitty Edge runs on a gateway PC co-located on the shop-floor subnet with the industrial HMI. The HMI operates in transparent gateway mode, collecting data from PLCs via native field-bus protocols and exposing tags over its built-in OPC UA server.

This pattern delivers two capabilities in a single Edge deployment:

- Remote HMI access -- operator opens a VNC tunnel in Privitty App; reaches the HMI's native VNC server via Edge without any inbound firewall rule.

- Continuous data path -- Edge OPC UA client subscribes to HMI tags; data flows to Watchtower for live monitoring, threshold alerting, and alarm history.

Pattern B -- Secure Remote Program Transfer

Privitty Edge runs on an engineering edge PC at the factory. Remote engineers send controller and HMI program packages via Privitty App, with cryptographic access policies. Engineers then open an RDP tunnel to the edge PC to run vendor engineering tools, apply the program to the controller or HMI over the LAN, and verify operation via a remote-HMI client.

This pattern replaces USB sticks, FTP, and VPN-based program delivery with a fully auditable, access-controlled, revocable engineering workflow.

What Privitty does not do

Understanding the boundary is important for integration planning:

- Privitty does not interpret or modify vendor-specific engineering file formats.
- Privitty does not write directly to controller memory -- program application remains with vendor tools on the edge PC or over the LAN.
- Privitty does not store production programs or session payloads in the cloud.
- Privitty does not require changes to the plant's existing field-bus wiring or controller configuration.

OEM and White-Label Deployment

Privitty is designed to be integrated into an OEM or system integrator's solution stack and delivered to end customers under the partner's brand.

Deliverable	White-label options
Privitty App	Partner-branded mobile application; partner-controlled distribution
Privitty Edge	Partner-branded service; partner-defined gateway image or installer
Privitty Watchtower	Dedicated instance per customer at a partner-controlled domain
Privitty Relay	Partner-operated relay cluster; no data shared with Privitty production infrastructure
Documentation	Partner-branded deployment, operator, and security guides

Data sovereignty: Because each customer operates on a private relay and a dedicated Watchtower instance, production program IP and operational telemetry never co-mingle with another customer's data. The partner organisation retains full control over infrastructure, data retention policies, and user access.

Proof of Concept Plan

A structured proof of concept validates both deployment patterns against the partner's specific HMI, controller, and engineering toolchain environment.

Proof-of-concept environment

Item	Specification
Edge hardware	Any Windows IoT Enterprise industrial PC or Linux industrial PC or gateway (ARM or x86) -- existing hardware or equivalent
Relay	Partner-private relay (staging instance)
Watchtower	Partner dedicated staging instance
Operator devices	One mobile + one desktop
Field equipment	Industrial HMI and controller (or lab equivalents)
Engineering software	Remote-HMI client, vendor engineering / design tools (as applicable)

Success criteria

#	Criterion	Verification
1	Administrator registers edge and operator in Watchtower	UI confirmation + device visible in fleet
2	Operator pairs with Edge via Watchtower invite	E2EE pairing established; visible in Watchtower
3	Operator sends controller program package to edge	Encrypted transfer completes; file staged on Edge
4	Operator sends HMI program package to edge	Encrypted transfer completes
5	Access policy enforced (no forward, revoke works)	Negative test: forward blocked; revoke: access withdrawn
6	Engineer opens RDP session to edge PC via Privitty	Full desktop accessible; bandwidth adequate
7	Program applied to controller/HMI using engineering tools	Field device updated and running new program
8	Remote-HMI client used to verify HMI	Remote HMI access confirmed
9	OPC UA tag data visible in Watchtower Telemetry	Live values shown; threshold alarm fires on test
10	Relay confirmed transport-only	Audit log reviewed; revoke tested

Indicative timeline

Week	Activity
1–2	Staging relay + Watchtower; edge install on Windows IoT Enterprise
3	Pairing, file transfer, access control validation
4	RDP/VNC tunnel + remote-HMI client + program apply workflow
5	OPC UA telemetry + Watchtower alarm rules validation
6	Security review, documentation, stakeholder demo

Implementation Roadmap

Phase 1 -- Proof of concept

- Partner-private relay (staging environment)
- Dedicated Watchtower for the partner organisation
- Privitty Edge on Linux industrial gateway
- Branded Privitty App (beta release) for Android
- Remote program transfer, RDP access, OPC UA telemetry validated

Phase 2 -- Pilot (field trial)

- Multi-site Edge onboarding with role-based operator management
- Automated program ingest via JSON-RPC on the Edge (for MES/SCADA integration)
- Hardened Edge installer for partner gateway hardware image
- Customer administrator training and SOPs

Phase 3 -- Production release

- Production relay cluster (high availability, partner-operated)
- Production Watchtower with optional partner SSO integration
- Windows Edge service for Windows IoT Enterprise edge PCs (roadmap)
- Desktop and iOS Privitty App distribution (roadmap)
- Security certification package and deployment playbooks

Appendix A -- Event Flow Summary

Phase	Event	Initiator	Result

Provisioning	Administrator registers Edge in Watchtower	Administrator	Edge visible in fleet; awaiting activation
Provisioning	Administrator invites operator by email	Administrator	Operator receives licence + pairing QR
Activation	Operator imports licence in Privitty App	Operator	App activated; device registered
Pairing	Operator scans Edge QR in App	Operator	SecureJoin completes; verified E2EE channel
Pairing	Administrator sets access window	Administrator	Time-bounded session policy applied
Discovery	Edge publishes capabilities (RDP/VNC/file)	Edge	Operator UI shows available actions
Transfer	Operator sends program package to Edge	Operator	E2EE package delivered; decrypted and staged on Edge
Access	Operator opens RDP/VNC tunnel	Operator	E2EE session to edge PC or HMI
Apply	Engineer applies program via vendor tools	Engineer	Controller/HMI updated
Telemetry	Edge pushes OPC UA data batch	Edge	Watchtower Telemetry UI updated; alarms evaluated
Governance	Watchtower records all session and transfer events	System	Audit log entry created
Teardown	Operator revokes file access	Operator / Admin	Access withdrawn immediately
Teardown	Tunnel closed; access window expires	Operator / System	Session ended; audit entry completed

Appendix B -- Component Reference

Name	Role	Typical deployment
Privitty Edge	Factory gateway daemon	Windows service on the edge PC
Privitty App	Operator client	Partner-branded mobile / desktop
Privitty Relay	Encrypted transport	Partner-private cluster
Privitty Watchtower	Fleet management web UI	Partner dedicated instance
Industrial HMI	Human-machine interface	Factory LAN
Programmable controller	Process control	Factory LAN

Summary

Privitty enables industrial organisations to give remote engineers and operators the access they need -- without opening factory networks, without storing production IP in the cloud, and without sacrificing the auditability and governance that OT environments demand.

Three components work together: Privitty Edge on the factory floor, Privitty App on the operator's device, and Privitty Watchtower in the administrator's browser. No VPN, no inbound firewall ports, no uncontrolled file transfers.

The recommended first step is a proof of concept on the partner's gateway hardware to validate program transfer, RDP/VNC engineering access, OPC UA telemetry, and Watchtower audit visibility -- followed by a multi-site field pilot.

“Machine data flows freely -- access never does”