CHAPTER 01

Qlekk White Paper

Communication Built Around People. Privacy Engineered Into the System.

Version 1.0 | English - Canonical Edition

Qlekk is a communication platform being developed around a central principle: a communication service should process only the information it genuinely needs to provide the service safely and effectively.

The platform brings together private messaging, voice and video communication, meeting functionality, temporary file exchange, multi-device access and multilingual interfaces.

> **What is the minimum information required to provide a useful, reliable and secure communication service?**

Four guiding principles

Privacy by Design
Data-protection considerations are incorporated into product and engineering decisions rather than added only after development.

Data Minimisation
Data should be processed only when there is a defined operational reason for doing so.

Purpose Limitation
Information should be connected to a specific and understandable purpose rather than accumulated for undefined future use.

Storage Limitation
Information that no longer needs to exist should not automatically become permanent historical data.

These principles are aligned with the direction of the European data-protection framework, including GDPR concepts such as purpose limitation, data minimisation, storage limitation, integrity and confidentiality, and accountability.

Qlekk does not claim absolute privacy, zero risk, regulatory certification or governmental approval. Its objective is more practical: reduce unnecessary data exposure, maintain clear technical boundaries, provide meaningful user control and make privacy part of the engineering process.

CHAPTER 02

The Communication Problem

Digital communication has become essential infrastructure for everyday personal and professional life. Messages, calls, meetings, documents and cross-device access are now expected to work quickly and continuously.

But convenience has also made communication systems increasingly data-heavy. A single interaction can involve far more than the visible message itself: account identifiers, session state, device information, contact relationships, timestamps, notifications, attachments, network information and operational logs.

> **Every additional category of retained information creates another responsibility: it must have a purpose, be protected, have controlled access and eventually reach the end of a defined lifecycle.**

Four forms of exposure

Security Exposure

Information that remains stored over time can increase the impact of unauthorised access, system compromise or operational mistakes.

Privacy Exposure

Data collected for one function can become vulnerable to secondary use if the system does not maintain clear purpose and access boundaries.

Trust Exposure

Users need understandable answers about what information exists, why it exists, who can access it and how long it remains.

Operational Exposure

Every additional dataset creates obligations around storage, security, backup, deletion, documentation and incident response.

A different starting point

Qlekk does not begin from the assumption that more data automatically creates a better communication service. Each significant category of processing should be able to answer one basic question: why is this data necessary for the function the user actually requested?

This approach reflects the logic of data minimisation and storage limitation: collect what is necessary for a defined purpose, avoid unnecessary duplication, and avoid retaining identifiable information indefinitely without a justified reason.

CHAPTER 03

Development Method

Qlekk has been developed incrementally rather than as a monolithic application. The objective has been to stabilise each important layer before expanding into the next one.

Development began with identity, authentication and session control. Once those foundations were tested, the platform expanded into contacts, private messaging, attachments, voice and video calls, notifications, meetings and multilingual interfaces.

> **Build narrowly. Test explicitly. Stabilise. Record. Then expand.**

A progressive development sequence

01 **Identity & Sessions**
Accounts, authentication, password recovery, device visibility and session revocation established the security foundation.

02 **Contacts & Conversations**
User discovery, contacts and conversation identity were added after the account layer was stable.

03 **Messaging & Temporary Files**
Private messaging and attachment handling introduced controlled lifecycles for communication content.

04 **Voice, Video & Notifications**
Real-time communication and notification delivery were added as separate operational layers.

05 **Meeting Rooms**
Meeting functionality extended the platform toward multi-party communication and is still undergoing broader validation.

06 **Internationalisation**
Multilingual interface support was introduced after the communication foundation had already been established.

Why this matters

A new capability is not considered successful merely because the new capability itself works. It must also preserve previously established security, privacy and operational guarantees.

Qlekk uses regression testing, stable checkpoints, backups and rollback procedures throughout development. Every important change should improve the platform without silently weakening what was already proven to work.

CHAPTER 04

Product Architecture

Qlekk Connect is the primary communication layer within the Qlekk platform. It is being developed as a modular system in which identity, messaging, real-time communication, meetings, language-interface and notification services can evolve independently without becoming unnecessarily dependent on one another.

This separation reduces the risk that a failure in one component becomes a failure across the entire communication experience.

> **A communication feature should have a clear responsibility, a clear data boundary and a clear failure path.**

Implemented & tested

Identity & Authentication
Account creation, login, password management and authenticated sessions.

Devices & Sessions
Session visibility, cross-device continuity and session revocation.

User Discovery & Contacts
Registered-user search, contacts and conversation identity.

Private Messaging
Person-to-person communication with controlled message lifecycles.

Images & Files
Temporary attachment handling with defined types, limits and validation.

Voice & Video Calls
Real-time one-to-one voice and video communication.

Notifications
Incoming communication alerts and background notification support.

Multilingual Interface
Language preference, broad interface language coverage and RTL support.

PWA Experience
Installable browser-based experience across supported devices.

Under active validation

Meeting Room
Core meeting controls and return-to-meeting behaviour have been tested, while broader multi-participant and network-resilience validation remains open.

Invitation Journey
Native invite sharing is implemented; the full invite-to-registration-to-communication journey remains part of final validation.

Roadmap

Additional business communication layers · **Advanced privacy-protection features**

A notification service should not become a second message archive. Meeting functionality should have its own lifecycle and failure handling. Session management should remain a security control rather than only a convenience feature.

This modular structure makes testing, maintenance, rollback and future development more manageable while also providing clearer boundaries around what information each component needs to process.

CHAPTER 05

Privacy by Design

Privacy is not intended to operate in Qlekk as an optional mode that a user must discover and enable after using the product.

The design objective is different: privacy-conscious decisions should exist in the architecture, defaults and development process before a feature reaches the user.

> **Privacy should influence how a feature is designed before it influences how that feature is described in a policy.**

Privacy by Design and by Default

Qlekk is being designed with reference to the GDPR concept of Data Protection by Design and by Default. In practice, technical and operational decisions should consider data protection from the beginning of a feature's lifecycle rather than treating privacy solely as a legal review at the end.

Purpose Limitation
Each significant category of personal data should be connected to a clear and understandable operational purpose.

Data Minimisation
A function should not process more personal information than is reasonably necessary to perform that function.

Storage Limitation
Retention should reflect operational and legal need rather than indefinite storage by default.

Access Limitation
Access to information should be constrained according to the role and responsibility of the component processing it.

Transparency
Users should be able to understand the main reasons information is processed and the controls available to them.

Accountability
Important processing decisions should be explainable, testable and documented rather than existing only as assumptions.

Six engineering questions

01
What information does this feature actually need?

02
Why is that information necessary?

03
Where is the information processed or stored?

04
Who or what can access it?

05
How long should it exist?

06
Can the same function work with less information?

Defaults should favour limited processing, limited retention and clearly bounded access where the service can operate effectively under those constraints.

The safest unnecessary dataset is often the one that was never created. Qlekk does not describe this approach as regulatory certification or proof of perfect compliance; it is an engineering direction designed with reference to European data-protection principles and subject to continued technical and legal review.

CHAPTER 06

European Data Protection Framework

Qlekk is being developed with reference to European data-protection principles.

The General Data Protection Regulation provides a central European framework for personal-data protection and informs Qlekk's privacy and engineering approach where applicable.

> **Regulatory principles should influence architecture, not only documentation.**

Core data-protection principles

Lawfulness, Fairness & Transparency
Personal-data processing should have an appropriate legal basis and should be understandable to the people affected by it.

Purpose Limitation
Information should be collected for defined purposes rather than for unrestricted or undefined future use.

Data Minimisation
Personal data should be adequate, relevant and limited to what is necessary for the intended purpose.

Storage Limitation
Identifiable information should not be retained longer than justified by operational or legal need.

Integrity & Confidentiality
Appropriate technical and organisational measures should protect personal data against unauthorised access, loss or misuse.

Accountability
Organisations should be able to demonstrate how data-protection principles are implemented in practice.

What this means for engineering

01 **Define the purpose**
Understand why a category of personal data is required.

02 **Limit the processing**
Use only the information reasonably necessary for that purpose.

03 **Protect the information**
Apply appropriate access, security and operational controls.

04 **Define the lifecycle**
Determine retention, deletion and review requirements.

05 **Document and verify**
Be able to explain and test how the controls operate.

No certification claim

Qlekk does not claim GDPR certification, EDPB approval or governmental certification. References to European regulatory principles describe the framework informing Qlekk's engineering and compliance work.

Primary regulatory references

- Regulation (EU) 2016/679 - General Data Protection Regulation.
- European Data Protection Board - Guidelines 4/2019 on Article 25.

CHAPTER 07

Security & Operations

Privacy depends on security, but security is not a single feature. Qlekk is being developed and tested using multiple technical and operational controls that work together across accounts, sessions, files, infrastructure and release management.

> **A secure feature is not only one that works correctly - it is one that fails predictably, limits exposure and can be verified.**

Security controls

Authentication & Sessions

Protected operations require authenticated access, while sessions can be validated and revoked when necessary.

Rate Limiting

Sensitive operations such as login and credential recovery are subject to rate controls intended to reduce automated abuse.

Attachment Validation

File handling uses defined types, size limits and validation rather than relying solely on filenames or extensions.

Access Boundaries

Sensitive operational storage is intended to remain separated from publicly reachable web paths and unauthenticated access.

Environment Separation

Development and validation are performed separately from production, with production deployment treated as a controlled release activity.

Regression Verification

Significant changes are tested against previously working behaviour so that new functionality does not silently weaken existing controls.

Operational discipline

01 **Create a known stable state**
Important milestones are recorded before further changes are introduced.

02 **Back up before change**
A recoverable copy should exist before modifying sensitive components.

03 **Test the new behaviour**
Verify both the new capability and the controls surrounding it.

04 **Run regression checks**
Confirm that previously established behaviour still operates correctly.

05 **Preserve rollback capability**
Maintain a clear path back to a known stable version if validation fails.

Experimental development should not silently alter the production environment. Production deployment is intended to use environment-specific configuration, storage, credentials and routing, together with backup, rollback readiness and post-deployment verification.

Qlekk does not claim that any software system can be perfectly secure. The objective is to reduce avoidable risk through layered controls, testing, limited exposure, documentation and continuous review.

CHAPTER 08

User Control & Data Lifecycle

Privacy is not only about reducing what a system collects. It is also about giving users meaningful control over access, sessions, devices and the lifecycle of information associated with their communication.

> **Data should have a reason to exist, a controlled period of use and a defined end to its lifecycle.**

User control

Session Visibility
Users should be able to understand where their account is currently active and distinguish the current device from other sessions.

Session Revocation
Access from another device should be revocable when the user no longer trusts or needs that session.

Credential Control
Password changes and recovery procedures should support the ability to invalidate existing access when appropriate.

Account Lifecycle
Account deletion and related data-lifecycle procedures form part of the product's planned release-readiness work.

Data lifecycle

01 **Purpose is defined**
A category of information should have a clear reason for being processed.

02 **Processing is limited**
The system should use only what is reasonably necessary for that purpose.

03 **Access is constrained**
Access should follow the responsibility of the component or user involved.

04 **Retention is bounded**
Information should not automatically remain identifiable forever.

05 **The lifecycle ends**
Data should be deleted, expired or otherwise removed when its purpose and applicable retention requirements have ended.

Content, metadata and temporary retention

Communication systems often require metadata to operate, but metadata and message content do not necessarily need to follow the same lifecycle. Qlekk's design direction is to avoid turning operational metadata into a duplicate content archive.

Different data types may require different retention periods for technical, security, legal or user-experience reasons. The principle is not 'delete everything immediately'; it is to avoid indefinite retention where there is no justified reason for it.

> **Retention should be a decision - not an accidental consequence of storage.**

CHAPTER 09

Accountability Model

Privacy and security controls are most useful when they can be explained, tested and verified rather than existing only as assumptions.

Qlekk treats accountability as an engineering discipline: important changes should leave behind enough evidence to understand what changed, why it changed and whether existing protections still work.

> **A system should be able to explain its important changes, not merely contain them.**

Stable checkpoints
Known working states provide a reference before further development.

Backups
Recoverable copies reduce the risk that a failed change becomes irreversible.

Regression results
Repeated checks help show whether previously working behaviour remains intact.

Version & hash verification
Recorded versions and file hashes can help verify that the intended build is the build being reviewed or deployed.

Change records
Significant decisions should be documented so that future development does not depend on reconstructing past reasoning from memory.

Rollback paths
A release process should preserve a practical route back to a known stable state when validation fails.

Accountability across the lifecycle

01 **Before development**
Define the purpose, expected data use and security boundaries.

02 **During development**
Keep changes narrow enough to test and understand.

03 **Before release**
Run verification and regression checks against known stable behaviour.

04 **At deployment**
Confirm environment, version, configuration and rollback readiness.

05 **After release**
Observe the system and retain enough evidence to investigate unexpected behaviour.

Nine questions for significant changes

01
What changed?

02
What information does the change process?

03
Why is that information necessary?

04
Where is it processed or stored?

05
Who or what can access it?

06
How long should it remain?

07
How was the change tested?

08
Did existing controls regress?

09
Can the change be rolled back safely?

Accountability is not intended to slow development. Its purpose is to make faster development safer by reducing ambiguity about what the system is doing and what changed between stable states.

CHAPTER 10

Roadmap & Commitment

Qlekk remains an actively developed communication platform. The objective is not to move from development to public release as quickly as possible, but to reach release through a controlled sequence of validation, regression and operational preparation.

> **Production is not the next development step. It is the result of successfully closing the development steps before it.**

Current closure path

01 Complete Meeting Room validation
Broader multi-participant, network-resilience and real-device testing remains part of final meeting validation.

02 Complete the invitation journey
Validate the full path from invite sharing through registration, login and communication.

03 Run final security & failure tests
Re-test authentication, rate controls, abnormal requests, files, network interruption and important protected endpoints.

04 Real-device interface regression
Verify supported interface languages, RTL behaviour, PWA installation and cross-device behaviour.

05 Legal & account-lifecycle work
Finalise privacy documentation, terms and account-deletion procedures before broad release.

06 Run full-system regression
Re-test accounts, sessions, contacts, messaging, files, calls, notifications, meetings and PWA behaviour together.

07 Release candidate stable state
Record the intended release with verified backups, versions, hashes and rollback procedures.

08 Prepare production independently
Production should use its own configuration, credentials, storage, routing and deployment controls.

09 Deploy gradually and verify
Use controlled deployment, live smoke testing, version verification and rollback readiness.

What Qlekk does not promise

No zero-risk promise
No modern communication system can responsibly guarantee that all security or privacy risk can be eliminated.

No certification-by-declaration
Regulatory principles referenced in this paper do not constitute certification, approval or governmental endorsement.

No permanent-feature assumption
Product capabilities may evolve when testing, security, legal requirements or user needs justify a change.

No data-maximisation objective
Future product growth should not depend on collecting additional personal information without a defined need.

The Qlekk commitment

Qlekk's commitment is not to claim perfection. It is to maintain a development discipline that makes privacy, security and accountability part of how communication technology is designed and released.

Collect less where possible. **Retain less where possible.** **Keep responsibilities separated.** **Use privacy-conscious defaults.**
Test important changes. **Document what the system does.** **Preserve a safe path backward.**

QLEKK

Communication Built Around People.

Privacy Engineered Into the System.