GLACIER · TECHNICAL WHITEPAPER

# Privacy, security, and transparency on iOS.

How we design, operate, and explain privacy and security in Glacier for iOS.

| EDITION | VERSION | PUBLISHED | SOURCE BASIS |
| --- | --- | --- | --- |
| Customer | 1.3 | 21 October 2026 | `800afba · 04befc6` |

Published by Glacier for customers, partners, security teams, and privacy reviewers

OUR COMMITMENT

### Privacy protection built into the product

We designed Glacier to reduce the personal information exposed during everyday browsing, DNS resolution, account operation, and calling. This paper shows the implemented controls behind that design, the privacy commitments we publish, and the providers that operate specific service boundaries.

EXECUTIVE SUMMARY

# Privacy by design, implemented in layers

Glacier for iOS combines a WireGuard virtual private network, DNS-over-TLS filtering, on-device security checks, encrypted local storage, and an optional Glacier phone number. Together, these controls reduce routine exposure to access networks, DNS observers, tracker and malicious domains, device misconfiguration, and disclosure of a personal phone number.

Data minimization runs through the design. We request permissions when a feature needs them, keep the detailed Security Center assessment on the device, encrypt our local application database, store keys and credentials in the iOS Keychain, avoid advertising trackers, and use pseudonymous account identifiers for managed DNS and calling resources.

## Four protection layers

01

### VPN protection

We use WireGuard to encrypt IP traffic between your iPhone and the selected Glacier VPN gateway and replace your access-network IP with the gateway address seen by ordinary destinations.

02

### Protected DNS

We use DNS-over-TLS for name resolution, apply security and tracker-blocking policy, and provision provider profiles to discard source IP and queried-domain values while retaining aggregate protection counts.

03

### Security posture

We check selected device and app conditions—including OS and app version signals, screen lock, biometrics, proxy use, and compromise indicators—and present a readable status.

04

### Private calling

An optional Glacier number separates your personal number from supported calls. Current provisioning uses a random account UUID instead of email in routine Twilio identity fields, with controlled migration tooling for earlier email-keyed resources.

VERSION 1.2 SECURITY REVIEW UPDATE

### Stronger account, network, and service boundaries

Following a targeted security review, we hardened iOS account separation and network-profile cleanup; added stricter entitlement, device-ownership, and provider-resource checks; shortened VPN configuration delivery; strengthened account-lifecycle revocation; restricted sensitive operational routes; and minimized support diagnostics by design.

01 ABOUT THIS PAPER

# How we substantiate our claims

We prepared this paper from Glacier iOS source revision `800afba` and reviewed Console backend source revision `04befc6` as of 29 August 2026, together with our public Privacy and Security Policies and relevant protocol, platform, and provider documentation. Source establishes implemented controls; policy establishes our operational commitments for production data handling.

We label the evidence behind each material statement so customers can distinguish an implemented control from a policy commitment or a provider-documented transport behavior.

| EVIDENCE SOURCE | WHAT IT SUPPORTS |
|---|---|
| Our iOS implementation | Permissions, local-data protection, Keychain use, Network Extension configuration, DNS health checks, security posture, and selected HTTPS pinning. |
| Our reviewed backend implementation | Pseudonymous provider IDs, entitlement and ownership gates, VPN/DNS lifecycle controls, authenticated service flows, restricted administrative routes, privacy-safe diagnostics, and aggregate privacy statistics. |
| Our published policies | Our commitments for VPN/DNS activity data, advertising tracking, data sale or sharing, support data, calling data, and privacy requests. |
| Platform, protocol, or provider documentation | The properties and boundaries of iOS security services, WireGuard, DNS-over-TLS, Twilio transport, and managed DNS settings. |

**Table 1.** Evidence sources used throughout this paper.

### Scope and currency

- This paper covers iOS behavior, permissions, local storage, authentication, DNS and VPN configuration, calling architecture, relevant backend privacy controls, published policies, and vulnerability reporting.
- This edition incorporates security-review-driven changes merged into the cited revisions. Source review verifies implementation, not rollout state, credential rotation, completion of every legacy-provider migration, or an external certification of live infrastructure.
- Use the latest App Store release and current Privacy Policy for current product and processing terms.

02 ARCHITECTURE AND TRUST MODEL

## A layered architecture limits unnecessary exposure

Glacier separates VPN, DNS, authenticated app APIs, calling, and local storage into purpose-specific data paths. This lets each feature receive the protection it needs without sending every category of customer data to every provider. The iPhone keeps local keys and encrypted records; the VPN gateway handles tunneled traffic; the DNS service handles protected name resolution; and Twilio handles calling only when that feature is used.

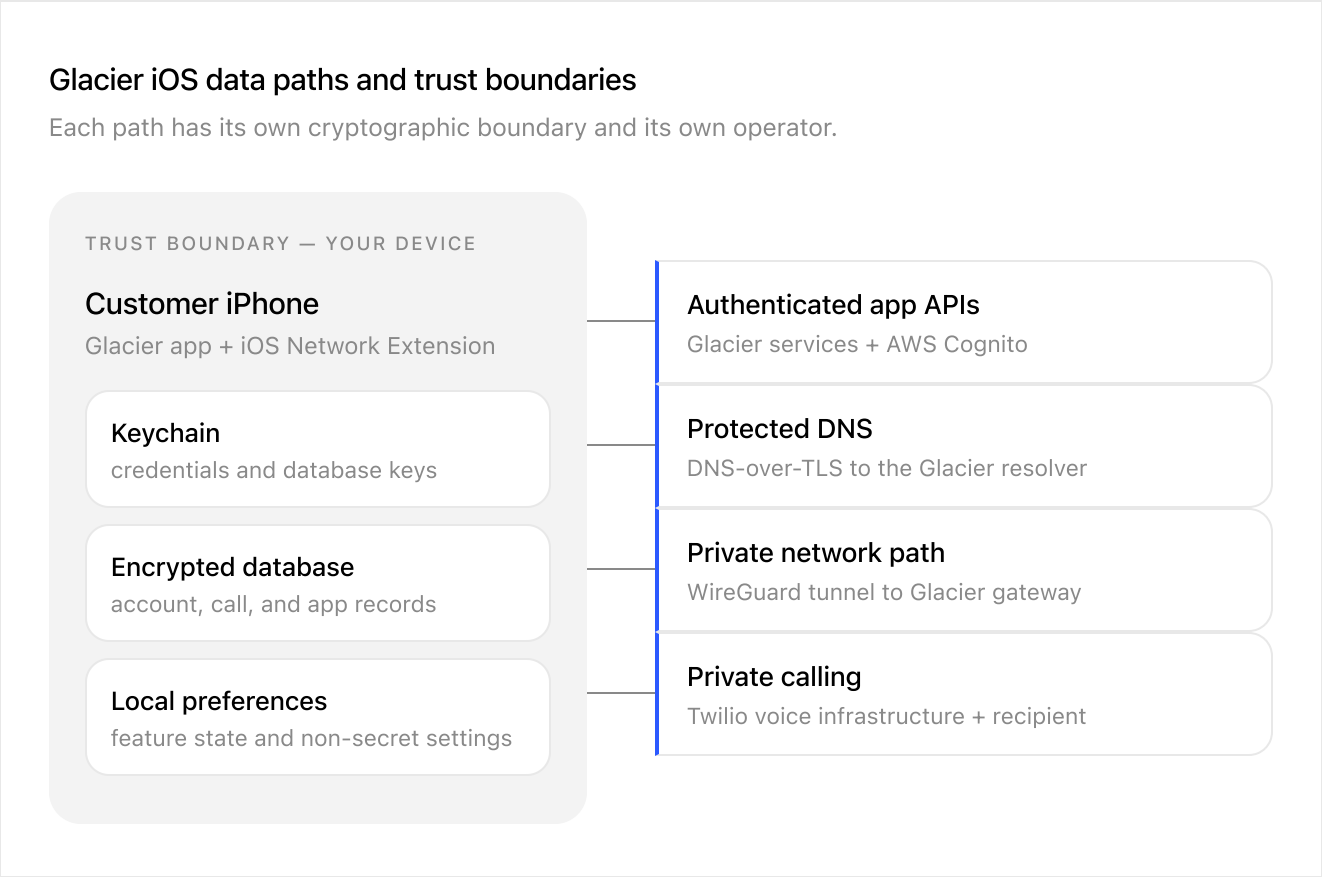



**Figure 1.** Our protection layers cross distinct trust boundaries.

| PATH | IMPLEMENTED PROTECTION | PRIMARY ENDPOINT | PRIVACY RESULT |
|---|---|---|---|
| VPN traffic | WireGuard tunnel encryption | Glacier VPN gateway | Local networks see an encrypted tunnel; ordinary destinations see the gateway IP |
| DNS requests | DNS-over-TLS plus provider-side activity minimization | Glacier DNS service | Queries are encrypted in transit; source IP and domain fields are dropped by the reviewed profile default |
| App and account APIs | HTTPS; pinning on core managers and selected domains | Glacier services and AWS Cognito | Authenticated service access with reduced persistent HTTP caching |
| Voice calling | Glacier number, UUID account identity, TLS signalling, and SRTP media | Twilio voice infrastructure | Personal number and email receive less routine provider exposure; legacy identity migration is controlled separately |
| Local app records | SQLCipher, iOS Data Protection, backup exclusion, and Keychain-held keys | Your iPhone | Local records remain encrypted at rest when keys are unavailable |

**Table 2.** Data paths, their implemented protection, and the privacy result for each.

03 PROTECTION GOALS

# The everyday exposures Glacier is built to reduce

Glacier is a consumer privacy and security application designed around practical, recurring risks: untrusted Wi-Fi, routine DNS observation, known tracker or malicious domains, device-security misconfiguration, unnecessary provider identifiers, and disclosure of a personal phone number.

| PRIVACY OR SECURITY GOAL | GLACIER CONTROL | PROTECTION SCOPE |
|---|---|---|
| Reduce local-network visibility | WireGuard encrypts traffic to the VPN gateway; HTTPS continues protection to participating destinations. | Protects the path from the iPhone to the gateway and masks the access-network IP from ordinary destinations. |
| Minimize DNS activity exposure | DNS-over-TLS encrypts queries; the reviewed backend default drops source IP and domain values from provider logs. | Supports aggregate counts without retaining itemized DNS activity under Glacier policy. |
| Block known trackers and threats | Managed resolver policy enables threat intelligence, Safe Browsing, cryptojacking, homograph, typosquatting, DGA, CSAM, and recommended privacy blocklists. | Blocks domains that match the active classification policy. |
| Surface device-security gaps | Security Center reports version, screen-lock, biometric, proxy, protection-state, and compromise signals. | Gives the customer actionable on-device indicators within the iOS sandbox. |
| Reduce calling-identity exposure | A Glacier number replaces the personal number for supported calls; provider resources use an account UUID rather than email, with controlled legacy migration tooling. | Reduces routine exposure to recipients and selected Twilio resource fields. |
| Protect local records | SQLCipher encryption, a random Keychain-held key, iOS file protection, and backup exclusion. | Adds layered at-rest protection to the application database. |

**Table 3.** Each protection goal, the control that addresses it, and the scope of that protection.

## Where separate safeguards still matter

Glacier complements, rather than replaces, current iOS, a strong device passcode, HTTPS, secure account credentials, and careful handling of links and content. A compromised unlocked device or a destination that receives information directly remains outside what a VPN, DNS filter, or caller-number substitution can prevent.

# WireGuard VPN: encrypted transport to the gateway

We use Apple's Network Extension framework and WireGuard to create an IP tunnel from your iPhone to a selected Glacier gateway. WireGuard's protocol uses modern cryptographic constructions, including ChaCha20-Poly1305 for authenticated encryption and Curve25519 for key agreement. [S5]

## How we implement the VPN

- We create and manage the tunnel through iOS Network Extension APIs, including on-demand behavior and region selection.
- Our service checks active entitlement on every profile request. It supports authenticated direct configuration delivery and keeps the compatibility download link to a 120-second lifetime with no-store response handling.
- You can activate, deactivate, and replace VPN profiles. On sign-out or account deletion, iOS turns off and removes Glacier tunnels and related cached network state.
- We coordinate DNS-over-TLS so DNS protection can remain available when trusted-network rules suppress the VPN.
- Backend account-lifecycle processing revokes WireGuard peers across claimed and prior core assignments and deletes related profile objects before destructive account finalization succeeds.

## What the VPN changes

Between your iPhone and our gateway, the local Wi-Fi operator or Internet access provider sees an encrypted tunnel rather than the contents and individual destinations visible inside that tunnel. The public-facing source address seen by ordinary destinations is the gateway address, not your immediate access-network address.

## Protection from gateway to destination

Traffic continues from the Glacier gateway to its destination, where HTTPS and the destination's privacy practices apply. The VPN prevents routine access-network inspection and IP exposure; application-layer encryption protects content beyond the gateway.

OUR PRIVACY COMMITMENT

### No VPN activity history

Our Privacy Policy states that we do not store, retain, or archive VPN activity logs, including destinations, source IP addresses, timestamps, bandwidth, or session records. We retain no browsing history for advertising or sale. [S1]

05 DNS PRIVACY AND FILTERING

# DNS-over-TLS: protected name resolution

DNS converts service names into network addresses. Traditional DNS can be visible to the local network or resolver. We configure an iOS DNS-over-TLS profile so queries to our selected resolver travel inside TLS, consistent with the transport defined by RFC 7858. [S6]

## Pseudonymous setup and health verification

The iOS profile must be installed and selected for protection to be active. Our app retrieves and applies the profile, then checks a randomized subdomain and expects a known response before treating DNS protection as healthy. It validates the DNS-over-TLS scheme and host before saving a configuration.

DNS selection is account-bound. The app uses the last successfully fetched profile for the current account or no profile; it does not silently persist the shared backup profile as that account's configuration. A bounded repair process can replace an active shared-backup profile with the signed-in account's profile, verifies the result, and reverts if verification fails. Sign-out clears the account-scoped cache and installed DNS profile.

The backend assigns a random UUIDv4 account ID for provider-facing profile names and issues a signed profile only for a device record owned by the authenticated account. Controlled migration tooling can rename legacy email-named DNS profiles in place, preserving their settings while removing the email from the provider-side name.

## Activity minimization, blocking, and aggregate reporting

The reviewed backend provisions managed DNS profiles with security and privacy filtering enabled. Provider logging remains available for aggregate statistics, while the default profile configuration instructs the provider to drop both the source IP and queried domain. Glacier retrieves totals such as

all queries and blocked queries—not a customer browsing history—from those profiles.

Our Privacy Policy makes the operational commitment explicit: we do not store, retain, or archive DNS activity logs, including query content, resolver activity, IP addresses, timestamps, or other personally identifiable network activity. The permitted exception is numeric totals used to display protection statistics; those totals contain no domains, destinations, or timestamps. [S1]

### DNS protection scope

DNS-over-TLS protects the resolver exchange and the active policy blocks classified domains. Full website traffic still relies on the VPN and HTTPS, and an iOS profile must be active for DNS protection to apply. Glacier checks profile health rather than treating installation alone as proof of protection.

06 DEVICE SECURITY POSTURE

## Security Center: actionable visibility on the device

Our Security Center turns selected iOS conditions into a practical posture view, helping you see whether key safeguards appear active and where attention may be needed. It checks signals available to an iOS application and presents them in customer-readable form.

| SIGNAL | WHAT WE CHECK | CUSTOMER VALUE |
|---|---|---|
| Device compromise | Selected jailbreak, rootless jailbreak, TrollStore, reverse-engineering, debugger, and simulator indicators. | Surfaces conditions that can weaken the iOS security model. |
| Network proxy | Whether a system proxy appears configured or active. | Makes an unexpected traffic intermediary visible for review. |
| Screen lock | Whether device passcode protection is available. | Prompts a foundational protection for an unattended device. |
| Biometrics | Whether biometric authentication is configured and available. | Shows whether biometric convenience and protection are available. |
| iOS and app version | Current version data compared with available release information. | Helps identify missing security and product updates. |
| Core protection | VPN and DNS state used by the home and widget experiences. | Keeps essential privacy controls visible at a glance. |

Table 4. Signals evaluated by Security Center and the value each one gives the customer.

### Where posture data lives

In the app version described here, we build a structured security-status object containing device model, iOS and app versions, update flags, compromise detail, screen-lock and biometric state, and core-protection state. We keep that detailed object on the device and share selected status locally with the widget; we do not upload the detailed object in this implementation. We will update this paper if that data flow changes.

HOW TO INTERPRET THE RESULT

### A useful posture signal, not forensic attestation

An “all clear” result means none of the selected checks fired at that moment. Because iOS apps operate inside a sandbox, keep iOS current and continue using a strong passcode, secure account credentials, and careful handling of links.

07 APPLICATION AND LOCAL-DATA SECURITY

## Defense in depth on the device

### Encrypted local database

Our local application database uses SQLCipher through GRDB. We generate 48 bytes of cryptographically secure random key material with Apple’s Security framework, store it in the iOS Keychain, apply it to SQLCipher, mark database files with iOS file protection, and exclude the database directory from device backups. Our app entitlement also declares complete iOS Data Protection as the default protection class. [S4]

Together, these controls protect copied storage when the database key remains unavailable. Like other at-rest protections, decrypted data can be available while the app is legitimately running on an unlocked device.

### Credentials and secrets

We use AWS Amplify and Amazon Cognito for account authentication. The current iOS configuration uses Cognito user-pool tokens without an unused identity-pool dependency, and a one-time migration refreshes previously stored credentials into that user-pool-only form. We configure

device-only, after-first-unlock Keychain accessibility for Keychain items, including a pending signup password, and remove that pending secret after login completes or the user returns to manual sign-in.

Ordinary password sign-in requests Glacier's emailed custom-code challenge where that backend flow is enabled. A narrow compatibility fallback to Cognito's standard password flow remains while deployment is completed, so this source review does not describe emailed codes as mandatory for every iOS sign-in. Social sign-in uses Apple or Google through the hosted authentication flow. Account and entitlement calls use authenticated requests.

### Transport security and pinning

We implement CA public-key pinning that first performs normal TLS certificate and hostname validation, then requires a matching pinned public key in the certificate chain. Our core Security Center and calling-backend managers use this pinned, ephemeral HTTPS configuration, and iOS App Transport Security pins selected authentication domains.

Pinning covers the core Security Center and calling-backend managers plus selected authentication domains. The separate WireGuard profile path uses standard TLS validation through an ephemeral session rather than the shared pinning manager.

### Privacy manifest and tracking

Our Apple privacy manifest declares tracking disabled, no tracking domains, and no collected-data types in the manifest; it also declares approved reasons for UserDefaults access. Our Privacy Policy separately states that we do not use advertising trackers and do not sell customer data. [S1]

The manifest describes the app's declared tracking posture. Account, subscription, calling, push, and support data are processed only where their corresponding features require them, as described below.

08 SECURITY REVIEW IMPROVEMENTS

## Defense strengthened across iOS and the service plane

Version 1.2 incorporates controls merged after a targeted security review. These changes reinforce account separation, reduce the useful lifetime of sensitive material, make authorization checks explicit at service boundaries, and limit operational access and diagnostic data.

| AREA | IMPLEMENTED IMPROVEMENT | SECURITY RESULT |
|---|---|---|
| Account separation on iOS | User-pool-only credential state; account-scoped DNS cache; full VPN and DNS removal on sign-out or account deletion. | Reduces the chance that network or authentication state carries into another signed-in account. |
| VPN authorization and revocation | Active entitlement is checked on every profile request; profile downloads expire after 120 seconds; account lifecycle revokes peers and removes stored profiles. | Narrows unauthorized reuse and closes access when an account is disabled or deleted. |
| DNS ownership and recovery | Signed profiles require an account-owned device; iOS fails closed to the current account's last valid profile and verifies or reverts automated repair. | Prevents cross-account or caller-supplied profile assignment and avoids persisting a shared fallback. |
| Calling authorization | Active entitlement is required for capability tokens and number search; caller ID and number release are checked against account ownership; Twilio webhooks are signature-validated. | Keeps calling capabilities and provider resources bound to the authenticated account. |
| API boundaries | Explicit input and response allowlists, generic external server errors, fail-closed Apple receipt verification, and enumeration-resistant sign-in discovery. | Reduces injection, excess-data disclosure, and account-discovery risk. |
| Operational infrastructure | Sensitive routes are tailnet-gated, public SSH ingress is removed, root volumes are encrypted, deploys use short-lived GitHub OIDC roles, and production images are pinned. | Reduces public administrative exposure and standing deployment credentials. |
| Transport policy | Public-cleartext Semaphore endpoints fail at boot outside private or local ranges; Console responses apply restrictive CORS and browser security headers. | Prevents accidental credential delivery to a public HTTP endpoint and strengthens browser-facing controls. |
| Support diagnostics | An admin-only, account-keyed outcome timeline uses a strict metadata allowlist and 180-day expiry; it excludes direct identifiers, content, credentials, network activity, and raw errors. | Supports incident diagnosis without building a browsing, communications-content, or secrets log. |

**Table 5.** Improvements merged after the August 2026 security review.

09 DATA TRANSPARENCY

# What data moves—and why

We believe data minimization is meaningful only when we name the data that remains. The table below combines our iOS implementation with our published policy. “Local” means our app uses the information on your iPhone; it does not mean the same information can never reach a service when you choose an action such as placing a call.

| DATA CATEGORY | PURPOSE | WHERE IT IS PROCESSED | PRIVACY DESIGN |
|---|---|---|---|
| Account identity | Create and authenticate an account; recover access; deliver service. | Glacier and AWS Cognito; limited local account state. | Provider-facing resources use a random account UUID where the reviewed backend supports it. |
| Subscription and purchase | Establish entitlement, restore purchases, reconcile billing. | Apple App Store and/or Stripe; Glacier entitlement services. | Our app can attach the signed-in Cognito account UUID to an App Store purchase token. |
| VPN configuration | Provision a selected gateway and operate the tunnel. | Glacier profile service, iOS Network Extension, VPN gateway. | Our policy states that we do not keep VPN activity logs. |
| DNS profile and totals | Route encrypted DNS, apply blocking policy, show aggregate protection activity. | iOS DNS settings and Glacier resolver/service. | Profile defaults drop source IP and domain values; our policy permits only aggregate numeric totals. |
| Calling identity and metadata | Provision numbers, place/receive calls, show history, deliver voicemail. | Glacier, Twilio, telephone network, local encrypted database. | A Glacier number protects the personal number; current resources use the account UUID, with controlled legacy migration tooling. |
| Contacts | Show a contact list, match caller numbers, start a chosen call. | Read into an in-memory/local index after permission. | We do not upload your address book; a selected or calling number is processed to complete the call. |
| Device protection status | Show posture and protection state to you. | On-device app and widget in this implementation. | We do not upload the detailed security-status object in this implementation. |
| Push token | Notify our app about calls and service events. | Apple Push Notification service, Twilio/Glacier where relevant, local state. | A device token is a service identifier, not message content. |
| Diagnostic outcomes | Troubleshoot account and delivery events without message or network content. | Restricted Glacier administrative service; 180-day expiry. | Strict allowlists exclude email, phone, IP, DNS names, tokens, content, provider IDs, VPN material, URLs, and raw errors. |

| DATA CATEGORY | PURPOSE | WHERE IT IS PROCESSED | PRIVACY DESIGN |
|---|---|---|---|
| Support and communications | Answer requests and send opted-in service or marketing messages. | Zendesk, MailerLite, and Glacier when you submit or opt in. | Content and contact details depend on what you provide. |

**Table 6.** Every data category the product touches, its purpose, and its privacy design.

10 IOS PERMISSIONS

# Permissions are feature gates, not blanket access

iOS mediates access to protected capabilities. You can deny a permission or revoke it later in Settings; the associated feature may then be unavailable or less useful. Our app presents contact and notification permissions during onboarding and requests other permissions when their features need them. [S7]

| PERMISSION OR CAPABILITY | CUSTOMER-FACING PURPOSE | TRANSPARENCY NOTE |
|---|---|---|
| Contacts | Display contacts, match incoming/outgoing numbers, and start a selected call. | We index the address book on device after permission; we do not upload the full address book. |
| Microphone | Capture audio for voice calls. | Required for two-way calling. Call media is processed through Twilio infrastructure. |
| Notifications / VoIP | Alert you to incoming calls and service events. | We share push tokens with the delivery providers needed for the feature. |
| Location while in use / Wi-Fi info | Read and display the connected Wi-Fi network and support trusted-network behavior. | The permission exists because iOS restricts access to the current SSID; it is not presented as continuous location tracking. |
| Bluetooth | Route call audio to Bluetooth devices. | Applies to call audio accessories. |
| Camera and Photos | Our app declares descriptions for video/photo features and saving or selecting media. | Our current CallKit configuration sets video support off; this edition does not present video calling as an active feature. |
| VPN and DNS settings | Install and operate the packet-tunnel and DNS profiles. | iOS shows system authorization and profile state; a downloaded profile may still require selection. |

**Table 7.** Each permission the app can request, and what it is used for.

## Practical permission control

- Review or revoke permissions under iOS Settings → Apps → Glacier.
- Review the active VPN and DNS configuration under iOS Settings → General → VPN & Device Management.
- Use contact access only if on-device contact matching is valuable; manual dialing remains the privacy-minimizing alternative where supported.

# Purpose-limited providers, minimized identifiers

We use service providers for defined functions such as account authentication, protected DNS, payment processing, support, communications, and calling. These are operational service relationships, not an advertising-data marketplace. Our Privacy Policy states that Glacier does not sell, license, sublicense, or share user-submitted data and that the app contains no ads or trackers. [S1]

| PROVIDER | ROLE | TYPICAL DATA BOUNDARY |
|---|---|---|
| Apple | Distribution, StoreKit subscriptions, iOS security services, and push notifications. | Apple account/purchase identifiers, push token, and platform data under Apple's terms. |
| Amazon Web Services / Cognito | Infrastructure and account authentication. | Account identifiers, authentication events/tokens, and service requests; encrypted root storage and short-lived deploy roles in the reviewed infrastructure. |
| NextDNS | Encrypted resolution, filtering, and aggregate counts. | UUID-named profiles; source IP and queried-domain fields dropped by the reviewed default. |
| Twilio | Calling numbers, signalling, media transport, and voicemail. | Calling data needed for the feature; current provisioning uses UUID identities, with controlled migration tooling for earlier resources. |
| Stripe | Supported non-App-Store payments. | Billing and transaction data under Stripe's payment role. |
| Zendesk | Customer support. | Messages, attachments, contact information, and support history that customers provide. |
| MailerLite | Opted-in communications. | Email address, preferences, and campaign interaction data for that relationship. |

**Table 8.** Providers, the function each performs, and the data boundary that applies.

## Private calling: number separation and minimized identity

A Glacier number reduces direct exposure of your personal number to the call recipient. Twilio documents TLS for signalling and SRTP for media between the iOS app and Twilio infrastructure. [S8] Twilio and the telephone network still process ordinary calls.

Current provisioning uses one random account UUID for Twilio number ownership, access tokens, and client routing.

12 CUSTOMER CONTROL, RETENTION, AND DELETION

# Controls you can exercise

01 **Protection state.** Turn VPN and DNS protection on or off, choose available VPN settings, and review status in the app and widget.

02 **Permission state.** Grant, deny, or revoke iOS access to contacts, microphone, location/Wi-Fi information, Bluetooth, photos, camera, and notifications.

03 **Account state.** Sign out or request account deletion from Settings. iOS removes Glacier VPN and DNS configurations and clears account-scoped caches; backend deletion revokes network access before destructive finalization completes.

04 **Subscription state.** Manage an App Store subscription through Apple. Deleting your Glacier account does not automatically cancel an Apple-managed subscription; our app offers a path to Apple's subscription management.

05 **Privacy requests.** Use the contact method in our current Privacy Policy for access, correction, or service-side removal questions that extend beyond your device or authentication account. [S1]

## Retention is purpose-based in the current public policy

Our Privacy Policy describes retaining service data only as long as needed to operate, secure, support, and comply with legal obligations. We do not publish a single numeric schedule that applies to every category. Calling records and voicemail, support correspondence, purchase records, and security records can have different operational or legal lifecycles.

For a technical deep dive into Glacier Calling, see our **Glacier Calling Technical Whitepaper**.

When you delete your account in the app, we ask Cognito to delete the authentication identity, remove local account and network-profile state, and trigger service cleanup. Backend lifecycle handling revokes WireGuard peers and removes stored VPN and DNS-only profile objects before destructive finalization succeeds. Account deletion does not automatically cancel an Apple-managed subscription and is not a promise of synchronous erasure from every provider. Use our Privacy Policy request channel for comprehensive service-side removal or confirmation. [S1]

OUR COMMITMENT

### We describe deletion as separate operations

We distinguish authentication-account deletion, local-device cleanup, App Store subscription management, and deletion of provider-held records. We do not collapse those separate operations into one promise.

13 CLEAR PROTECTION BOUNDARIES

## Strong privacy, precisely described

Glacier provides concrete privacy and security protections without collapsing different technologies into one promise. The boundaries below are the ones customers need in order to use those protections correctly.

| PROTECTION | WHAT GLACIER PROVIDES | NECESSARY BOUNDARY |
|---|---|---|
| VPN | An encrypted WireGuard tunnel to the Glacier gateway and a gateway IP visible to ordinary destinations. | HTTPS and destination privacy practices continue beyond the gateway. |
| DNS | Encrypted resolution, active filtering, provider-side activity minimization, and aggregate protection totals. | The iOS profile must be active; DNS protection does not encrypt the full website connection. |
| Calling | A Glacier number, reduced personal-number exposure, UUID account identity, and protected app-to-Twilio transport. | Twilio, carriers, and the telephone network remain in the call path; legacy migration completion is operationally separate. |
| Device posture | Readable checks for selected protection and compromise signals. | Signals are heuristic and do not replace current iOS or a strong device passcode. |
| Local data | SQLCipher, iOS Data Protection, backup exclusion, and Keychain-held keys. | Data in legitimate use on an unlocked device can be available to the app. |

**Table 9.** What each protection provides, and the boundary that necessarily remains.

### Configuration and currency

VPN and DNS protection depend on active iOS configurations and can be interrupted by another VPN or DNS tool, captive portal, or managed-network restriction. Glacier shows current status and performs health checks so configuration problems are visible.

This dated review describes implemented design and published policy. Source controls still require verified deployment, configuration, and operations; this paper is not an external certification of live infrastructure. The current Privacy Policy and Terms of Use govern Glacier's current commitments.

14 SECURITY GOVERNANCE

## Vulnerability reporting and supported versions

We maintain a Security Policy for good-faith vulnerability research. Report security issues to us privately, not through a public issue tracker. [S3]

We support the latest App Store release and develop security fixes against our current codebase. Our response targets below are process commitments, not a security certification.

This revision records the principal implementation changes made after our August 2026 security review. We use the control-and-evidence matrix below to distinguish source-verified protections from policy commitments and provider-documented behavior.

| STAGE | PUBLISHED TARGET |
| --- | --- |
| Acknowledgement | Within 3 business days of the initial report. |
| Triage and initial assessment | Within 7 business days, including rough severity and next steps. |
| Status updates | At least every 14 days while the report remains open. |
| Coordinated disclosure | A reasonable remediation window, typically 90 days, before public disclosure. |
| Resolution priority | Critical and high-severity issues are prioritized for the next scheduled release; lower severity is scheduled into a future release. |

**Table 10.** Published response targets for vulnerability reports.

## Report privately

Vulnerability intake: Glacier private vulnerability form. Include affected app and iOS versions, device model, reproduction steps, impact, and non-sensitive supporting evidence when possible.

## How we keep this paper current

- We version this whitepaper and publish a dated change history when material data flows, providers, retention practices, or cryptographic boundaries change.
- We work to keep our App Store privacy disclosures, website, in-app permission copy, and Privacy Policy aligned with implemented behavior.
- We distinguish production retention and deletion outcomes by provider instead of describing separate systems as one operation.
- When independent security reviews, penetration tests, or certifications are available, we identify their scope and date as separate evidence.

APPENDIX A

# Control and evidence matrix

| CONTROL | EVIDENCE BASIS | CUSTOMER RESULT / SCOPE |
|---|---|---|
| WireGuard tunnel through iOS Network Extension | iOS implementation + WireGuard protocol documentation | Encrypts traffic to the Glacier gateway and masks the access-network IP from ordinary destinations. |
| DNS-over-TLS profile and randomized health check | iOS implementation + RFC 7858 | Encrypts resolver traffic and verifies that the selected iOS profile responds as expected. |
| DNS profile drops source IP and domain fields | Reviewed backend default profile + provider documentation | Supports aggregate statistics without itemized domain or client-IP activity. |
| Account-bound DNS profiles and repair | iOS DNS controller/healer + reviewed backend device ownership | Uses the current account's profile, requires an owned device, verifies repair, and avoids persisting the shared backup profile. |
| Pseudonymous DNS provider profile names | Reviewed backend provider-identity services and migration tooling | Uses an account UUID instead of email; controlled tooling can rename legacy profiles while preserving settings. |
| VPN profile authorization and lifetime | Reviewed backend entitlement and profile-delivery services | Checks entitlement for every request and limits compatibility links to 120 seconds; authenticated direct delivery is also implemented. |
| VPN lifecycle revocation | Reviewed backend account lifecycle | Revokes WireGuard peers and deletes profile objects across claimed and prior core assignments before finalization. |
| DNS/VPN no-activity-log commitment | Published Privacy Policy | Commits Glacier not to store itemized DNS or VPN activity; aggregate query totals are the stated exception. |
| SQLCipher application database | iOS implementation | Encrypts application records at rest. |
| Random database key in Keychain | iOS implementation + Apple Keychain model | Separates encrypted database content from its cryptographic key. |

| CONTROL | EVIDENCE BASIS | CUSTOMER RESULT / SCOPE |
|---|---|---|
| iOS Data Protection and backup exclusion | iOS implementation and entitlement | Adds platform protection and keeps the database directory out of backups. |
| CA public-key pinning | iOS pinning implementation and ATS configuration | Adds a pinned CA public-key check on core managers and selected authentication domains. |
| Ephemeral HTTP sessions | iOS implementation | Reduces persistent cookies, credentials, and URL caching in covered request paths. |
| Security posture checks | iOS implementation | Surfaces selected device, version, proxy, and protection signals locally. |
| No advertising tracking | Privacy manifest + published policy | Declares tracking disabled and commits Glacier to no ads or trackers. |
| On-device contact matching | iOS implementation | Keeps the full address book on device; selected numbers enter the chosen calling flow. |
| Pseudonymous Twilio identity | Reviewed backend provider-identity, token, number services, and migration tooling | Uses an account UUID in routine identity fields; controlled tooling can migrate legacy email-keyed resources. |
| Calling entitlement and ownership | Reviewed backend token, number, caller-ID, release, and webhook services | Requires entitlement for new calling capability, enforces number ownership, and authenticates provider callbacks. |
| Restricted administrative plane | Reviewed infrastructure and reverse-proxy configuration | Removes public SSH, gates sensitive routes to the tailnet, encrypts root volumes, and uses short-lived deployment roles. |
| Privacy-safe diagnostic timeline | Reviewed customer-debug event schema and access path | Keeps bounded outcome events for 180 days while rejecting direct identifiers, content, secrets, network activity, and raw errors. |
| Twilio call transport | Glacier integration + Twilio documentation | Uses TLS signalling and SRTP media on the app-to-Twilio leg. |

| CONTROL | EVIDENCE BASIS | CUSTOMER RESULT / SCOPE |
|---|---|---|
| Account deletion | Cognito, iOS cleanup, and reviewed backend lifecycle flow | Deletes authentication identity, removes local network state, and revokes backend VPN access; subscription and provider operations remain distinct. |
| Private vulnerability process | Security Policy | Provides a private reporting channel and published response targets. |

**Table 11.** Every control in this paper, its evidence basis, and the customer result.

APPENDIX B

# Glossary

| TERM | PLAIN-LANGUAGE DEFINITION |
|---|---|
| Aggregate | A numeric summary that does not contain the underlying itemized events, such as a total blocked-query count without domain names or timestamps. |
| CA public-key pinning | A TLS control that requires a server certificate chain to contain a known certificate-authority public key in addition to passing normal validation. |
| DNS | The Domain Name System, which translates service names into network addresses. |
| DNS-over-TLS (DoT) | DNS carried inside a TLS connection between a client and resolver. |
| End-to-end encryption (E2EE) | Encryption in which only the communicating endpoints possess content-decryption keys. Transport encryption to an intermediary is not E2EE. |
| Ephemeral session | An iOS networking configuration that avoids persistent cookies, credentials, and URL cache storage by the session. |
| Pseudonymous identifier | An identifier that omits a direct name but can still distinguish or associate a device, installation, or account. |
| SQLCipher | An encrypted SQLite-compatible database implementation we use for local application records. |
| SRTP | Secure Real-time Transport Protocol, used to protect real-time media on a transport leg. |
| Trust boundary | A point at which data protection, operator control, or visibility changes. |
| WireGuard | A VPN protocol using modern cryptography to create an authenticated encrypted IP tunnel. |

**Table 12.** Terms used in this paper.

APPENDIX C

# Source basis and references

The following public sources support our policy, protocol, platform, and service-boundary statements. Evidence about Glacier's implementation comes from the dated iOS and reviewed Console backend source described in Section 1.

S1 **Glacier Privacy Policy** — theglacierapp.com/privacy. Data-use commitments, provider list, VPN/DNS logging statements, calling-data categories, and privacy-request controls.

S2 **Glacier App Store listing** — apps.apple.com. Public version, compatibility, product description, and Apple privacy disclosure available at review time.

S3 **Glacier vulnerability reporting** — private intake form. Our private intake channel for good-faith vulnerability reports.

S4 **Apple Keychain Services** — developer.apple.com. Apple platform model for storing passwords, keys, certificates, and other secrets.

S5 **WireGuard protocol and cryptography** — wireguard.com/protocol. WireGuard handshake, authenticated encryption, key agreement, and protocol properties.

S6 **RFC 7858: DNS over Transport Layer Security** — datatracker.ietf.org. Internet standard defining DNS over TLS.

S7 **Apple Network Extension** — developer.apple.com. Apple framework for VPN, DNS, and related network configurations.

S8 **Twilio Voice SDK network connectivity** — twilio.com/docs. Twilio documentation for TLS signalling and SRTP media on the SDK-to-Twilio transport legs.

## Our iOS implementation areas

- Application privacy manifest, entitlements, and Info.plist permission descriptions.
- `GlacierPinningConfiguration`, `SecurityCenter`, `DnsOverTlsController`, `DNSProfileHealer`, `WireGuardManager`, and VPN settings/onboarding paths.

- `DBStorageCoordinator` and Keychain-backed account/storage models.
- `TwilioBackendManager`, `CallManager`, `ContactsManager`, and call-history/voicemail models.
- User-pool authentication migration, emailed-code sign-in flow, pending-signup Keychain storage, StoreKit entitlement, Settings sign-out, and account-deletion flows.

## Our reviewed backend implementation areas

- Provider identity, NextDNS, device ownership, entitlement, VPN profile-delivery, and account-lifecycle services.
- Twilio token, number ownership, caller-ID, release, webhook, and legacy-identity migration services.
- Authentication discovery and email-code enforcement, explicit API input/output schemas, Apple receipt verification, and generic external error handling.
- Tailnet administrative routing, HTTP security policy, public-SSH removal, encrypted storage, pinned images, and GitHub OIDC deployment configuration.
- Privacy-report aggregate DNS analytics and the allowlisted, 180-day customer diagnostic outcome timeline.

## Document history

| VERSION | DATE | SCOPE |
|---|---|---|
| 1.0 | 31 July 2026 | Initial customer edition covering Glacier for iOS privacy, security, data flows, controls, and limitations. |
| 1.1 | 2 August 2026 | Reframed around concrete protections and added reviewed backend evidence for pseudonymous Twilio/DNS identities and DNS activity minimization. |
| 1.2 | 29 August 2026 | Updated after the August security review: account-bound iOS network state, entitlement and ownership enforcement, short-lived VPN delivery and revocation, restricted operations, privacy-safe diagnostics, and controlled legacy provider-identity migration. |
| 1.3 | 21 October 2026 | Editorial update. |

OUR PUBLICATION PRACTICE

## Security transparency is a living record

For our current privacy terms and customer controls, consult our published Privacy Policy and the latest App Store release. We will revise this whitepaper when providers, data use, cryptographic boundaries, or deletion behavior change materially.