Throw Away the Key Encryption | End-to-End Encryption

August 26, 2026. We may update this paper from time to time to reflect the improvements we make to delight customers, such as strengthening protections, optimizing performance, and increasing resilience.

Abstract

Ring's mission is to make neighborhoods safer. Privacy, security, and customer control are foundational to that mission. Until now, Ring has protected customer videos using industry-standard encryption. By default, all customer video content is encrypted in transit and at rest in the cloud. Ring also offers End-to-End Encryption as an opt-in for customers who want maximum control and was the first major provider of smart home security cameras to do so. This paper describes a significant upgrade to Ring's encryption offerings.

Summary

Throw Away the Key Encryption (TAKE) is rolling out gradually as the new default encryption for all Ring customers. It gives you even greater control over how and when Ring can process your videos, while keeping the features you already use. With TAKE, your videos are encrypted with keys that change frequently; Ring can only unlock its copy of those keys within a secure enclave. Ring's access to the enclave is restricted by access controls, cryptography, and hardware isolation, so that Ring can only use keys to power the features active on your account. Ring permanently deletes its copy of your keys after 24 hours. If you later enable a feature that needs to process older videos, your Ring app can temporarily provide the keys to power that feature, and Ring deletes them again afterward.

End-to-End Encryption (E2EE) continues to be available to customers as an option. In this optional mode, Ring does not have access to decrypted content, even temporarily. Ring stores only encrypted video and cannot decrypt it; only you hold the keys needed to decrypt video. Shared Users and features that require Ring to process video, such as Video Search and Video Description, are not available under E2EE. Features that do not require Ring to process video, such as Live View, event playback, and 24/7 recording, keep working.

Why this matters: Ring designed TAKE to be our strongest default[1] encryption to date. With TAKE, Ring's copy of your keys is unlocked inside a secure enclave, which releases them only to power the features active on your account, and Ring permanently deletes that copy after 24 hours; only you keep a lasting copy. This upgrade reinforces a commitment: privacy is a core principle of how we build and design at Ring. This architecture is a foundation that we will continue to invest in and strengthen.

Both modes (TAKE and E2EE) will be available to all supported Ring camera customers regardless of subscription status. TAKE is rolling out gradually, and customers will be notified when they're eligible to enroll. Full feature compatibility will become available over the TAKE rollout period – you can visit the Ring Help Center for information about which features are compatible with TAKE when you are invited to enroll.

This paper outlines Ring's new encryption architecture for video captured by Ring cameras and stored on Ring's infrastructure.[2,3] Videos that customers choose to share with third parties outside of Ring are covered by those third parties' own security frameworks.

[1] TAKE will become Ring's default level of encryption once it is fully rolled out to all customers.
[2] This white paper describes how Ring's encryption architecture protects videos stored on behalf of customers in connection with a Ring subscription (including free trials). It does not apply to videos customers choose to share with Ring or others, such as videos shared through Ring's video donation tool, share links, or Neighbors.
[3] As described further in this paper, certain recovery methods and feature compatibility will not be available until TAKE has been rolled out to all customers.

What This Means in Practice

With TAKE:

1. Your doorbell detects motion at your front door and notifies you. Your videos are encrypted with keys that change frequently. Ring receives a copy of those keys and unlocks them in a secure enclave.
2. Ring temporarily uses those keys to process the video to support features active on your account, such as Video Search.
3. After 24 hours, Ring permanently deletes its copy of the keys and can no longer process your videos on its own.
4. A week later, you open the Ring app to watch the clip. Because Ring no longer holds the keys, your app provides the key Ring needs to prepare the video for playback, such as optimizing the video quality for your connection. Ring deletes that key when that processing finishes, and your phone, tablet, or web browser decrypts the video so you can watch it.

With End-to-End Encryption:

1. Your doorbell detects motion at your front door and notifies you. Your videos are encrypted with keys that change frequently. Ring does not receive a copy of those keys.
2. Ring stores the encrypted video and delivers it to your devices, but cannot decrypt it at any point.
3. Ring doesn't hold the keys, so there is nothing for Ring to delete.
4. A week later, you open the Ring app to watch the clip. No key is sent to Ring, so Ring cannot prepare the video for playback, such as optimizing the video quality for your connection; your phone, tablet, or web browser decrypts and plays it on its own.
5. Because only you have the keys, features like Video Search, Video Description, and Shared Users, are not available.

TAKE gives you full feature access with stronger controls. Customers who prefer sole control with no cloud feature access, may enable E2EE. Videos remain protected under the mode they were recorded in, even if you later change modes. Changing modes only affects new recordings.

What Changes with TAKE

TAKE will be the new default for Ring videos, replacing the current default once fully rolled out to all customers. It builds on Ring's existing protections and adds several important changes:

	TODAY	WITH TAKE
DEFAULT ENCRYPTION	Video Content is encrypted in transit and at rest	All existing protections, plus each camera adds its own extra layer of encryption with unique keys that change regularly and automatically.
CLOUD ACCESS	Ring's cloud services can access video	Ring cloud services can only access keys to power features active on your account, and only during the 24-hour window. After that, Ring can access a key only if your Ring app provides it for a feature you use.
KEY DELETION	No deletion	Ring deletes keys on a rolling 24-hour basis. After deletion, Ring no longer retains those keys.
YOUR PHONES, TABLETS, AND WEB BROWSERS	You can view your videos	Same. Your phones, tablets, and web browsers keep their own keys, so you can view your videos at any time, even after Ring deletes its copy.

Key Recovery

TAKE and E2EE offer several recovery methods so that customers maintain access to their videos in the event of a lost device; Ring cannot perform these actions on the customer's behalf. Recovery is self-service: the customer chooses from multiple recovery methods, with all methods available once TAKE is fully rolled out. If you are enrolled in TAKE, you can check which recovery methods are available in the Ring App by going to Control Center > Video Encryption > Enrolled Cameras.

RECOVERY METHOD	HOW IT WORKS	TAKE	E2EE
CLOUD BACKUP (IOS)	Your key is stored in iCloud Keychain, restored automatically on your new iPhone or iPad	Yes	Yes
CLOUD BACKUP (ANDROID)	Your key is stored in Google Blockstore and Google Drive, and restored on your new Android phone or tablet	Yes	Yes
PUSH NOTIFICATION / QR CODE	One of your existing phone or tablet approves the new one	Yes	Yes
PASSPHRASE	You enter your recovery passphrase on the new phone or tablet	Yes	Yes
CAMERA-BASED RECOVERY	Your key is stored on your Ring camera. You open the Ring app on a new phone, stand near your Ring camera, and the app connects over Bluetooth or Wi-Fi to restore access. On by default for TAKE for cameras with compatible hardware. Can be disabled if you do not want to store keys on your Ring cameras.	Yes	No
PASSKEY	You sign in with a passkey that can restore your access.	Yes	Yes

Recovery is designed to be automatic whenever possible. For customers who need to recover manually, the passphrase generated during enrollment works on any device and any platform, making it the universal fallback.

Sharing Videos and Access under TAKE

Only your phones, tablets, and web browsers can view decrypted videos. If you remove a phone or tablet from your account, it no longer has access to your videos. Sharing access to cameras with Shared Users works the same way: when you give another person access to one of your cameras, their devices can view decrypted videos from that camera. People you add cannot view recordings captured before they were added, and you can remove their access at any time; once removed, they no longer have access to your videos.

You decide if, when, and with whom to share your videos. You can share recordings directly from the Ring app with friends, family, or others you choose, and you can also download videos to share through other methods. In regions where Ring's Community Requests feature is available, you can choose to share relevant recordings with a local public safety agency to support an official investigation in your area. We have also updated our Law Enforcement Guidelines to reflect changes resulting from the TAKE architecture.

ring

// Throw Away the Key Encryption (TAKE)

// End-to-End Encryption (E2EE)

// Architecture

1. Protocol Foundation: Messaging Layer Security (MLS)

MLS (RFC 9420) was specifically designed for group encryption scenarios where multiple devices need secure access to shared content. The standardization process addressed known cryptographic vulnerabilities, and the protocol design accommodates future cryptographic advances, including support for post-quantum cipher suites as they become standardized.

1.1 MLS Protocol Overview

Each participant in an MLS group holds cryptographic key material that allows them to encrypt and decrypt group messages. The protocol manages how this key material evolves as members join, leave, or update their own keys. The protocol operates through a series of epochs, where each epoch represents a distinct cryptographic state of the group. When group membership changes or keys are rotated, the group advances to a new epoch with fresh cryptographic material. This epoch-based structure provides the foundation for forward secrecy and post-compromise security, as each epoch uses different keys derived through a cryptographic ratcheting process.

MLS uses a binary tree structure to organize group members and manage key distribution efficiently. Each member occupies a leaf position in the tree, with intermediate nodes representing shared secrets between subsets of members. In a balanced tree, this structure achieves logarithmic complexity for key updates rather than requiring linear operations proportional to group size. When a member updates their key material, only the nodes along the path from their leaf to the root require updates.

1.2 Security Properties

MLS provides forward secrecy through its epoch-based structure and key deletion requirements. When the group advances to a new epoch, members delete cryptographic material from previous epochs, preventing compromise of current state from exposing historical content.

Post-compromise security protects future communications if a member's cryptographic state is compromised. Each member automatically refreshes its own key material by issuing Commit messages as part of normal protocol operation, and each refresh replaces any previously exposed key material. From that point, an attacker holding the exposed keys can no longer decrypt new content or forge messages the group would accept.

The protocol provides authentication through both digital signatures and message authentication codes (MACs). Digital signatures verify message provenance and integrity, while MACs provide additional authentication that prevents forgeries by attackers who compromise signature keys. The protocol also includes mechanisms to detect and prevent replay attacks.

2. Shared Architecture - Throw Away the Key Encryption & End-to-End Encryption

The following architectural components are Ring-specific extensions built on top of the standard MLS protocol described in Section 1.

2.1 Group Types

Rings implementation uses two distinct types of MLS groups that serve different purposes.

Account-Level Groups manage the cryptographic membership of a customer’s devices and recovery mechanisms. Each customer has one account-level group that contains all their mobile applications, backup members used for account recovery, and the devices of any shared users (Section 5.7). The backup member allows storing an encrypted copy of the customers MLS group state, which can be unlocked through one of the recovery methods described in Section 5. This enables customers to restore access to their encrypted content on a new device. This group handles operations related to account management, client device enrollment, and recovery processes.

Device-Level Groups manage the encryption of video content from individual cameras. Each Ring camera has its own device-level group that contains the camera, the customers mobile devices, and any other authorized consumers of that cameras content. When a camera captures video, it encrypts the content using keys derived from its device-level groups MLS state. When a mobile device plays back video, it decrypts using keys derived from the same group state.

This group separation provides several architectural benefits. Account management operations, such as enrolling a new mobile device or updating recovery settings, occur in the account-level group without affecting device-level groups. Customers can also control shared users separately for each of their cameras.

When a customer enrolls a new mobile device, the device first joins the account-level group, establishing its cryptographic identity. The device then joins each cameras device-level group, gaining access to decrypt video content. When a device is removed, it is removed from both group types. This change advances each group to a new epoch with new keys that are distributed only to remaining members, cryptographically preventing the removed device from decrypting future content.

2.2 Key Lifecycle

Cryptographic keys in the system progress through distinct lifecycle phases.

MLS member keys represent the long-term cryptographic identity of each device. When a mobile device enrolls, it generates an Ed25519 signing key pair. The device stores the private signing key securely in the application's secure storage space. While these keys serve as the device's identity, they can be rotated by the client during the lifetime of enrollment for improved security.

Epoch keys have shorter lifecycles tied to MLS epochs. Key material changes when the group advances to new epochs through Commit operations. When a device is removed, the remaining members advance to a new epoch with key material unknown to the removed device, preventing it from decrypting subsequent content.

Content encryption keys have the shortest lifecycle, tied to the time-based rotation intervals. New five-minute keys are derived at each interval boundary. Customer devices retain the keys needed to decrypt content they are authorized to view. Ring's cloud services retain these keys only for a limited retention window (described in Section 3.2), after which they are permanently deleted. Section 2.5 describes how these keys are derived.

Recovery key material follows a distinct lifecycle tied to account recovery. Recovery keys are stored in customer-controlled cloud storage (iCloud Keychain, Google Blockstore, or Google Drive), within passkeys via the PRF extension, on enrolled Ring cameras (under TAKE), or derived from a customer-memorized passphrase. Recovery keys do not automatically rotate with each MLS epoch change, but they can be rotated when required. This independence from epoch changes allows recovery even if the customer has been offline during many group updates.

2.3 Privacy Enforcement Policy

Each MLS group maintains a Privacy Enforcement Policy (PEP) as part of its cryptographic state. The PEP defines which group members have permission to perform specific operations, such as adding or removing members, changing encryption settings, or modifying access control rules. These permissions are embedded in the MLS group state, whose authenticity and integrity are enforced by the protocol. Any modification to the PEP is validated by all group members, and only members with the necessary permissions can change them.

Under TAKE, a cloud member managed by the Cloud Member Management Service (CMMS) is added to each camera's encryption group to enable content decryption for Ring features (described in Section 3). This cloud member's group permissions are restricted to content access. It cannot modify group membership, change encryption settings, or alter PEP rules. Adding or removing the cloud member can only be done by one of the customer's authorized apps or browsers, because PEP grants that permission to those members alone. Ring's cloud cannot add itself to a group. Ring's infrastructure also validates that proposed changes comply with PEP rules as an early validation layer, but the primary enforcement occurs on customer devices. Enabling End-to-End Encryption removes the cloud member, so under E2EE no cloud member is present in the group (Section 4.1).

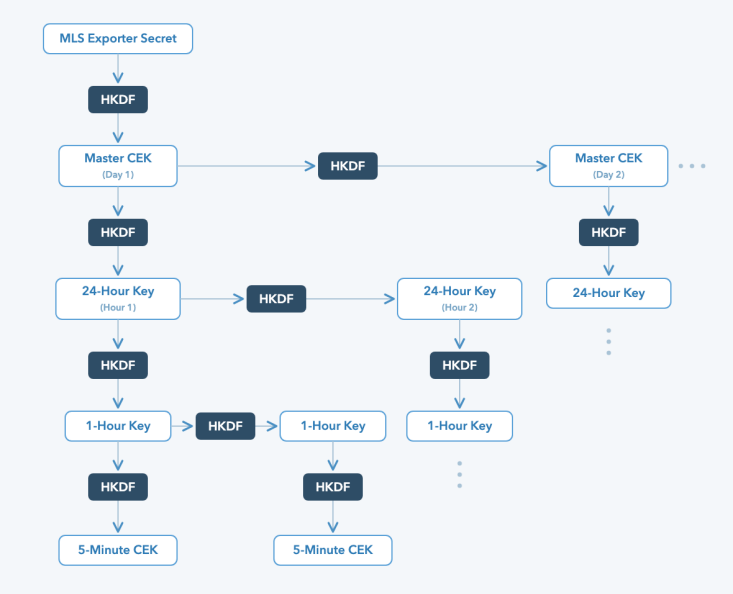
2.4 SecureFrame Video Encryption

Ring developed its own custom encrypted frame format (Ring SecureFrame), for securing video and audio content in streaming applications. Video streaming requires a format optimized for the high bandwidth and real-time constraints of continuous media streaming and playback. SecureFrame provides this format while maintaining the security via the underlying MLS key material.

Each SecureFrame encrypted media frame consists of the encrypted video or audio payload, a unique initialization vector, the encryption key identifier, a continuity counter as well as an authentication tag that allows detection of tampering. The format supports the multiple video and audio codecs being used by Ring. One of the specific properties of our SecureFrame implementation is that it allows leaving the video codec configuration data and headers unencrypted within each frame, while the portions containing image data are protected. The unencrypted headers are still covered by the frame's authentication tag, so tampering with them is detectable. This allows deploying SecureFrame encryption on existing streaming and media infrastructure, because those systems can handle and route encrypted frames at the container level without needing to decode the video stream. This property is valuable because it enables Ring to maintain its existing infrastructure performing operations such as RTP packetization, transmuxing, and snapshot extraction on encrypted content without requiring access to decryption keys. These operations work at the container level using the unencrypted headers; the image data itself stays encrypted, so an extracted snapshot remains encrypted and can be decrypted only by devices that hold the keys.

The frame-level encryption approach provides several operational advantages. Each frame can be encrypted and decrypted independently, allowing random access to any point in a recorded video without requiring decryption of surrounding frames. This independence is critical for real-time streaming scenarios which are lossy by nature (not all frames are guaranteed to be delivered) and enables features like timeline scrubbing.

2.5 Content Encryption Key Hierarchy



Rings implementation includes a hierarchical key derivation system that bridges between MLS group secrets and the content encryption keys used for individual video frames. The system derives a root content encryption key from the current MLS groups exporter secret using a key derivation function. From this root key, the system derives time-bounded content encryption keys through intermediate derivation levels:

- 24-hour keys derived from the root key
- 1-hour keys derived from 24-hour keys
- Final 5-minute Content Encryption Keys derived from 1-hour keys

Within each five-minute interval, all frames from a camera are encrypted using the same time-bounded key, with each frame using a unique initialization vector constructed from frame counters and stream identifiers. Each frame also carries a key identifier in its SecureFrame metadata, allowing receiving devices to select the correct key for decryption.

The boundaries of these time intervals are anchored to the commit time of the MLS epoch they belong to. The commit time is included in the commit itself as a PEP setting, and the MLS Delivery Service validates these timestamps against an allowed drift window, rejecting commits whose timestamps fall outside the permitted range.

This hierarchical structure limits the amount of content encrypted under any single derived key, reducing exposure if a content encryption key is compromised. The separation between MLS epochs and content encryption key rotation allows the system to rotate content keys independently of MLS group membership changes and thus even without network access or any kind of explicit synchronization mechanism with the other members of the group. This hierarchical structure is also critical to provide applications with fast random access to any of the time-bounded Content Encryption Keys.

2.6 Content Flow

When a Ring camera captures video, the camera's video encoder compresses the video into encoded frames. For each frame, the camera retrieves the current five-minute content encryption key, constructs a unique initialization vector, and encrypts the frame using AES-128-GCM. The authentication tag is computed over the entire frame, included as part of the SecureFrame footer, and the camera transmits the encrypted frame to Ring's infrastructure. Ring's cloud infrastructure receives and stores the encrypted frames along with metadata such as timestamps.

The customer's mobile device maintains its MLS group memberships through background synchronization. The Ring application periodically retrieves new MLS messages, verifies their signatures and PEP compliance, and updates its local group states. This synchronization happens independently of playback requests.

When a customer requests playback or initiates live view, the application looks up its local camera device-level group state, which it was given access to as part of the enrollment process, and derives the content encryption keys needed for the requested time range through the hierarchical key derivation. As encrypted frames arrive, the application verifies the authentication tag, decrypts the frame payload, and passes the decrypted frame to the video decoder for display.

2.7 MLS Delivery Service

The MLS protocol architecture (RFC 9750) defines a Delivery Service responsible for ordering and routing MLS messages between group members. Ring's MLS Delivery Service (MDS) fulfills this role as the cloud-based platform that manages MLS group communications. MDS stores group information, maintains group state, and facilitates secure message distribution among group members such as cameras and applications. MDS routes Welcome messages to new members joining groups and Commit messages that advance group state. MDS validates that all proposed group changes comply with PEP rules before accepting them. MDS operates only on the public portions of MLS messages (such as routing metadata and PEP rule changes). It does not possess the keys needed to decrypt customer video content or the encrypted portions of MLS messages.

3. Throw Away the Key Encryption

TAKE is designed for all Ring customers. It provides stronger encryption protections while preserving access to the full set of Ring features. The cloud member, CMMS, and the key retention and deletion behavior described in this section exist only under TAKE; under End-to-End Encryption, Ring's cloud never holds decryption keys (Section 4).

3.1 Cloud Member, Key Management, and Content Processing

Under TAKE, a cloud member is added to each camera's device-level encryption group. This cloud member represents Rings cloud services within the MLS group and allows those services to derive the keys needed to decrypt video content for customer features.

The cloud member's access is scoped. It is granted access to content encryption keys to power features active on the customer's account, such as cloud-based Smart Alerts, Video Search, Smart Video Descriptions, and other cloud-based analysis. The cloud member participates in the standard MLS key derivation process and is subject to the same PEP rules as any other group member.

Content encryption keys for cloud services are generated and managed through the Cloud Member Management Service (CMMS), which operates within AWS Nitro Enclaves. Nitro Enclaves are isolated, hardened compute environments that prevent the root key material from being accessed outside the enclave boundary. Platform services that power customer features retrieve individual content encryption keys from CMMS only when required to power a cloud-based feature, and strict access control rules are enforced by CMMS. These access control rules are based on the PEP defined within each MLS group.

Content processing (such as object detection for Smart Alerts or Video Search) occurs within the platform service of the feature. As designed, decrypted content does not leave the platform service boundary in unencrypted form. Content encryption keys are constrained to the SDK memory boundary, do not persist to disk, and are overwritten in memory (zeroized) after processing completes.

3.2 Key Retention and Deletion

TAKE implements a 24-hour key retention window. During this window, content encryption keys are managed within AWS Nitro Enclaves and made available to platform services only for the duration of feature processing. During the 24-hour retention window, cloud services retrieve keys from CMMS for the duration of feature processing, such as Smart Alerts and other features active on the customer's account. This access is governed by the CMMS access control rules described above.

Deletion is continuous rather than a single event at the 24-hour mark. Content encryption keys rotate every five minutes, and as each key ages past 24 hours, CMMS ratchets forward the corresponding intermediary secret in the key derivation hierarchy. Ratcheting applies a one-way key derivation function to overwrite a managed secret with its derived output and discards the original, making it cryptographically infeasible to reconstruct. Ratcheting is triggered both reactively, when requests are made to CMMS, and on a scheduled basis through periodic upkeep. The CMMS database is configured with no backups. This deletion is designed to be irreversible.

Once a key has been deleted, the content encrypted under it is no longer accessible by Ring (except as described in Section 3.3). Customer devices retain their own keys and can continue to access their content. Section 3.5 describes the safeguards that enforce this window and the further hardening Ring is exploring.

3.3 On-Demand Key sharing from the App

When a customer takes an action that requires cloud processing, for example playing back older content, sharing a video, or authorizing a feature that needs to process past recordings, the Ring application on the customer's device determines which content encryption keys are needed and provides them to Ring's cloud services for on-demand processing such as bandwidth adaptation, transcoding, or feature-specific analysis. Key delivery from the app is push-only, so a key is sent only as a result of something the customer chose to do, meaning Ring's cloud cannot retrieve a key on its own or cause a device to release one. The app provides keys at the level of the derivation hierarchy that covers the time range the action requires, which is what lets a customer scrub through a recording without a round trip for every segment. The keys are not persisted and are discarded when the processing session finishes.

3.4 Two Encryption Paths

TAKE supports all Ring cameras, though the point at which encryption occurs varies by hardware capability:

- **Newer camera models** with encryption-capable firmware perform encryption on the camera itself. Video is encrypted before it leaves the device, using keys derived from the camera's device-level MLS group. These cameras support both TAKE and End-to-End Encryption.
- **Older cameras that do not support** on-device encryption rely on encryption at cloud ingress. Content is encrypted using keys managed by CMMS as soon as it reaches Ring's infrastructure. The result is the same: videos are encrypted with keys that change frequently, cloud access is scoped and time-limited, and keys are deleted after 24 hours. This method supports TAKE only; it does not support End-to-End Encryption.

3.5 Access Safeguards and Future Hardening

Two questions follow naturally from this design: what prevents Ring itself from reading keys while it holds them, and what prevents recovering them after they are deleted. While keys are held, access is constrained by several independent layers. Key material is generated and managed inside AWS Nitro Enclaves, which provide no persistent storage, no interactive operator access, and no general network connectivity. Root secrets never leave the enclave; the short-lived content keys derived from them are released to a platform service only when a feature needs them and remain confined to that service's SDK memory (Section 3.1). The key management policies protecting stored encrypted key material release it only to an enclave (CMMS) that cryptographically attests to running the exact approved software image; a Ring operator, or a tampered image, cannot satisfy that attestation. These policies are maintained as code and validated in the deployment pipeline. CMMS authenticates every request from the platform services, authorizes each service only for the narrow operations its feature requires, and encrypts each key to the requesting SDK instance, so nothing between CMMS and that instance can read it. Every release of key material is further governed by the PEP rules in the customer's MLS group and is logged and monitored with automated alarms.

Those layers apply to key material that CMMS holds and releases. Keys supplied by the customer's app (Section 3.3) are sent directly to the encryption SDK inside the service doing the processing. The app encrypts each key to the receiving service before it leaves the device. The key is decrypted only inside that service's encryption SDK, stays in the SDK's memory, is not written to disk, and is discarded when the processing finishes (Section 3.1).

Once the retention period ends, prevention shifts from access control to cryptography. Ratcheting replaces each stored secret with the output of a one-way key derivation function and discards the prior value, so reconstructing expired keys is cryptographically infeasible. The CMMS database keeps no backups, so once a secret is overwritten neither the expired content encryption keys nor the material needed to derive them again exists anywhere in Ring's systems. Ratcheting is scheduled from timestamps that CMMS must store in readable form, so that the database can trigger the job on time. Each stored record carries a signature that only the AWS Nitro Enclave can produce. Any attempt to postpone the ratcheting by modification of these records will invalidate their signatures and result in a detectable failure. Independently of the schedule, the enclave will not release a key outside the time range the customer's group configuration allows, so a delayed deletion does not by itself make content readable.

Ring is continuing to harden this architecture. Work Ring is exploring includes releasing content keys only to services that themselves run inside an enclave and can attest to it, extending enclave protection from CMMS to the services that use the content keys, and adding post-quantum protection to key exchange.

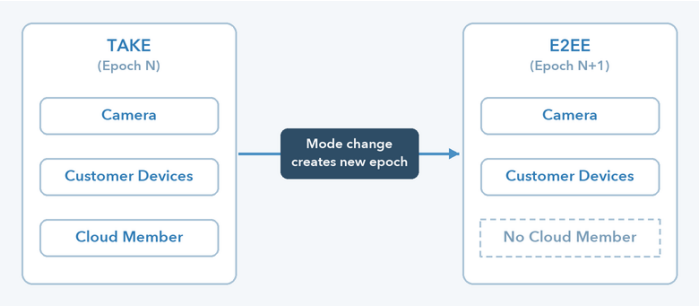
4. End-to-End Encryption

End-to-End Encryption is an opt-in mode for customers who want the highest level of control Ring offers.

4.1 Cloud Member Removal

The mechanism that distinguishes E2EE from TAKE is the removal of the cloud member from each camera's device-level encryption group. This enforcement is built into each member of the group via the Privacy Enforcement Policy (PEP). When a customer enables E2EE:

1. The cloud member is removed from the device-level group of each camera on which the customer enables E2EE. E2EE is controlled per camera, so a customer can run TAKE on some cameras and E2EE on others at the same time. The PEP enforces this requirement.
2. Each group advances to a new MLS epoch with fresh cryptographic material.
3. The cloud member's previous keys become invalid and cannot be used to derive the keys needed to decrypt content encrypted in the new epoch.
4. Shared Users attached to that camera are removed at the same time, and their devices are removed from the camera's device-level group (Section 5.7).



From this point forward, only the customer's phones, tablets, and web browsers hold the keys needed to decrypt video content. Ring's cloud infrastructure continues to store and transmit encrypted content but cannot decrypt it.

When E2EE is later disabled on a camera, the cloud member is added back to the device-level group and the group advances to a new epoch. The cloud member receives keys only for this new epoch forward. Content that was encrypted while E2EE was enabled was encrypted using keys from earlier epochs that the cloud member never possessed. As a result, that content remains inaccessible to Rings cloud services regardless of the current encryption mode. This is a direct consequence of the epoch-based key structure. Each recording is encrypted with the keys from its epoch, and mode changes create new epochs with new keys.

4.2 Feature Trade-offs

Because CMMS does not have decryption keys under E2EE, features that depend on cloud-based video or audio analysis or processing are not available. This includes Video Search, Smart Video Descriptions, cloud-based Smart Alerts, and other features that require Ring's infrastructure to process decrypted content.

5. Enrollment and Recovery

5.1 Sign-In vs. Content Access

Ring's encryption framework introduces a distinction between account access and content access. Customers can sign into their Ring account on any device using their standard authentication methods: passwords, passkeys, or other credentials. However, signing in alone does not grant access to encrypted video content. To view encrypted videos, the customer's device must possess the content encryption key material.

The system is designed so that, wherever possible, gaining access to encrypted content happens as part of the normal sign-in process without extra steps for the customer. A customer on iOS with iCloud Keychain enabled signs in normally and their device retrieves the key material in the background. A customer who still has a phone or tablet with access to encrypted content can also approve a new one from it. Section 5.3 describes the full set of methods and how each one works.

5.2 Enrollment

Enrollment occurs through the Ring mobile application during camera setup or when the customer enables TAKE or End-to-End Encryption. The enrollment process:

1. Creates an account-level MLS group for the customer (if one does not already exist), containing the enrolling mobile device and a backup member for account recovery.
2. Establishes PEP rules defining which members can perform various operations.
3. Generates a recovery passphrase (described below).
4. Configures automatic cloud-based backup of recovery key material where available.
5. Creates device-level MLS groups for each camera, with the camera and mobile devices as members.

For TAKE, a cloud member is added to each device-level group during enrollment. For E2EE, the cloud member is excluded.

Cloud-Assisted Enrollment

When a customer first enables TAKE or End-to-End Encryption, the system uses cloud-assisted enrollment to add all of the customer's existing signed-in Ring apps to the MLS groups at once. Cloud-assisted enrollment exists so that customers who already own cameras can move to TAKE without friction, by avoiding the need to manually enroll each device they already have. Only devices with an active sign-in session on the account are included. The MLS Delivery Service (MDS) provides the MLS Key Packages for each existing app. MDS validates the provenance and authenticity of each Key Package, checking that it was issued by a device belonging to the correct Ring user account. The enrolling device is responsible for the actual addition of each Key Package to the MLS group; at no point does Ring's cloud have the permissions required to add these members to the group directly. Cloud-assisted enrollment is also used after an encryption account reset, to re-add the customer's other signed-in devices to the newly created groups. Customers can review what cloud-assisted enrollment did on their behalf using the post-enrollment identity verification and Activity Log features described in Sections 5.4 and 5.5.

Cameras that are already installed are enrolled the same way, and for the same reason. When a customer enables TAKE, their existing cameras have no proximity connection to the app at that moment, and requiring a physical re-setup of every camera would make encryption by default impractical for anyone who already owns cameras. Instead, the app creates the device-level group for each existing camera, including the camera's own membership in it, and stages the camera's group state with Ring's cloud. Ring's cloud forwards that state to the camera, which takes it over. This happens once per camera. A new camera often ships with older firmware. It is set up first, updates its firmware, and then enrolls, so it takes the same staged path. Where a camera is already running encryption-capable firmware at setup, including any later re-setup, enrollment uses the Bluetooth or Wi-Fi setup flow instead, with the app and the camera exchanging keys directly and no cloud involvement. Before the camera encrypts anything, it advances to a new encryption epoch, so no key in the staged package can open any video the camera records.

Apart from the cases above, cloud-assisted enrollment is not used again. Adding a new device to the account instead relies on strong identity verification of the new device's Key Package before it is added to the group. A customer can add a new device using any of the recovery methods described in Section 5.3, for example approval from an existing phone or tablet via push notification or QR code, a cloud backup restore, the recovery passphrase, or Camera-Based Recovery. In every case, only the customer, through key material they control, can authorize a new device to access encrypted content.

5.3 Recovery Methods

Because the encryption framework places encryption keys under customer control, the system provides multiple redundant recovery methods. Not every method is available at launch; all of them will be available once TAKE is fully rolled out. Providing multiple methods

maximizes the chances that a customer will be able to recover access to their encrypted videos if they switch phones, lose a device, or need to set up a new one.

iOS Cloud Backup
The Ring app on the customer's device stores recovery key material in iCloud Keychain. When a customer signs in on a new iOS device using the same Apple ID, the Ring app automatically retrieves the key from iCloud Keychain and restores access to encrypted content.

Android Cloud Backup
On Android, Ring uses two complementary mechanisms:

- Google Blockstore is where the Ring app stores key material that is automatically backed up and restored during the Android device-to-device transfer process. When a customer sets up a new Android device by restoring from a backup using the same Google account, the key is automatically available.
- Google Drive Backup provides an additional cloud-based recovery path. The Ring app can store key material in the customer's Google Drive, providing recovery options beyond device-to-device transfers. This requires the customer to grant Ring permission to access Google Drive during enrollment.

Push Notification / QR Code from Another Authorized Device
When a customer has access to a phone or tablet with access to encrypted content, they can add a new device directly. Two methods are available, both of which work across operating systems (an iPhone can approve an Android device, and vice versa).

With QR code approval, the new device displays a QR code containing a fingerprint of its MLS Key Package. The customer scans the code with their authorized device's camera. Because the fingerprint is compared directly between devices without passing through any intermediary, no additional verification is needed. The authorized phone or tablet then adds the new device to the customer's encryption groups through MDS, and the new device receives MLS Welcome messages containing the group state needed to derive content encryption keys.

With push notification approval, the key exchange passes through Ring's cloud infrastructure. To protect against tampering in transit, both devices exchange fresh random salts and compute a verification code derived from both devices' public keys and salts through a password-based key derivation function. This protocol is designed so that any attempt to substitute key material in transit would produce mismatched codes, even when the communication channel is treated as untrusted. After the customer confirms a matching code on both devices, the authorized device adds the new device to the encryption groups through MDS, and the new device receives the Welcome messages needed to derive content encryption keys.

Manual Passphrase
During enrollment, the system generates a twelve-word recovery passphrase from the standardized BIP39 word list, which carries 128 bits of entropy. The customer is responsible for recording and storing this passphrase securely. Ring does not store the passphrase. When setting up the Ring application on a new device, the customer can enter the passphrase to restore access. That entropy is high enough that the passphrase can be used directly to derive a private recovery key without a computationally intensive key stretching step; if a shorter passphrase format is adopted in the future, a key derivation function such as PBKDF2 would be applied to compensate. The backup holds the recovery member's own private MLS state, encrypted to the corresponding public recovery key and stored on Ring's infrastructure. Recovery restores that member, brings it up to date with the group's latest changes, and uses it to add the customer's new device. Because the backup is encrypted using asymmetric encryption, the passphrase (and the private key derived from it) is never used during backup creation, only during recovery. The Ring app also encrypts the public recovery key itself under a secret held by the MLS group before storing it, so only a group member can produce a valid encrypted backup. Ring's infrastructure stores only the encrypted public key and the encrypted backup, neither of which can be used to recover the passphrase or decrypt the backup.

This method works on any device and any platform, making it the universal fallback when other methods are unavailable. Customers can regenerate a new passphrase from any authorized phone or tablet, which generates a new recovery key pair and flags all group backups for re-encryption to the new key.

Camera-Based Recovery (TAKE Only)
Under TAKE, the Ring application provisions a recovery key onto each of the customer's cameras. The recovery key is an asymmetric key pair: the camera stores only the private key, while the public key is stored encrypted in the account-level group. Backups of the customer's MLS group states are encrypted to this public key (alongside other recovery keys), so the camera's private key can be used to decrypt them during recovery. The key is provisioned through the MLS group itself, encrypted so that only the target camera can read it.

When a customer needs to recover on a new mobile device, they sign into the Ring app, select one of their cameras, and physically approach it. The app connects to the camera via Bluetooth or direct Wi-Fi, verifies that the camera and app belong to the same Ring account, and retrieves the recovery key over this local connection. Physical proximity to the camera is required, which is designed to protect this recovery path from remote attacks.

Camera-Based Recovery is enabled by default under TAKE. Customers can disable it while retaining all other recovery methods. Camera-Based Recovery requires Ring's encryption SDK on the camera hardware; legacy cameras that rely on cloud ingress encryption do not support it.

Camera-Based Recovery is not available for accounts that use End-to-End Encryption. Under E2EE, only the customer's authorized phones, tablets, and web browsers hold recovery key material. All other recovery methods (cloud backup, passphrase, passkeys, and push notification or QR code from a device with access to encrypted content) remain available under E2EE.

Passkey Access

Passkeys support a PRF (pseudo-random function) extension that allows applications to derive additional secrets from the passkey itself. Ring uses the PRF extension secret as a recovery encryption key. When a customer signs in with a passkey that has this PRF data, they get access to encrypted content without additional steps.

Passkeys have known limitations in complex multi-device and cross-platform scenarios. Passkeys stored in platform-specific providers (iCloud Keychain, Google Password Manager) do not sync across operating systems. Multiple passkeys for the same account can exist across different providers. Passkeys created under one person's credentials cannot be shared with another person on a shared Ring account. The other recovery methods address scenarios where passkey limitations exist.

Passkeys created before encryption enrollment that lack PRF data are upgraded automatically during sign-in. When a customer signs in with such a passkey, the system automatically provisions a PRF secret for the passkey as part of the sign-in process.

5.4 Post-Enrollment Identity Verification

Ring offers optional post-enrollment identity verification. This feature allows customers to cryptographically verify that devices in their MLS groups are the actual devices they intended to enroll, independent of MDS or any other cloud service.

Identity verification is implemented via QR code scanning between devices. The user's manager device scans a QR code displayed on the target device to verify that the MLS LeafNode public signing key matches what is expected. Once a member is verified, the validation can be persisted by embedding an identity-validated flag in the PEP (the group's cryptographic state).

This allows the customer to verify that Ring Cloud did not insert unauthorized devices into the group during cloud-assisted enrollment.

5.5 Device Activity Log

Customers can audit access to each of their cameras through the device settings in the Ring app. The Activity Log records group membership changes as cryptographic events within the MLS group: devices added, devices removed, E2EE enabled, and E2EE disabled. Because these events are recorded within the MLS groups cryptographic state, they form a tamper-evident audit trail that cannot be modified without detection by group members.

This gives customers visibility into who has access to their camera and when membership changes occurred. The Activity Log is available only to authorized devices for cameras that are part of an encryption group.

5.6 Self-Service Design

The recovery architecture places encryption keys under customer control. Access to encrypted content is managed exclusively through the recovery methods described above. If a customer loses access to all recovery methods (their cloud backup, their passphrase, their authorized devices, their cameras, and their passkeys), they will not be able to access encrypted content. This is an inherent property of the encryption design.

A customer who has exhausted every recovery method can perform an encryption account reset from the Ring app. A reset creates new encryption groups for the account, so the customer can record and view new video normally again. Video recorded under the previous keys stays inaccessible, because no copy of those keys exists. A reset also requires the customer to set up each of their cameras again, over the same local Bluetooth or Wi-Fi setup flow used when a camera is first installed, and a camera that is not set up again does not rejoin. The recovery methods described above exist so that a reset is rarely needed.

5.7 Shared Users

Ring lets an account owner share access to one or more of their cameras with other people, such as family members or, for business accounts, employees and installers. Sharing is managed per camera and per person, and the person receiving access keeps their own encryption keys on their own devices. The owner can change or revoke that access at any time. Shared Users is not available with E2EE, and enabling E2EE for a camera removes attached Shared Users.

Invite and Acceptance

Because a shared user may not yet have a Ring account when they are invited, and because they should not gain access to any content until they accept, the system uses a dedicated invite flow rather than the standard device-enrollment flow. When the owner extends an invite, their app creates a placeholder position in the relevant groups and provisions a single-use signing key that is delivered only to the invited person, allowing them to perform an external join operation on the MLS groups. When the invited person accepts, their device

replaces the placeholder with its own credentials and joins the groups, updating its signing key so that the single-use key shared in the invite can no longer be used to join. The invited person is added both to the owner's account-level group and to the shared camera's device-level group, which preserves the trust relationship even if a specific shared camera group is later removed.

Permissions

What a shared user can do is governed by the same Privacy Enforcement Policy that governs every other group member. Ring's existing roles map to a defined set of group permissions. Access can be granted at two scopes: to specific cameras, or to an entire location. A location-scoped share automatically extends to cameras added to that location later, because the owner's app adds the shared user to each new camera's group as it is created. A shared user with a content-only role can view the cameras shared with them but cannot manage other members, while a higher-privileged role can also invite or remove other members for a location. Some roles, such as an installer role, are time-limited and are removed automatically once they expire.

Past Content and Recovery

A shared user's device joins the camera's device-level group at the epoch in force when their access begins, and it cannot derive key material from earlier epochs, so an invite does not grant access to content recorded before it was accepted. This keeps historical recordings from being exposed to additional parties. Shared users set up their own recovery methods for their own devices. If a shared user loses access and cannot recover, the account owner can re-approve them.