

Ontaric GmbH

Resourcecloud

Änderungsprotokoll

Wer wann was von wo geändert hat – und wie du dich darauf verlassen kannst.
Anleitung für Administratoren, Auditoren und Datenschutzbeauftragte.

Stand 11.10.2026

Ontaric GmbH · Jürgen-Töpfer-Straße 113 · 22763 Hamburg
datenschutz@resourcecloud.io

Inhalt

Kurzfassung

1. Was aufgezeichnet wird
2. Wer es sehen darf und wie du es einrichtest
3. Die Seite im Alltag
4. Manipulationen erkennen
5. Betrieb: Schlüssel, Speicher, Sicherung
6. Aufbewahrung und Datenschutz
7. Schnittstellen
8. Prüfung durch Dritte
9. Grenzen
10. Häufige Fragen

Kontakt

Änderungsprotokoll

Wer wann was von wo geändert hat – bei Aufgaben, Projekten, Personen, Rechten, Einstellungen, Rechnungen und Zeitbuchungen. Einträge lassen sich nicht ändern oder löschen, und eine Prüfkette macht jede spätere Änderung erkennbar.

Kurzfassung

Eine eigene Seite

Admin > Änderungsprotokoll. Nur RC Admin und die Rolle *Auditor* können sie öffnen; sie erscheint auf keiner Aufgaben- oder Projektseite.

Nur lesen

Niemand kann Einträge bearbeiten oder löschen, auch kein Administrator. Eine Prüfkette macht Änderungen an der Datenbank erkennbar.

Ohne Zutun

Das Protokoll läuft nach der Aktualisierung von selbst. Schlüssel und Ankerkopien legt Resourcecloud im S3-Speicher ab, den es ohnehin für Dateien nutzt. Nur eine Kopie des Schlüssels musst du selbst aufbewahren.

Du entscheidest

Wer Auditor ist, wie lange Einträge aufbewahrt werden (Standard 10 Jahre) und wann eine Person pseudonymisiert wird.

Was du tun solltest

1. Prüfe, dass der **Scheduler-Container läuft** und der **S3-Dateispeicher** eingerichtet ist (Abschnitt 5). Beides braucht das Protokoll, um Einträge zu versiegeln und Ankerkopien abzulegen.
2. Gib die Rolle **Auditor** den Personen, die lesen sollen (Abschnitt 2).
3. Stelle die **Aufbewahrungszeit** ein, wenn 10 Jahre nicht zu euch passen (Abschnitt 6).
4. **Lade den Schlüssel herunter** und bewahre ihn bei euren wichtigen Geheimnissen auf; bestätige das auf der Seite (Abschnitt 5). Du als RC Admin bist für diese Kopie verantwortlich.
5. Sichere **Datenbank und S3-Bucket gemeinsam** (Abschnitt 5). Der Schlüssel zur Prüfung liegt im Bucket.
6. Informiere Betriebsrat und Datenschutzbeauftragte, dass es das Protokoll gibt (Abschnitt 6).

1. Was aufgezeichnet wird

Wird aufgezeichnet	Wird nie aufgezeichnet
Anmeldung: Anmeldung (Passwort, LDAP, Passkey, SSO), Abmeldung, fehlgeschlagene Versuche Personen und Zugriff: Benutzer angelegt, geändert, aktiviert, deaktiviert, gelöscht; Passwortwechsel; Sicherheitsrollen; Teamzugehörigkeit Einstellungen: globale Einstellungen, Teams, API-Tokens, Webhooks Aufgaben: angelegt, geändert, erledigt, wieder geöffnet, gelöscht, wiederhergestellt – mit altem und neuem Wert Projekte und Meilensteine: Status, Einstellungen, Termine, Kosten Geld und Zeit: Rechnungen (Status, Beträge), geänderte oder gelöschte Zeitbuchungen, Urlaubsanträge und -entscheidungen, Arbeitszeitkorrekturen Stammdaten: Kunden, Firmen, eigene Felder, Projektvorlagen, Basispläne Daten, die das System verlassen: Rechnungs-Downloads, Einmalpasswörter eines Sammelimports, Exporte des Protokolls selbst Importe: ein zusammenfassender Eintrag je Importlauf	Passwörter, Tokens und andere Geheimnisse (erscheinen als ***) Der Text von Kommentaren und Beschreibungen – nur, dass sich eine Beschreibung geändert hat, und ihre Länge Name, E-Mail-Adresse und Telefonnummer einer Person innerhalb einer Änderung – nur deren Länge Was jemand nur <i>angesehen</i> hat Terminverschiebungen, die der Autoplaner selbst vornimmt Das Anlegen einer Zeitbuchung (jeder Kommentar bucht Minuten); das Ändern und Löschen einer Buchung wird aufgezeichnet

Jeder Eintrag nennt: **wer** (eine Person, ein API-Token, der MCP-Zugang, der Scheduler oder das System), **wodurch** (Web, REST-API, MCP, Mail, Import, Scheduler), von **welcher IP-Adresse**, **was** geschah, an **welchem Objekt** (mit dem Namen zu dem Zeitpunkt), zu welchem **Team und Projekt** es gehört, und was sich geändert hat als *alt* → *neu*.

Private Aufgaben hinterlassen nur die Tatsache, dass etwas geschah – ohne Titel und Inhalt. Der Eintrag ist nur für die Beteiligten der Aufgabe sichtbar, auch nicht für einen RC Admin, der nicht dazugehört.

Das Protokoll beginnt mit der Aktualisierung, die es einführt. Ältere Geschichte bleibt in den bisherigen Protokollen je Bereich (Projektprotokoll, Story-Protokoll, Rechnungsereignisse).

2. Wer es sehen darf und wie du es einrichtest

Zwei Rollen dürfen das Protokoll lesen: der **RC Admin** und der **Auditor**. Teamleitungen und Projektleitungen sehen es nicht – es ist kein Management-Bericht. Der Auditor kann lesen, filtern, exportieren und die Integrität prüfen; ändern kann er nichts, auch nicht die Aufbewahrung. Aufbewahrung und Pseudonymisieren sind dem RC Admin vorbehalten.

Die Rolle Auditor vergeben

1. Öffne **Admin > Organisation** und den Benutzer, der lesen soll.
2. Füge in den Zusatzrollen (Kategorie *Administration*) die Rolle **Auditor** hinzu und speichere.
3. Der Benutzer sieht im Admin-Menü den Eintrag **Änderungsprotokoll**.

Die Vergabe der Rolle steht selbst im Protokoll (*Rollen geändert*).

3. Die Seite im Alltag

- **Filtern** nach Zeitraum, Wer, Was geschah, Suchtext und Einträgen je Seite (25 bis 200). Beim Feld *Was geschah* genügt der Anfang einer Aktion: `user.` zeigt alles zu Benutzern, `task.done` nur erledigte Aufgaben, `auth.login_failed` fehlgeschlagene Anmeldungen.
- **Blättern** mit *Ältere Einträge* und *Neuere Einträge*. Die Seite bleibt schnell, wie groß das Protokoll auch wird.
- **Exportieren** als CSV oder JSON, was die aktuellen Filter zeigen (alles, nicht nur die sichtbare Seite). Der Export ist selbst ein Eintrag. Zellen, die in einer Tabellenkalkulation wie eine Formel wirken würden, werden entschärft.
- Die Spalte **Einzelheiten** zeigt Änderungen als *Feld: alt → neu*.

Typische Fragen

Frage	So findest du es
Wer hat dem Benutzer die Finance-Rolle gegeben?	Was geschah: <code>user.roles_changed</code> , Suche: Benutzername
Wer hat das Fälligkeitsdatum dieser Aufgabe geändert?	Suche: Aufgabentitel (oder Objekt-Nummer), Details zeigen alt und neu
Wer hat die Rechnung heruntergeladen?	Was geschah: <code>export.</code>
Wann wurde diese Einstellung geändert?	Was geschah: <code>settings.</code> , Details nennen das Feld
Gab es Anmeldeversuche, die nicht geklappt haben?	Was geschah: <code>auth.login_failed</code> ; ein Benutzername erscheint nur, wenn das Konto existiert
Was hat dieser API-Token getan?	Wer: der Name des Token-Besitzers; die Spalte <i>Wer</i> nennt darunter den Zugang (REST-API oder MCP)

4. Manipulationen erkennen

Jeder Eintrag entsteht in derselben Datenbanktransaktion wie die Änderung selbst: Wird die Änderung zurückgerollt, verschwindet auch ihr Eintrag. Etwa eine halbe Minute später **versiegelt** der Scheduler neue Einträge: Er gibt jedem eine Position und berechnet einen Schlüssel-Hash über den Eintrag und den Hash des vorigen. So bilden alle Einträge eine **Kette**. Wird später ein Eintrag geändert, entfernt oder eingefügt, passt die Kette ab dieser Stelle nicht mehr.

Zusätzlich verweigert die Datenbank UPDATE und DELETE auf versiegelten Einträgen, wo sie das Anlegen dieses Schutzes erlaubt (die Seite zeigt es an), und der Scheduler legt **einmal am Tag eine signierte Ankerkopie** im S3-Speicher ab: „An Position N hatte die Kette den Hash H.“ Wer Datenbank *und* Schlüssel hat und die ganze Kette neu schreibt, kommt an diese Kopien nicht heran – der Vergleich fällt auf.

Integrität prüfen

Der Knopf **Integrität prüfen** auf der Seite berechnet die ganze Kette neu und vergleicht die Ankerkopien. Er liest das gesamte Protokoll; nutze ihn, wenn es darauf ankommt, nicht jede Minute. Mögliche Ergebnisse:

Meldung	Bedeutung	Was du tust
Die Kette ist unversehrt	Alle Einträge passen zueinander, und die Ankerkopien stimmen mit der Kette überein.	Nichts.
... wurde geändert	Ein Eintrag oder sein Vorgänger wurde nach dem Versiegeln verändert.	Sichere den Zustand, wende dich an euren Sicherheitsverantwortlichen und an Ontaric. Die Meldung nennt die Position des ersten betroffenen Eintrags.
... ein Eintrag fehlt	Zwischen zwei Einträgen fehlt einer.	Wie oben.
... Verbindung passt nicht / Einträge am Ende fehlen	Die Reihenfolge wurde verändert, oder die jüngsten Einträge wurden entfernt.	Wie oben.
... mit einem anderen Schlüssel versiegelt	Für Einträge fehlt der Schlüssel, mit dem sie versiegelt wurden (die Meldung nennt seine Kennung). Meist wurde der Bucket oder die Konfiguration gewechselt.	Füge den Schlüssel hinzu, den du heruntergeladen hast (Karte <i>Aufbewahrung und Datenschutz, Älteren Schlüssel hinzufügen</i>). Ist er verloren, wähle <i>Als verloren akzeptieren</i> (Abschnitt 5).

Ankerkopie ... hat sich geändert / fehlt / Signatur falsch	Das Protokoll wurde nach dem Erstellen der Kopie umgeschrieben, oder die Kopie selbst wurde verändert.	Wie bei „wurde geändert“.
--	--	---------------------------

In der Kopfzeile der Seite siehst du, wie viele Einträge **versiegelt** sind und wie viele noch **warten**. Warten Einträge länger als ein paar Minuten, läuft der Scheduler nicht.

5. Betrieb: Schlüssel, Speicher, Sicherung

Was Resourcecloud braucht

Baustein	Wofür
Scheduler-Container	Versiegelt Einträge (jede Minute), legt täglich die Ankerkopie ab und entfernt Einträge nach Ablauf der Aufbewahrung. Läuft er nicht, bleibt das Protokoll lückenlos, aber unversiegelt, und es gibt keine Ankerkopien.
S3-Dateispeicher	Hier liegen der Prüf Schlüssel (<Präfix>/audit/audit.key), ältere Schlüssel, die ihr hinzufügt (<Präfix>/audit/keys/), und die Ankerkopien (<Präfix>/audit/anchors/). Der Speicher bleibt über Neustarts, Updates und mehrere Replikate hinweg erhalten – anders als die Platte eines Containers.
Datenbank	Enthält das Protokoll selbst.

Du musst nichts konfigurieren: Beim ersten Gebrauch legt Resourcecloud den Schlüssel im Bucket an. Alle Container (Web, Scheduler, Replikate) lesen denselben Schlüssel.

Ohne S3-Dateispeicher fällt Resourcecloud auf eine Datei auf der Platte des Servers zurück. Bleibt diese Platte nicht erhalten (Container ohne Volume, mehrere Replikate), fängt jeder Neustart einen neuen Schlüssel an, und Einträge, die davor versiegelt wurden, lassen sich nicht mehr prüfen. Die Seite zeigt dann eine Warnung. Richte den S3-Speicher ein, oder setze den Schlüssel selbst (nächster Absatz).

Den Schlüssel herunterladen und aufbewahren

Sobald Resourcecloud den Schlüssel angelegt hat, erscheint oben auf der Seite ein Hinweis: „**Audit-Schlüssel herunterladen und sicher aufbewahren**“. RC Admin und Auditor können den Schlüssel dort als kleine Textdatei herunterladen (resourcecloud-audit-key-<Kennung>.txt). Er steht außerdem immer als Link *Schlüssel herunterladen* im Bereich *Integrität* bereit.

- **Bewahre die Datei dort auf, wo ihr wichtige Geheimnisse aufbewahrt** (Passwortmanager, Tresor), nicht nur auf dem Server und nicht im Datenbank-Backup.
- Klicke danach als RC Admin auf „**Ich habe ihn gesichert**“. Der Hinweis verschwindet, und die Seite zeigt, wer die Kopie bestätigt hat. Du bist als RC Admin dafür verantwortlich, dass es diese Kopie gibt.
- Jeder Download und jede Bestätigung steht im Protokoll (ohne den Schlüssel selbst).
- Wer den Schlüssel hat, kann die Kette prüfen – und auch nachbauen. Gib die Datei deshalb nur an Personen, die das dürfen.

Den Schlüssel selbst setzen (optional)

Wenn der Schlüssel nicht im selben Bucket wie die Dateien liegen soll, setze ihn in den Geheimnissen deiner Bereitstellung: die Umgebungsvariable `RC_AUDIT_KEY` (mindestens 32 zufällige Zeichen, für alle Container gleich). Sie hat Vorrang vor allem anderen. Dann liegt der Schlüssel dort, wo du auch die Datenbank-Zugangsdaten verwaltest, und nicht mehr im Bucket.

Sicherung und Wiederherstellung

- **Sichere Datenbank und Bucket zusammen.** Die Datenbank enthält das Protokoll, der Bucket den Schlüssel und die Ankerkopien. Versionierung oder Replikation des Buckets, wie ihr sie für Dateien ohnehin nutzt, genügt.
- **Datenbank und Bucket wiederherstellen:** Die Prüfung funktioniert weiter, der Schlüssel kommt aus dem Bucket.
- **Nur die Datenbank in einen neuen, leeren Bucket wiederherstellen:** Es entsteht ein neuer Schlüssel, und neue Einträge werden damit versiegelt. *Integrität prüfen* meldet dann, dass der Schlüssel der älteren Einträge fehlt (mit seiner Kennung). Füge ihn als RC Admin unter *Aufbewahrung und Datenschutz > Älteren Schlüssel hinzufügen* ein (füge die ganze heruntergeladene Datei ein oder nur den Schlüssel aus der Zeile „Key“). Danach lässt sich die ganze Kette wieder prüfen, über beide Schlüssel hinweg.
- **Schlüssel verloren** (keine Kopie mehr): Es geht kein Eintrag verloren; sie bleiben les- und exportierbar. Wähle nach der Meldung **„Als verloren akzeptieren“** und bestätige. Danach werden diese Einträge nicht mehr inhaltlich geprüft, aber weiter auf Reihenfolge und Verbindungen: Wird einer entfernt oder eingefügt, fällt es auf. Die Prüfung nennt, wie viele Einträge nur so geprüft wurden. Wer es akzeptiert hat, steht im Protokoll. Fügt du den Schlüssel später doch noch hinzu, werden die Einträge wieder vollständig geprüft.

Ankerkopien besser schützen

Je schwerer die Ankerkopien zu verändern sind, desto mehr leisten sie. Nutze für das Präfix `audit/anchors/` die **Objektsperre (Object Lock)** deines S3-Anbieters oder zumindest Versionierung, und gib die Zugangsdaten des Buckets nicht an die Datenbankadministration.

Speicherbedarf und Last

Ein Eintrag braucht rund 1 KB; eine Million Einträge belegen etwa 1 GB Datenbank. Das Schreiben eines Eintrags kostet weniger als eine Millisekunde, das Versiegeln läuft im Scheduler und belastet keine Anfrage.

6. Aufbewahrung und Datenschutz

RC Admins sehen auf der Seite die Karte **Aufbewahrung und Datenschutz**.

- **Einträge aufbewahren (Jahre):** Standard 10, der übliche Zeitraum für Geschäftsunterlagen; **0 bewahrt unbegrenzt auf**. Einmal am Tag entfernt der Scheduler ältere Einträge vom ältesten Ende her; der Rest bleibt prüfbar. Das Entfernen und jede Änderung der Einstellung sind selbst Einträge.
- **Person pseudonymisieren** (für ein Löschersuchen): Gib den Benutzernamen ein und bestätige. Danach zeigt jeder Eintrag von oder über die Person ein Pseudonym wie pseudonym-ab12cd34. Dieselbe Person bekommt (solange der Schlüssel derselbe bleibt) dasselbe Pseudonym, die Einträge gehören also weiter zusammen, der Name lässt sich aber nicht zurücklesen. Das **lässt sich nicht rückgängig machen**, funktioniert auch für gelöschte Konten, und die Prüfkette bleibt gültig. Vorname, Nachname und E-Mail-Adresse stehen ohnehin nicht im Klartext in den Einträgen.

Was personenbezogen ist

Das Protokoll enthält Benutzernamen und IP-Adressen der Handelnden. Es zeichnet **Änderungen** auf, nicht, was jemand angesehen hat, und nicht die Arbeitszeit. Weise in eurem Verzeichnis der Verarbeitungstätigkeiten darauf hin, nenne den Zweck (Nachvollziehbarkeit und Sicherheit von Änderungen) und die Aufbewahrungszeit.

Betriebsrat: Informiere den Betriebsrat, dass es das Protokoll gibt, wer es öffnen kann und wie lange es aufbewahrt wird. Wie Resourcecloud mit der Auswertung einzelner Beschäftigter umgeht, steht im *Betriebsrats-Paket*. Diese Anleitung ist keine Rechtsberatung.

7. Schnittstellen

Das Protokoll lässt sich nur **lesen**; keine Schnittstelle kann hineinschreiben. Der Zugriff folgt derselben Regel wie die Seite: Der Benutzer, dem der API-Token gehört, muss RC Admin oder Auditor sein. Ein Token mit Lesebereich genügt.

Aufruf	Zweck
GET /api/v1/audit-log	Einträge, neueste zuerst, mit denselben Filtern wie die Seite; blättern mit next und prev.
GET /api/v1/audit-log/verify	Anzahlen, Aufbewahrung, Ankerstatus und das Ergebnis der Prüfung. Liest das ganze Protokoll.

Dieselben Abfragen gibt es als MCP-Werkzeuge (`list_audit_log`, `verify_audit_log`). Die vollständige Beschreibung steht in der API-Dokumentation. Eine Weiterleitung des Protokolls an ein SIEM-System gibt es noch nicht; bis dahin lässt sich der JSON-Export oder die Abfrage regelmäßig abholen.

8. Prüfung durch Dritte

Eine Prüferin muss dem Knopf nicht vertrauen. Jeder versiegelte Eintrag trägt im JSON-Export und in der API-Antwort einen Block chain mit den Rohwerten, über die der Hash gebildet wurde. Für den Eintrag an Position n und den Hash des vorigen Eintrags p (64 Nullen beim ersten):

Hash = HMAC-SHA256(Schlüssel, $p + "|" + \text{JSON-Array}$), das Array aus diesen Feldern in dieser Reihenfolge: `id`, `seq`, `at`, `actor_id`, `actor_type`, `via`, `ip`, `request_id`, `action`, `object_type`, `object_id`, `team_id`, `project_id`, `changes`, `meta`, `restricted_to` (jeweils als Text, null bleibt null; JSON ohne Maskieren von Schrägstrichen und Nicht-ASCII-Zeichen). Namen von Handelndem und Objekt gehören **nicht** zum Hash – deshalb lässt sich eine Person pseudonymisieren, ohne die Kette zu brechen. Den Schlüssel erhält die Prüferin von euch. Eine ausführlichere technische Beschreibung liegt in der Anleitung *audit-log-guide*.

9. Grenzen

- Die Kette macht Änderungen **erkennbar**, sie verhindert sie nicht: *manipulationssicher nachweisbar*, nicht *unveränderbar*. Mit den Ankerkopien im geschützten Bucket ist auch das vollständige Umschreiben durch jemanden mit Datenbank und Schlüssel erkennbar.
- Einträge der letzten halben Minute sind noch nicht versiegelt.
- Was ein Browser selbst exportiert (die Export-Knöpfe der Tabellen auf Berichtsseiten), erreicht den Server nie und steht nicht im Protokoll. Downloads, die der Server erzeugt (Rechnungen, Passwörter eines Sammelimports, das Protokoll), schon.
- Änderungen direkt in der Datenbank, an Resourcecloud vorbei, werden nicht aufgezeichnet; ein fehlender oder veränderter Eintrag in der Kette und die Ankerkopien zeigen aber, dass das Protokoll angefasst wurde.
- Das Protokoll hält nicht fest, was jemand nur angesehen hat.
- Wird die Datenbank direkt manipuliert, bevor der Scheduler die Einträge versiegelt hat, lässt sich dieses Fenster von höchstens einer Minute nicht schützen.

10. Häufige Fragen

Kann ein RC Admin Einträge löschen oder ändern?

Nein. Es gibt keine Funktion dafür. Wer es direkt in der Datenbank versucht, bricht die Kette; die Prüfung nennt die Stelle.

Warum sehe ich meine eigene Anmeldung von heute Morgen nicht?

Anmeldungen werden ab der Aktualisierung aufgezeichnet, die das Protokoll einführt. Wer vorher angemeldet war und angemeldet blieb, taucht erst nach der nächsten Anmeldung auf.

Warum steht bei einigen Einträgen ein Schloss?

Das sind Einträge zu privaten Aufgaben; nur die Beteiligten sehen sie.

Warum „wartet“ ein Teil der Einträge auf das Siegel?

Der Scheduler versiegelt jede Minute Einträge, die älter als etwa eine halbe Minute sind. Bleiben welche länger liegen, läuft der Scheduler nicht.

Müssen wir den Schlüssel sichern?

Ja: Lade ihn herunter, wenn Resourcecloud dazu auffordert, und bewahre ihn bei euren Geheimnissen auf (Abschnitt 5). Er liegt zusätzlich im S3-Bucket und wird mit ihm gesichert. Setzt ihr RC_AUDIT_KEY selbst, gehört er ohnehin zu euren Geheimnissen.

Wir haben den Schlüssel verloren. Was nun?

Du brauchst keinen Support: *Integrität prüfen* nennt die Kennung des fehlenden Schlüssels. Füge ihn hinzu, wenn du eine Kopie findest, oder wähle *Als verloren akzeptieren*. Neue Einträge werden in jedem Fall mit dem aktuellen Schlüssel versiegelt und geprüft.

Was passiert, wenn wir den Bucket wechseln?

Kopiere das Präfix `audit/` mit, oder setze den Schlüssel als `RC_AUDIT_KEY`. Sonst entsteht ein neuer Schlüssel (siehe „Sicherung und Wiederherstellung“).

Können wir das Protokoll abschalten?

Nein. Es ist Teil der Plattform. Ihr bestimmt aber, wie lange Einträge aufbewahrt werden.

Kontakt

Ontaric GmbH

Jürgen-Töpfer-Straße 113

22763 Hamburg

Anfragen: datenschutz@resourcecloud.io

Stand 11.10.2026. Der Inhalt entspricht dem Stand des Produkts zu diesem Datum und ist, soweit technisch möglich, durch automatische Tests belegt. Keine Rechtsberatung.