

Verfahrensdokumentation

Beschreibt, wie AuditMail Server E-Mails archiviert, sichert und löscht. Gedacht als Grundlage für die Verfahrensdokumentation nach GoBD Rz. 151 ff., die einem Betriebsprüfer vorzulegen ist.

Kein Rechtsrat. Ob Ihre Archivierung den steuer- und handelsrechtlichen Anforderungen genügt, beurteilt Ihr Steuerberater. Dieses Dokument beschreibt die technische Arbeitsweise — es ersetzt weder die Prüfung des Einzelfalls noch eine Zertifizierung, die es für diese Software nicht gibt. Der Abschnitt Grenzen benennt, was das Verfahren *nicht* leistet.

1. Gegenstand

Archiviert werden E-Mails, die als Handels- oder Geschäftsbrief gelten (§ 257 Abs. 1 HGB) oder Buchungsbelege sind (§ 147 Abs. 1 AO) — also die gesamte geschäftliche Korrespondenz, ein- und ausgehend, aus allen angebundenen Postfächern.

Nicht archiviert werden Ordner, die beim jeweiligen Konto ausgenommen sind; üblich sind Spam und Papierkorb.

2. Erfassung

Quelle sind die eingerichteten IMAP-Postfächer. Jedes Konto steht unter `Einstellungen → E-Mail-Konten` mit eigenem Zugang, Zeitfenster und Ausschlussliste; die Zugangsdaten liegen mit AES-256-GCM verschlüsselt in der Datenbank. Jeder archivierte Eintrag trägt die Kennung seines Herkunftskontos (`konto_id`) und den Ordner in `quelle`, ist also jederzeit einem Postfach zuzuordnen.

Der Abruf öffnet jeden Ordner ausschließlich lesend (`EXAMINE`) und liest Nachrichten mit `BODY.PEEK[]`. Im Postfach wird dadurch nichts verändert — keine Mail wird als gelesen markiert, verschoben oder gelöscht.

Wird ein Konto später entfernt, bleiben seine bereits archivierten Mails erhalten; sie sind Teil der Hash-Kette und dürfen nicht verschwinden.

Zeitpunkt ist der Abruf, nicht der Empfang. Zeitgesteuert nach dem eingestellten Intervall — voreingestellt 30 Minuten, änderbar unter `Einstellungen → Automatischer Betrieb` — oder von Hand. Angestoßen wird der Abruf von einem einzigen Cronjob (`cron.php`) im Viertelstundentakt; ob ein Abruf fällig ist, entscheidet die Anwendung selbst.

Umfang je Durchlauf begrenzt durch das Zeitfenster (Standard 90 Tage rückwärts) und die Höchstzahl neuer Mails (Standard 50). Beides ist eine Vorsichtsmaßnahme gegen die Laufzeitgrenze des Webserver, kein inhaltlicher Filter: wiederholte Durchläufe arbeiten den Rückstand ab.

Dublettenschutz über die `Message-ID` der Nachricht (RFC 5322 § 3.6.4), abgeglichen gegen den Index. Ein zweiter Abruf derselben Mail erzeugt keinen zweiten Eintrag. Zusätzlich merkt sich die Anwendung abgeholte IMAP-UIDs, was allein aber nicht ausreicht — UIDs können sich beim Verschieben ändern.

Selbst gesendete Mails können sendende Systeme über die Schnittstelle `archive` direkt übergeben, unabhängig davon, ob sie im Postfach liegen.

3. Ablage

Jede Mail wird **bytwweise unverändert** als `.eml` gespeichert:

data/mail/JJJJ/MM/<kennung>.eml	Originalnachricht, RFC 5322
data/mail/JJJJ/MM/<kennung>.json	Metadaten, auch ohne Datenbank lesbar

Die Datei wird einmal geschrieben und danach nie mehr angefasst. Anzeige, Suche und Auswertung entstehen durch Parsen zur Laufzeit — das Original bleibt Original, einschließlich aller Kopfzeilen, Signaturen und Verschlüsselungs- Hüllen.

Ein Index in der Datenbank hält Absender, Empfänger, Betreff, Datum, Kategorie, Referenz, Anhangsnamen, Hashes, Frist und Herkunft.

4. Unveränderbarkeit

Der Kern des Verfahrens ist eine Hash-Kette. Beim Archivieren wird berechnet:

```
dateiHash = SHA256( Inhalt der .eml-Datei )
eintrag   = SHA256( dateiHash || hash des Vorgängereintrags )
```

Der erste Eintrag verwendet 64 Nullen als Vorgänger. Gespeichert werden `hash` und `vorgaenger_hash`.

Daraus folgt:

- **Inhaltsänderung** — der Dateihash ändert sich, `hash` passt nicht mehr
- **Eintrag entfernt** — die Verkettung des Nachfolgers bricht
- **Eintrag eingeschoben** — der Vorgänger-Hash passt nicht in die Kette
- **Kette neu gerechnet** — auffällig, weil der Vorgang im Audit-Log steht

Die Reihenfolge ergibt sich aus der Spalte `seq` (fortlaufend, vom Datenbanksystem vergeben), nicht aus einem änderbaren Datum.

Beim Einfügen läuft eine Transaktion mit Zeilensperre (`FOR UPDATE`) auf dem letzten Eintrag. Zwei gleichzeitige Abrufe können die Kette dadurch nicht gabeln.

Prüfung über *Integrität* in der Oberfläche oder `verify` in der API: für jeden Eintrag wird der Dateihash neu berechnet und gegen den gespeicherten Wert sowie den Vorgänger geprüft. Das Ergebnis nennt jeden abweichenden Eintrag einzeln.

5. Nachvollziehbarkeit

Protokolliert werden mit Zeitpunkt (Mikrosekunden), Benutzer und IP-Adresse:

Vorgang	Auslöser
ARCHIVIERT	Mail aufgenommen
ANGESEHEN	Mail geöffnet
ANHANG-GEOEFFNET	Anhang abgerufen
EML-EXPORT	Originaldatei heruntergeladen
ZUGEORDNET	Kategorie oder Referenz geändert
INTEGRITAETSPRUEFUNG	Prüfung ausgeführt, mit Ergebnis
GELÖSCHT	Eintrag nach Fristablauf entfernt
HASH-KETTE-NEU	Kette neu berechnet
EINSTELLUNGEN-GEAENDERT	Konfiguration geändert
ANMELDUNG, ANMELDUNG-FEHLGESCHLAGEN, ABMELDUNG	Zugriff
BENUTZER-ANGELEGT, BENUTZER-GEAENDERT, BENUTZER-GELOESCHT	Rechteverwaltung
IMAP-ABRUF	Abruf, der etwas archiviert hat oder auf ein Problem stieß
IMAP-FEHLER, IMAP-WIEDER-ERREICHBAR	Postfach nicht abholbar, und ab wann wieder
KETTENANKER	Kettenkopf nach außen gegeben, mit Empfänger
LOESCHSPERRE-GESETZT, LOESCHSPERRE-AUFGEHOBEN	Sperre wegen laufender Prüfung, mit Anlass
LOESCHUNG-ABGEWIESEN	Löschversuch bei gesetzter Sperre
CRON-ABGEWIESEN, CRON-FEHLER	Zugriffsversuch, Störung

IMAP-FEHLER wird beim Eintritt der Störung geschrieben, nicht bei jedem Folgelauf: Bei einem Viertelstundentakt ergäbe ein dauerhaft nicht erreichbares Postfach fast hundert gleichlautende Zeilen am Tag. Der Eintrag beim Wechsel und die Gegenbuchung **IMAP-WIEDER-ERREICHBAR** grenzen den Zeitraum ein, in dem nicht abgeholt wurde; der aktuelle Zustand steht am Konto (`letzter_status`). Ergebnislose Durchläufe erzeugen aus demselben Grund keinen Eintrag — dass sie stattfanden, hält `letzter_abruf` fest.

Doppelt geführt: in der Datenbank (auswertbar) und als Textdatei `data/logs/audit.log` (lesbar, auch wenn die Datenbank ausfällt).

6. Aufbewahrung und Löschung

Kategorie	Frist	Grundlage
Rechnung, Gutschrift, Mahnung	8 Jahre	§ 147 Abs. 3 AO (BEG IV, ab 01.01.2025; vorher 10)
Angebot, Auftrag, Vertrag, Allgemein, ohne Kategorie	6 Jahre	§ 257 Abs. 4 HGB

Die Frist beginnt mit dem **Schluss des Kalenderjahres**, in dem die Mail entstanden ist (§ 147 Abs. 4 AO) — nicht am Tag der Archivierung. Der Löschtermin wird bei der Aufnahme berechnet und bei jeder Änderung der Kategorie neu.

Gelöscht wird nur auf ausdrückliche Anweisung eines Administrators (*Frist-Löschung*), nie automatisch. Entfernt werden Originaldatei und Indexeintrag; der Audit-Eintrag über die Löschung bleibt. Anschließend wird die Kette neu aufgebaut, weil die planmäßige Löschung Lücken hinterlässt.

Rechtsgrundlage der Löschung ist Art. 17 DSGVO, sobald die Aufbewahrungspflicht endet — bis dahin steht ihr Art. 17 Abs. 3 lit. b DSGVO entgegen.

Die Frist ist bewusst nicht überschritten worden, wenn ein Eintrag als „löschar“ angezeigt wird: die Anzeige ist ein Vorschlag, die Entscheidung liegt beim Verantwortlichen. Bei laufenden Verfahren, Gewährleistungsfällen oder einer Außenprüfung (§ 147 Abs. 3 Satz 5 AO) ist die Löschung zurückzustellen.

Löschsperre. Für diesen Fall gibt es einen Schalter unter `Einstellungen → Nachweis`: Ist er mit Anlass gesetzt, verweigert die Frist-Löschung den Dienst, statt nur zu warnen. Der Versuch wird als `LOESCHUNG-ABGEWIESEN` protokolliert, Setzen und Aufheben mit Anlass als `LOESCHSPERRE-GESETZT` bzw. `LOESCHSPERRE-AUFGEHOBBEN`. Die Prüfung sitzt in `archive_purge()` selbst, nicht in der Schnittstelle — kein Aufrufweg umgeht sie. Wahlweise lässt sich die Sperre in der Konfiguration verankern (`'pruefung_aktiv' => true`); dann kann sie über die Oberfläche nicht aufgehoben werden.

6a. Kontrollen

Die GoBD verlangen ein internes Kontrollsystem mit **durchgeführten** Kontrollen (Rz. 100 ff.). Dafür gibt es den Kontrolllauf, einmal täglich ab der eingestellten Stunde, angestoßen über denselben Cronjob wie der Abruf (einzeln erreichbar als `cron_verify.php`):

- **Integritätsprüfung** über den gesamten Bestand, Ergebnis als `INTEGRITAETSPRUEFUNG` im Protokoll. Findet er Abweichungen oder fehlende Dateien, antwortet er mit HTTP 409, damit ein externer Aufrufdienst Alarm auslöst und der Befund nicht in einem stillen Erfolg untergeht. Der Befund selbst steht als `BEFUND-GEMELDET` im Protokoll, sein Ende als `BEFUND-BEHOBEN`; das Ergebnis jeder Prüfung wird unabhängig davon festgehalten, auch wenn der Alarm nach außen nur bei der Flanke ausgelöst wird.
- **Kettenanker:** Der Kettenkopf — der Hash des letzten Eintrags, der rekursiv den gesamten Bestand abdeckt — wird in `am_anker` festgehalten und an eine Adresse außerhalb dieses Servers gesendet. Dort liegt er mit dem Empfangszeitstempel eines Dritten.

Der Anker ist die Antwort auf die Grenze „wer die Datenbank kontrolliert, kann die Kette neu schreiben“: Ein nachträglich umgeschriebener Bestand ergibt einen anderen Kopf als der, der beim Empfänger liegt. Ein Bruch der Ankerfolge ist dabei nicht automatisch verdächtig — eine fristgerechte Löschung baut die Kette neu auf und verändert den Kopf zwangsläufig. Aussagekräftig ist der Bruch **ohne** zugehörigen Protokolleintrag (`GELÖSCHT`, `HASH-KETTE-NEU`).

7. Zugriffsschutz

- Jeder Zugriff erfordert eine Anmeldung, auch der lesende
- Rollen: `admin` (alles) und `pruefer` (lesen, suchen, prüfen, zuordnen)
- Löschung, Kettenneuaufbau, Einstellungen und Benutzerverwaltung nur `admin`
- Passwörter als bcrypt-Hash; Sperre nach Fehlversuchen je Konto und IP
- Sitzungscookie `HttpOnly`, `Secure`, `SameSite=Strict`, neue ID bei Anmeldung
- Archivbestand, Programmbibliothek und Konfiguration per `.htaccess` gesperrt
- IMAP-Zugangsdaten mit AES-256-GCM verschlüsselt, nie im Browser gespeichert

8. Maschinelle Auswertbarkeit

GoBD Rz. 126 ff. verlangt, dass Daten in maschinell auswertbarer Form vorliegen. Erfüllt durch:

- `.eml` im offenen Standardformat, lesbar mit jedem Mailprogramm
- unverschlüsselte Ablage der Inhalte — die Prüfung braucht diese Software nicht
- Download einzelner Originaldateien über die Oberfläche
- Metadaten als JSON neben jeder Mail und in der Datenbank
- Volltextsuche über Kopfdaten, Referenz und Anhangsnamen

9. Datensicherung

Zum Archiv gehören zwei Teile, die nur gemeinsam vollständig sind:

1. `data/mail/` — die Originaldateien
2. die Datenbanktabellen — Index, Hash-Kette, Protokoll

Beide müssen im selben Durchgang gesichert werden. Eine Sicherung nur eines Teils macht die Integritätsprüfung wertlos: fehlen Dateien, meldet die Prüfung „Datei fehlt“; fehlt der Index, ist die Kette verloren.

Meldet die Prüfung nach einer Wiederherstellung fehlende Dateien, ist die richtige Reaktion das Ergänzen aus der Sicherung — **nicht** der Neuaufbau der Kette. Der würde den Verlust verdecken.

Reihenfolge

Ohne Kommandozeilenzugang läuft die Sicherung üblicherweise als Dateiabzug per FTP. Der erfasst `data/mail/` — die Datenbank nicht. Dafür gibt es `cron_backup.php`: ein täglicher Lauf, der einen SQL-Abzug der Tabellen dieser Installation nach `data/backup/` schreibt, wo der Dateiabzug ihn mitnimmt.

Die Reihenfolge ist nicht beliebig:

1. `cron_backup.php` — Datenbankabzug
2. **danach** der Dateiabzug

`archive_store()` schreibt zuerst die `.eml` und trägt sie danach in den Index ein. Sichert man in dieser Reihenfolge, ist eine dazwischen eingegangene Mail höchstens eine Datei ohne Indexeintrag — folgenlos, weil die Prüfung über den Index läuft. Andersherum entstünde ein Indexeintrag ohne Datei, und die Prüfung meldete nach jeder Wiederherstellung „Datei fehlt“.

Alte Stände räumt der Lauf selbst auf (Standard: 14, über `backup_behalten` einstellbar). Das ist kein Komfort, sondern Notwehr: Ein volllaufender Webserver legt auch den Abruf still.

Der Anwendungsschlüssel gehört dazu

Postfach-Passwörter und das Cron-Token stehen verschlüsselt in der Datenbank; der Schlüssel dafür (`app_key`) steht in der Konfigurationsdatei. Ein Datenbankabzug allein gibt sie also nicht preis — für eine vollständige Wiederherstellung muss die Konfigurationsdatei aber mitgesichert werden, und zwar an einem Ort, an dem sie **nicht** neben dem Datenbankabzug liegt.

Wenn ein Dienstleister betreibt

Wird die Anwendung nicht vom Steuerpflichtigen selbst betrieben, sondern für ihn gehostet, verschiebt das die Verantwortung nicht:

- **Gegenüber der Finanzverwaltung bleibt der Steuerpflichtige verantwortlich.** § 147 AO verpflichtet ihn, nicht seinen Dienstleister. Eine Auslagerung ändert daran nichts; sie ist zulässig, entbindet aber nicht.

- **Der Betreiber ist Auftragsverarbeiter** nach Art. 28 DSGVO. Dafür braucht es einen Vertrag, der Sicherung, Wiederherstellung, Löschung und die Rückgabe des Bestands bei Vertragsende regelt.
- **Die Sicherung ist die Hauptleistungspflicht**, nicht eine Nebenpflicht. Wer ein Archiv betreibt, das die gesetzliche Aufbewahrung eines anderen trägt, kann sich für deren Verlust nicht wirksam freizeichnen.

Daraus folgen drei praktische Punkte, die in dieselbe Richtung wirken:

1. **Eine eigene Sicherung, unabhängig vom Host.** Dessen Sicherung hilft bei dessen Ausfall — nicht bei einem versehentlich gelöschten Verzeichnis, das erst Wochen später auffällt.
2. **Eine Kopie beim Auftraggeber.** Ein regelmäßiger ZIP-Export in seine Hand ist der wirksamste Einzelschritt: Er verlegt einen Teil des Risikos dorthin, wo die gesetzliche Pflicht ohnehin liegt.
3. **Der Kettenanker an einen Dritten.** Nach einer Wiederherstellung lässt sich damit belegen, dass der Bestand derselbe ist wie vorher. Ohne ihn steht Aussage gegen Aussage.

Das offene Ablageformat wirkt in dieselbe Richtung: `.eml` und JSON sind ohne diese Anwendung lesbar (Abschnitt 8). Ein wiederhergestellter Bestand bleibt auswertbar, auch wenn die Software des Betreibers nicht mehr existiert.

Die Verteilung der Verantwortung ist hier fachlich beschrieben, nicht rechtlich beraten. Vertragstexte, Haftungsgrenzen und Versicherungsschutz gehören in die Hand eines Anwalts.

Grenzen

Was dieses Verfahren nicht leistet:

Kein qualifizierter Zeitstempel. Die Kette belegt die Reihenfolge und Unverändertheit *relativ zueinander*, nicht den absoluten Zeitpunkt. Der Kettenanker (Abschnitt 6a) mildert das: Der Kopf liegt bei einem Dritten, mit dessen Empfangszeitstempel. Das ist ein Indiz, kein qualifizierter Zeitstempel. Wer gegenüber Dritten die Beweiswirkung nach Art. 41 eIDAS braucht, benötigt einen RFC-3161-Zeitstempel eines Vertrauensdienstes auf denselben Kettenkopf — vorgesehen, aber nicht umgesetzt.

Keine Zertifizierung. Für diese Software liegt kein Testat nach IDW PS 880 oder vergleichbar vor. Die GoBD verlangen das nicht, ein Prüfer kann es aber als Nachweis erwarten.

Wer die Datenbank kontrolliert, kann die Kette neu schreiben. Ein Administrator mit Datenbankzugriff kann Dateien ändern und die Hashes nachziehen. Die Kette schützt gegen unbemerkte Änderung im laufenden Betrieb und gegen Manipulation durch die Anwendung selbst — nicht gegen den, der den Server beherrscht.

Der Kettenanker (Abschnitt 6a) begrenzt das, sobald er eingerichtet ist: Die außerhalb verwahrten Köpfe kann derselbe Administrator nicht mehr ändern, ein umgeschriebener Bestand fällt beim Vergleich auf. Vollständig aufheben lässt es sich nur mit WORM-Speicher oder einer unabhängig geführten Zweitablage. Ohne eingetragenen Ankerempfänger bleibt es bei der ursprünglichen Einschränkung — die Anker liegen dann nur in derselben Datenbank.

Vollständigkeit hängt am Postfach. Archiviert wird, was zum Zeitpunkt des Abrufs im Postfach liegt. Vor der Inbetriebnahme gelöschte Mails, Nachrichten außerhalb des Zeitfensters und Mails in ausgeschlossenen Ordnern fehlen. Wer Vollständigkeit ab einem Stichtag zusichern will, muss den Erstbestand vollständig einlesen und darf im Postfach danach nichts mehr löschen, bevor es archiviert ist.

Verschlüsselte Mails bleiben verschlüsselt. S/MIME- und PGP-Nachrichten werden als Original gespeichert und sind damit aufbewahrt, aber im Archiv nicht im Klartext durchsuchbar.

Die Kategorieerkennung ist eine Heuristik. Sie erkennt die Nummernkreise RE-, ANG-, AU-, GS- und einige Schlüsselwörter. Was sie nicht erkennt, bleibt ohne Kategorie und erhält die kürzere Frist von 6 Jahren. Die Kennzahl *Ohne Zuordnung* im Archiv zeigt diese Einträge; sie gehören regelmäßig durchgesehen, damit Buchungsbelege nicht zu früh löschar werden.