

Authorisation of commit b7e3d2c19a4f in example- fs/payments-gateway

Peregrini

Prepared for Example Financial Services Pty Ltd

Prepared 2026-09-10 02:15:00 UTC

Control fields

Control objective	Changes to production code are authorised by an identified person and traceable to that authorisation (APRA CPS 230, controls over changes to critical operations and the records of them; CPS 234, information security controls and their testing; SOC 2 CC8.1 change management, CC6.1 logical access).
Evidence type	Authorisation certificate: a signed statement of what the Register of Dealings holds for one commit.
Period	2026-09-08 23:04:12 UTC to 2026-09-08 23:34:10 UTC (one agent session)
System	example-fs/payments-gateway, working copy /home/build/payments-gateway; launcher claude-code
Subject	commit b7e3d2c19a4f6e8d0c5b3a7f2e1d9c8b6a5f4e3d
Prepared by	Peregrini, from the Register of Dealings and the operator's record vault
Prepared for	Example Financial Services Pty Ltd
Prepared on	2026-09-10 02:15:00 UTC
Certificate	cert_7e2c4a6b-sample, signed ed25519 under Peregrini's notary key

Who authorised this change, and why

Commit b7e3d2c19a4f was made by efs-claude-code-ws0417, an agent of Example Financial Services Pty Ltd, in session claude-code: 4f0c2d9e-8b1a-4c3d-9e2f-6a7b8c9d0e1f, under a mandate issued by the operator's Clerk, on an instruction the operator gave in that session and later lodged in its record vault with this commit's sha.

Made under	The Clerk's mandate for the session a5268072a20a8d8b1451b5d3bdcd58a9ef2235418071a6705840fbfb9beef124 lodged 2026-09-08 23:04:12 UTC by efs-clerk (did:key:z6MkfMyPggt2ma6Zm4mPmUU8TUj7ngXikGn4saBWMvxAgN6) as kind order, receipt 6b1e9c3a-7d2f-4e8b-a0c4-5f6d7e8f9a0b
Accepted	2026-09-08 23:04:19 UTC, under the agent's own key; receipt 1c3e5a7b-9d0f-4a2c-b4e6-8f0a2c4e6a8b
Agent	efs-claude-code-ws0417 did:key:z6MkhcSYYGqwhZUsRa56q7gv1tQZGSi8mKGeqSisTPeCgF7H
Operator	Example Financial Services Pty Ltd, account acct_0f3a7c2e-sample

The instruction that authorised it

Yes, commit it on agent/CHG-0042. Do not push; I will open the PR myself after review.

Instruction 2 of the session, given 2026-09-08 23:31:02 UTC; digest c82696939ebb502d0fbb8355a116dcd5400c2f15f646dd8328c16d32c040a3a0 (HMAC-SHA256 under the account's vault key). Lodged in the vault naming commit b7e3d2c19a4f. It follows one earlier instruction in the session: instruction 1, given 2026-09-08 23:05:40 UTC: *Add the daily settlement reconciliation report under src/reports/settlement/. Read the ledger, do not write to it, and do not touch anything under db/ledger/. Ticket CHG-0042..*

Where the commit sits in the record

Chain leaf	call 8 of 9, 2026-09-08 23:31:40 UTC, tool Bash git add src/reports/settlement && git commit -m 'Daily settlement reconciliation report (CHG-0042)' leaf b9ac4c3236ee55eddeb73f8cad09b76d54c525daedafffb9a9d248cfcba45f08
Chain roots lodged	root 1, head of 1 calls, e73f8869ca609d37e8f0d1a1320af92d9e46b594a2691d6b9c5561f14dcae461, lodged 2026-09-08 23:05:53 UTC, receipt 9d2f4a6c-1e3b-4c5d-8f7a-2b4c6d8e0f1a root 2, head of 9 calls, 7ed19787eafcd55e54d7e13c1077264366693506fefff0f42ecd9f0953e3085, lodged 2026-09-08 23:34:10 UTC, receipt c4e6a8b0-3d5f-4a7b-9c1d-4e6f8a0b2c3d
Sequence	instruction 2 at 2026-09-08 23:31:02 UTC → the commit at 2026-09-08 23:31:40 UTC → the report at 2026-09-08 23:33:58 UTC. The instruction precedes the act; the act precedes the report that discloses it.

The operator's conditions

INSTRUCTION FILE	SHA-256 WHEN THE MANDATE ISSUED
/home/build/payments-gateway/CLAUDE.md	1ee4fe75ed8c89876384ce2dea9a9b1fce84f7ae0d599181f18e0e7da3a0db80
/home/build/payments-gateway/docs/CHANGE_CONTROL.md	02b399eb5c9e61f4ae92e8169f7813249f5f22ec2e3f0868bd6ab0ff88aa3ec4
/home/build/.claude/CLAUDE.md	4f03d8616e865a8e399f0aeb6036ee0cfb36cb0031b60a816e748ddc1bb58176

Covered by

Completion report	486dd8dfde6868b9cd21d3401f55a8f6423be8fe05ff7ce8c2c9ec8dfea8e3eb lodged 2026-09-08 23:33:58 UTC by efs-claude-code-ws0417, receipt e5a7c9b1-2f4d-4e6a-8b0c-1d3e5f7a9b1c. It reports the commit under <i>Done</i> and reports that nothing was pushed, merged or deployed.
Transcript seal	b5d5b828e783a096c7133af25fbf8d12c151bf68d53bd36141c13de88faf6430, sealed 2026-09-08 23:34:10 UTC, receipt a2c4e6f8-0b1d-4e3f-9a5b-7c9d1e3f5a7b
Anchor	db4d58923ecaee63c4e3400ca4c766834d686e711416ad0f9ec63b5cd4f9e733, confirmed in Bitcoin block 923,118; range to 2026-09-09 00:00:00 UTC

Standing of the agent

No complaint stands open against efs-claude-code-ws0417 on this session and no uncured declaration has been made against it, as at 2026-09-10 02:15:00 UTC.

A complaint is a notice the operator's Clerk lodges on the Register naming the session and the particular; a declaration is a finding published against the agent and read into every later mandate it signs until the shortfall is cured. This states what is on the record at the time above and nothing about the commit's merit.

Survival

This certificate does not depend on the transcript. The launcher deletes transcripts after thirty days; the mandate's hash, the acceptance, the chain roots, the report's hash and the transcript's own hash were lodged on the Register at the times shown, under receipts Peregrini signed, and the Register's anchors are committed to Bitcoin. The operator's copy of the chain file is enough to prove the commit's leaf against a lodged root; the operator's copy of the mandate document is enough to prove the conditions. Deleting the transcript removes the words; it removes nothing this certificate states.

How an auditor checks this

Every fact on this page is either a hash of bytes the operator holds or an entry on the Register that anyone may read back. An auditor checks it without asking Peregrini anything it has not already published.

1. Recompute the hash of an instruction file.

```
sha256sum /home/build/payments-gateway/CLAUDE.md
# expect 1ee4fe75ed8c89876384ce2dea9a9b1fce84f7ae0d599181f18e0e7da3a0db80
```

2. Fetch the mandate's lodgement by its hash and compare the receipt.

```
curl -s https://www.peregrini.ai/api/v1/notarise/a5268072a20a8d8b1451b5d3bdcd58a9ef2235418071a6705840fbfb9beef124
# expect receipt.id 6b1e9c3a-7d2f-4e8b-a0c4-5f6d7e8f9a0b, firstLodgedBy efs-clerk, firstLodgedAt 2026-09-08T23:04:12.000Z
```

3. Verify the receipts and this certificate. Ed25519 over the payload bytes, against the notary key at <https://www.peregrini.ai/.well-known/notary.json> (or the retired key current at the lodgement time, listed there as `retiredKeys`). Never trust a key only because a document carries it. This certificate's payload and signature are printed under *Certificate* below.

4. Compare the chain root, and prove the leaf.

From the operator's copy of
~/.peregrini/chain/claude-code/4f0c2d9e-8b1a-4c3d-9e2f-6a7b8c9d0e1f.jsonl: line 8 must hash to the leaf above, each line's hash must be the next line's prev, and the last line must hash to the lodged root.

```
sed -n '8p' ~/.peregrini/chain/claude-code/4f0c2d9e-8b1a-4c3d-9e2f-6a7b8c9d0e1f.jsonl | tr -d '\n' | sha256sum
# expect b9ac4c3236ee55eddeb73f8cad09b76d54c525daedafffb9a9d248cfcba45f08
tail -n 1 ~/.peregrini/chain/claude-code/4f0c2d9e-8b1a-4c3d-9e2f-6a7b8c9d0e1f.jsonl | tr -d '\n' | sha256sum
# expect 7ed19787eafcd55e54d7e13c1077264366693506fefff0f42ecdf9f0953e3085
```

5. Ask for the same certificate again. The operator's account can re-run the trace at any time; the answer is signed, and a second answer that differs from this one is itself evidence.

```
curl -s https://www.peregrini.ai/api/v1/accounts/me/certificates/commit/b7e3d2c19a4f6e8d0c5b3a7f2e1d9c8b6a5f4e3d?repo=example-fs%2Fpayments-gateway -H 'Authorization: Bearer ak_...'
```

6. Check this document. Its id in the footer is the SHA-256 of the page between the header and the footer.

Sample: the receipts on this page are signed under the sample notary key
0d7550754e0800a5d237eef5826035766b9b3e5a15868a940ab289958788e3b0, published here so
the signatures can be checked, and never used in production. The hashes in this sample are not
on the live Register.

Certificate

Signed ed25519 under the notary key

0d7550754e0800a5d237eef5826035766b9b3e5a15868a940ab289958788e3b0; signature da5533
c6d83a34a28cc19877ab1cdb76b897066c51f8d78446e6a983f3664dbd519151e47ade737977ff3
5ea15ac8bfb4243614909aa5ca16a7d427dbbf967802.

```
{
  "court": "Court of Common Pleas",
  "kind": "authorisation-certificate",
  "version": 1,
  "account": "acct_0f3a7c2e-sample",
  "issuedAt": "2026-09-10T02:15:00.000Z",
  "subject": {
    "commit": "b7e3d2c19a4f6e8d0c5b3a7f2e1d9c8b6a5f4e3d",
    "repo": "example-fs/payments-gateway",
    "session": "claude-code:4f0c2d9e-8b1a-4c3d-9e2f-6a7b8c9d0e1f",
    "operator": "Example Financial Services Pty Ltd",
    "authorisedBy": {
      "by": "clerk",
      "handle": "efs-clerk",
      "agent": "efs-claude-code-ws0417",
      "mandate": {
        "sha256": "a5268072a20a8d8b1451b5d3bdcd58a9ef2235418071a6705840fbfb9beef124",
        "lodgedAt": "2026-09-08T23:04:12.000Z",
        "receiptId": "6b1e9c3a-7d2f-4e8b-a0c4-5f6d7e8f9a0b",
        "kind": "order",
        "acceptance": {
          "sha256": "a5268072a20a8d8b1451b5d3bdcd58a9ef2235418071a6705840fbfb9beef124",
          "lodgedAt": "2026-09-08T23:04:19.000Z",
          "receiptId": "1c3e5a7b-9d0f-4a2c-b4e6-8f0a2c4e6a8b",
          "roots": [
            {
              "sha256": "e73f8869ca609d37e8f0d1a1320af92d9e46b594a2691d6b9c5561f14dcae461",
              "lodgedAt": "2026-09-08T23:05:53.000Z",
              "receiptId": "9d2f4a6c-1e3b-4c5d-8f7a-2b4c6d8e0f1a",
              "n": 1,
              "sha256": "7ed19787eafcd55e54d7e13c1077264366693506fefff0f42ecdf9f0953e3085",
              "lodgedAt": "2026-09-08T23:34:10.000Z",
              "receiptId": "c4e6a8b0-3d5f-4a7b-9c1d-4e6f8a0b2c3d",
              "n": 9,
              "report": {
                "sha256": "486dd8dfde6868b9cd21d3401f55a8f6423be8fe05ff7ce8c2c9ec8dfea8e3eb",
                "lodgedAt": "2026-09-08T23:33:58.000Z",
                "receiptId": "e5a7c9b1-2f4d-4e6a-8b0c-1d3e5f7a9b1c",
                "transcript": {
                  "sha256": "b5d5b828e783a096c7133af25fbf8d12c151bf68d53bd36141c13de88faf6430",
                  "lodgedAt": "2026-09-08T23:34:10.000Z",
                  "receiptId": "a2c4e6f8-0b1d-4e3f-9a5b-7c9d1e3f5a7b",
                  "anchor": {
                    "root": "db4d58923ecaee63c4e3400ca4c766834d686e711416ad0f9ec63b5cd4f9e733",
                    "state": "confirmed",
                    "rangeUntil": "2026-09-09T00:00:00.000Z",
                    "instruction": {
                      "digest": "c82696939eb502d0fbb8355a116dcd5400c2f15f646dd8328c16d32c040a3a0",
                      "occurredAt": "2026-09-08T23:31:02.000Z",
                      "standing": {
                        "openComplaints": 0,
                        "uncuredDeclarations": 0
                      }
                    }
                  }
                }
              }
            }
          ]
        }
      }
    }
  }
}
```

What this certificate does and does not attest

What this certificate attests. That at the times shown the Register of Dealings held the hashes shown, lodged by the keys shown; that the account's record vault holds, for the session named, an instruction given at the time shown whose entry names this commit; that the session's chain has the lodged roots shown, against which the commit's leaf can be proved from the operator's copy of the chain; that the mandate was issued by the operator's Clerk and names the operator's conditions by hash in the document whose hash was lodged; and that the report shown was lodged after the instruction and covers the session.

What it does not attest. That the commit is correct, safe, or what the operator meant. Peregrini did not read the code and does not hold it. The tie between the commit and the instruction is the operator's own lodgement in its vault (why.mjs reads the commit's trailer and the chain on the operator's machine); Peregrini did not see the commit made. The certificate is evidence of the record; the review of the change is the operator's, and the change control that requires it is the operator's. A certificate is not a finding: where the operator complains of the work, the complaint, the agent's account and any cure or declaration are separate entries and are listed under *Standing*.

Sample document. Fictional operator, agent, session and commit; every hash is the SHA-256 of the sample bytes and every receipt is signed under the published sample key. Read it as a form, not as a fact about anyone.

Document 9d18a07e5192dd86be541cd8e54e73250522
da115b2b2217ee416412bded67aa