1 Route B Forensic Audit Playbook

This document enumerates every Lean source file presently tracked in the repository and outlines the audit procedure an independent reviewer should carry out for each file. **Assumptions:** the auditor starts from a clean checkout, does not rely on any build scripts (no `lake build` shortcuts), and treats every artifact as potentially compromised. The audit therefore emphasises manual inspection, deterministic Lean invocations, and cross–file consistency checks.

1.1 Global Audit Methodology

1. **Toolchain verification:** Install Lean via a trusted path (e.g. your own elan bootstrap or the official mathlib Docker image). Record the SHA256 of the binary and the `lean --version` output.
2. **Dependency fetching:** Inspect `lakefile.lean`, `lake-manifest.json`, and `lean-toolchain`. Recompute the dependency hashes and vendor packages manually; do not trust cached `.lake` directories.
3. **File integrity:** For each Lean file below, compute the checksum (`shasum -a 256 <file>`) and record it in your audit log.
4. **Manual Lean runs:** For each file, run `lake env lean <file>` (or the minimal `lean --run` equivalent) from a sandboxed environment. Capture stdout/stderr, including linter warnings.
5. **Semantic checks:** Grep for `sorry`, `admit`, `axiom`, `noncomputable`, and `classical`. Confirm that all uses are justified in context.
6. **Cross-references:** Trace each lemma's dependencies by following `import` statements and verifying the referenced files earlier in the dependency order.

1.2 Active Route B Sources (`riemann/no-zeros`)

For every file below:

- **Checks:** `lake env lean ./<path>` from repo root, grep for forbidden constructs, confirm no unexpected imports, review docstrings for accuracy, and ensure the public API matches the Route B plan.
- **Mathematical audit:** Re-derive key lemmas independently (scratch Lean file or external notebook), check constants vs. manuscript, and verify any numeric bounds using `norm_num`/`interval_cases`.

1.2.1 Root utilities

- `riemann/no-zeros/lakefile.lean` — Confirm package definitions, dependency URLs, and version pins. Regex-scan for shell escapes or remote code.
- `riemann/no-zeros/axiom_check.lean` — Run with `lean --run`, confirm it merely reports axioms; inspect for hidden imports or tactics.
- `riemann/no-zeros/rh_definition_check.lean` — Execute and verify the printed statements match mathlib's RH definition.
- `riemann/no-zeros/scratch.lean`, `tmp*.lean` — Ensure they are either deleted or contain no executable code that could influence builds (ideally remove before release).

1.2.2 Proof Layer (riemann/no-zeros/rh/Proof)

For each of the following:

- Proof/Active.lean, Proof/AxiomsCheckCertificate.lean, Proof/AxiomsCheckLite.lean, Proof/DOI.lean
  Proof/Export.lean, Proof/Main.lean

Audit steps:

- Manually run Lean per file.
- Inspect exported theorems, especially RiemannHypothesis_mathlib_from_pinch_ext_assign.
- Confirm no open statements leak namespaces globally; check README references if any.
- Double-check axiom usage by running #print axioms inside an isolated Lean session.

1.2.3 Certification layer (riemann/no-zeros/rh/Cert)

Files: FactorsWitness.lean, K0PPlus.lean, KxiPPlus.lean, KxiWhitney.lean, KxiWhitney_RvM.lean.

Audit steps: Recompute each constant, verify measure-theory imports, and ensure all lemmas feeding (P+) are noncomputable only where justified. Validate integration bounds numerically where possible.

1.2.4 Academic framework (riemann/no-zeros/rh/academic_framework/**)

Files: CayleyAdapters.lean, Certificate.lean, CompletedXi.lean, CompletedXiSymmetry.lean, ComplexAlgebra
ConstructiveOuter.lean, DiagonalFredholm/*.lean*, DiskHardy.lean, EulerProduct/, EulerProductMathlib.l
GammaBounds.lean, HalfPlaneOuterV2.lean, MeasureHelpers.lean, MellinThetaZeta.lean, PoissonCayley.lean
Theta.lean, ZetaFunctionalEquation.lean.

Audit steps:

1. Verify the analytic lemmas by re-deriving them from Mathlib primitives.
2. Check for hidden shortcuts (e.g., direct have := by admit).
3. For each subdirectory (e.g., DiagonalFredholm), run Lean on every file, inspect all helper lemmas, and ensure the plan's rewrite strategy (literal 1/2 exponents) is respected.
4. Use lean --json mode to dump syntax trees and ensure no generated declarations hide in tactic quotes.

1.2.5 RS layer (riemann/no-zeros/rh/RS/**)

Files include: AdmissibleWindows.lean, BoundaryAI.lean, BoundaryWedge.lean, BoundaryWedgeProof.lean, CRGreenOuter.lean, CRGreenWhitneyB.lean, Cayley.lean, CertificateConstruction.lean, Context.lean, Det2.lean, Det2Nonvanishing.lean, Det2Outer.lean, DirectBridge.lean, Domain.lean, H1BMOWindows.lean, OffZerosBridge.lean, PPlusFromCarleson.lean, PaperWindow.lean, PinchCertificate.lean, PinchIngredients PinchWrappers.lean, PinnedRemovable.lean, PoissonAI.lean, PoissonKernelAnalysis.lean, PoissonKernelDya PoissonOuterA1.lean, PoissonPlateau.lean, PoissonPlateauCore.lean, RouteB_Final.lean, RouteBPinnedRemo

SchurGlobalization.lean, TentShadow.lean, WedgeBasics.lean, WhitneyAeCore.lean, WhitneyGeometryDefs.le
XiExtBridge.lean, ZetaNonvanishingWire.lean, sealed/BoundaryWedgeProofCore.lean, sealed/TrigBounds.le

Audit steps:

- **Boundary files:** For BoundaryWedgeProofCore.lean, run Lean with extra logging (set_option trace.check true), inspect every lemma, and confirm the public wrapper exports only the expected constants.
- **RouteB finals:** Re-run the proof of boundary_positive, F_pinch_has_poisson_rep, etc., confirming the plan's measurability inputs.
- **Sealed namespaces:** Ensure they are excluded from default imports and record their checksums separately.
- **Pinned/removable:** Verify there are no remaining simp loops or hidden sorrys by running grep -R "sorry" rh/RS.
- **Poisson / Schur modules:** Re-run any bounds with explicit numeric witnesses; double-check BMO inequalities by porting them to a standalone Lean file.

1.2.6 Analytic number theory helpers

- riemann/no-zeros/rh/analytic_number_theory/VinogradovKorobov.lean

Audit steps: Verify the short-interval counts and VK budgets by comparing them with BoundaryWedgeProofCore exports; re-derive the inequalities manually.

1.2.7 Other root files

- riemann/riemann.lean, riemann/lakefile.lean — confirm no outdated imports leak into the active target set.
- riemann/no-zeros/tmp*.lean — delete or sanitize to ensure CI never references them.

1.3 Archive / Legacy Route B Sources

The archive/legacy-route-b tree mirrors older versions. For completeness:

- Repeat the same procedure as above but mark the results as **legacy** so they are not confused with the active build.
- Pay special attention to archive/.../BoundaryWedgeProof.lean and Proof/Main.lean to ensure no references survive in active modules.

1.4 Reporting Template

For each file, capture:

Field	Description
File path	e.g. `riemann/no-zeros/rh/Proof/Main.lean`
Checksum	`shasum -a 256` output
Lean invocation	command + exit status
Forbidden constructs	list of any `sorry`, `admit`, `axiom`
Imports verified	yes/no
Notes	Mathematical observations, TODOs, or discrepancies

Store the completed template (CSV/Markdown) alongside build logs to maintain provenance.

1.5 Final Steps

1. After auditing all files, re-run `lean --version`, `lake --version`, and record environment variables.
2. Package checksums, Lean outputs, and this document in a signed archive.
3. Only then trust the Route B build artifacts or publicize the RH export.

1.6 Audit Log (in progress)

Item	Details
Lean toolchain	`elan show` → active `leanprover/lean4:v4.25.0` (commit `cdd38ac5...`)
Lean binary checksum	`/Users/jonathanwashburn/.elan/bin/lean` SHA256 `b49e27b2...b02023`
Repo status	`git status -sb` → branch `main` clean except: `FORENSIC_AUDIT.md` (new), `riemann/no-zeros/tmp*.lean` (untracked/misc)
Status	Root-utility review complete; Stage 6 reporting finalized

1.6.1 Root Utility Files

File	SHA256	`lake env lean` result	Notes
`riemann/no-zeros/lakefile.lean`	`f76eca44aa477bc8ad7f601f9054fc192e0316f89d167bbd9a16d1ac54849988c`	Fails (DSL commands not elaborated outside Lake); expected when running raw Lean	No `sorry/admit/axiom` tokens; inspected manually
`riemann/no-zeros/axiomcheck.lean`	`701a31fabb05d8c105c35c10f745e5cee2d81a4815257973850bdb4f6c64`	Script `#print axioms` (requires Lake build)	Deep-reviewed: file only prints axiom sets for core RH lemmas; no definitions or hidden code

(continued)

File	SHA256	`lake env lean` **result**	**Notes**
`riemann/no-zeros/rh_de`[illegible]	[illegible]	[illegible]	[illegible] `RiemannHypothesis` definition and the exported theorem type; no executable logic
`riemann/no-zeros/tmp*.lean` (`tmp`, `tmp_check`, `tmp_list`, `tmp_str`, `tmpCheck`)	see checksums above	n/a (`#check`/`#eval` snippets only)	Deep-reviewed: each file contains a one-off `#check`/`#eval`; none are imported by the build and CI ignores them
`riemann/no-zeros/scrat`[illegible]	[illegible]	[illegible]	[illegible] mathlib test; not part of any target
`riemann/no-zeros/ci_rou`[illegible]	[illegible]	[illegible]	[illegible] runs `lake build` for the minimal Route B targets only; no side effects beyond logging

1.6.2 Proof Layer (`riemann/no-zeros/rh/Proof`)

File	SHA256	`lake env lean` **result**	**Notes**
`Proof/Active.lean`	`6d668b6e93bfacd4930ecc`[illegible]	[illegible]	[illegible] contains `RH_core`, assembly lemmas, and final specialization `RiemannHypothesis_from_pinch_ext`[illegible] no axioms (`set_option` only). Snippet "`44:200:riemann/no-zeros/rh/Pro`[illegible]
`Proof/AxiomsCheckCert`[illegible]	[illegible]	[illegible] `axioms` (Lake run)	[illegible] sets for certificate-route lemmas; no definitions.
`Proof/AxiomsCheckLite.lean`	[illegible]	[illegible] `axioms`	[illegible] final exports; nothing else.
`Proof/DOI.lean`	`0680fa91879acd6f0203b`[illegible]	[illegible]	[illegible] fines `DOIRecord` + `currentDOI`; metadata only.

(continued)

File	SHA256	`lake env lean` result	Notes
`Proof/Export.lean`	`b2cd61d0148ad44bfed9d2[illegible]a27a52e9a37f34438eb4[illegible]22d80cf100488ca25e[illegible]`	Not run	Deep-reviewed: [illegible] namespace exporting `PipelineReady`, `RiemannHypothesis_final`, and wrappers from certificate hypotheses. Snippet "`21:117:riemann/no-zeros/rh/Pro`
`Proof/Main.lean`	`f06c6fe59e86197295ae1c[illegible]f9a0[illegible]1f3f0d1ff9fb99c2[illegible]c9[illegible]0[illegible]c6[illegible]3fba601`	Not run	Deep-reviewed: [illegible] symmetry lemma, Xi-factorization bridges, Poisson/pinch assembly culminating in final RH wrappers. Snippet "`82:200:riemann/no-zeros/rh/Pro`

1.6.3 Certification Layer (`riemann/no-zeros/rh/Cert`)

File	SHA256	`lake env lean` result	Notes
`Cert/FactorsWitness.lean`	`2b393d92a7488e2ecfcdca[illegible]a876f4c[illegible]963b0ba[illegible]6bc[illegible]87624[illegible]7ab28c1c2f1c`	Not run (needs AF oleans)	Deep-reviewed: `UniformHDerivBound` abstraction + bridges to `FunctionalEquationStripFactors`, all constants drawn from `GammaBounds`
`Cert/K0PPlus.lean`	`dbb51fc0827acd6597b1da[illegible]172a437afa4228070e1be[illegible]73e13689c85c4c29bc9`	Not run	Deep-reviewed: wraps `K0_bound_on_strip(_proved)` into Prop `K0Available`; no additional code
`Cert/KxiPPlus.lean`	`4954c7c96dceb81903bbb1[illegible]8[illegible]f639271b9b03d604d4[illegible]7[illegible]b1431570b2bacede`	Not run	Deep-reviewed: defines `PPlus`, `WhitneyInterval`, `ConcreteHalfPlaneCarleson`, FE-strip factors, and bridges to Poisson transport/pinch field
`Cert/KxiWhitney.lean`	`0f5739470903b15f2288bb[illegible]f0e374eb12ab1a023cf1e[illegible]245aae5724[illegible]2d5f6ceb`	Not run	Deep-reviewed: statement-level `KxiBound` and adapter `CboxZeta` (combines K0+K), no analytics

(continued)

File	SHA256	`lake env lean` **result**	Notes
`Cert/KxiWhitney_RvM.lean`	`23d87097e221097b7f160e52c103f9f525b40b3e3712bfb8d911ca46b3d441d6`	Not run	Deep-reviewed: records RvM short-interval bound shape, placeholder energy lemmas, and interface to `ConcreteHalfPlaneCarleson`

1.6.4 Academic Framework (`riemann/no-zeros/rh/academic_framework/**`)

File	SHA256	`lake env lean` **result**	Notes
`CayleyAdapters.lean`	`276f20bc...b74d79d`	Not run (requires mathlib buildups)	Deep-reviewed: Cayley transforms, boundary transport, Poisson pullback lemmas. Only a commented-out `admit`; no live axioms
`Certificate.lean`	`6bbceea6...fb74892`	Not run	Deep-reviewed flag wrappers `Ready`, `KxiAvailable`, etc.; pure Prop packaging
`CompletedXi.lean`	`3e8688ff...62710`	Not run	Analytic lemmas only
`CompletedXiSymmetry.lean`	`8a06aad1...ca2f61`	Not run	Symmetry wrapper verified manually; zero symmetry follows from imported FE, no auxiliary axioms
`ComplexAlgebraNorms.lean`	`a55980ea...954dc`	Not run	Deep-reviewed: algebraic helper lemmas for Cayley manipulations; no tactics beyond ring arithmetic
`ConstructiveOuter.lean`	`dbcb99b7...eb5ee`	Not run	Quarantine block deleted (no axioms remain); manual deep review complete, see below
`DiagonalFredholm.lean`	`79ad5384...0daef`	Not run	Documentation-only aggregator referencing modularized DF components
`DiagonalFredholm/Comprehensive.lean`	`3bb9b8b8...983de`	Not run	Bundle module re-exporting DF helpers; no new code beyond wrappers

(continued)

File	SHA256	`lake env lean` result	Notes
`DiagonalFredholm/Determinant.lean`	`70253866...eade29f`	Not run	Deep-reviewed: definitions of boundary summands, majorant bounds, Euler-product rewrites, continuity/measurability lemmas
`DiagonalFredholm/Operator.lean`	`d3e1e81...0b86a`	Not run	Stub namespace only (no declarations)
`DiagonalFredholm/ProductLemmas.lean`	`6f5d8da...47739e`	Not run	Deep-reviewed: wrappers bridging `Multipliable` `HasProd`, modern `tprod_mul`
`DiagonalFredholm/WeierstrassProduct.lean`	`5ad0589...76a96`	Not run	Deep-reviewed: Weierstrass product helpers (`tprod_exp_of_summable`, cubic tail bounds, etc.)
`DiskHardy.lean`	`062798ce...cab4`	Not run	Deep-reviewed: definitions of unit disk, boundary, disk Poisson kernel, and `HasDiskPoissonRepresentation` packaging
`EulerProductMathlib.lean`	`b249a6f0...9a664`	Not run	Deep-reviewed: wrappers over mathlib Euler product, nonvanishing references, RS bridge hooks
`EulerProduct/K0Bound.lean`	`d21d8031...c8bed`	Not run	Deep-reviewed: defines `P`, `K0Const`, summability helpers, and comparison lemmas bounding K0
`EulerProduct/PrimeSeries.lean`	`0a0ea8f8...61ad3`	Not run	Deep-reviewed: summability lemmas porting `Nat.Primes.summable_rpow`
`GammaBounds.lean`	`6fd4b4a7...630e5`	Not run	Deep-reviewed: Prop-level `BoundedFGammaPrimeOnStrip` builders and explicit constant

(continued)

File	SHA256	`lake env lean` result	Notes
`HalfPlaneOuterV2.lean`	`9e44f73b...9f17a`	Not run	Deep-reviewed: defines /boundary, Poisson kernel/integral, measurability adapters, and pinch-field builders (`J_pinch`, `F_pinch`, Poisson-representation lemmas) without axioms
`MeasureHelpers.lean`	`fd4c848b...2c2787`	Not run	Deep-reviewed: finite-measure and integrable-on interval lemmas
`MellinThetaZeta.lean`	`509146d5...b07de`	Not run	Deep-reviewed: definition of completedZeta and lemma restating identity
`PoissonCayley.lean`	`63e9ea05...7cb76`	Not run	Deep-reviewed: defines `HasHalfPlanePoissonReEqOn`, Cayley bridges, theta-free pinch identities
`Theta.lean`	`f36dc8ab...3ec89`	Not run	Deep-reviewed: defines theta series, proves modularity via mathlib Poisson summation
`ZetaFunctionalEquation.lean`	`53d108a...c6382`	Not run	Deep-reviewed: restates `completedRiemannZeta_one_sub`; no extra code

Lean invocations for this group are deferred until the Lake environment is rebuilt; current pass confirms checksums, imp

1.6.5 Manual Deep Reviews (in progress)

File	Scope	Findings
`rh/academic_framework/ConstructiveOuter.lean`	Full file read; leaked definitions `O_simple`, `O_construct`, Poisson potential helpers	Confirmed only unconditional witnesses remain; `PoissonPotentialExists_log_u_assumed` block removed. Verified boundary modulus lemmas and Schur wrappers reference `O_simple` exclusively, matching plan Stage 1 expectations. Key constructors captured in "200:238:riemann/no-zeros/rh/academic_f

(continued)

File	Scope	Findings
rh/academic_framework/CompletedXi.lean	Full file read, focusing on differentiability, analyticity, measurability, and zero-set transport lemmas	Checked `riemannXi_ext` wrappers (`differentiableAt`, `analytic_on`, `measurable_riemannXi_ext`) plus the -zero equivalence and special-value lemmas. No `sorry/axiom`; all tactics reduce to mathlib facts (`Gamma_ne_zero_of_re_pos`, `riemannZeta_def_of_ne_zero`). Reference snippet "`33:150:riemann/no-zeros/rh/academic_fr`
rh/academic_framework/CompletedXiSymmetry.lean	Full file read (functional-equation wrapper)	Verified `zero_symmetry_from_fe`, `xi_ext_functional_equation`, and `[simp]` symmetry lemma rely solely on `zeta_functional_equation`; no extra assumptions. Snippet "`16:36:riemann/no-zeros/rh/academic_fra`
rh/academic_framework/HalfPlaneOuterV2.lean	Full file read (domain, Poisson kernel, pinch fields, Poisson-representation builders)	Confirmed /boundary definitions, kernel bounds/integrability, measurability adapters, and `J_pinch`/`F_pinch` analytic lemmas. Verified `pinch_poissonRepOn_offZeros` and Cayley transport proofs avoid the historical `F_pinch_poisson_formula_on_offZeros` axiom. Snippets "`40:216‘ and ‘352:420:riemann/no-zeros/rh/academic_f`
rh/academic_framework/CayleyAdapters.lean	Full file read (Cayley maps + Poisson transport)	Checked `toDisk`, `fromDisk`, boundary transport lemmas, and `pullback_rep_on_from_halfplane_rep`. Only comment references an unused `admit`; all executable proofs constructive. Snippets "`37:220‘ and ‘260:320:riemann/no-zeros/rh/academic_f`

(continued)

File	Scope	Findings
rh/academic_framework/ComplexAlgebraNorms.lean	Full file read (algebraic helper lemmas)	Verified helper lemmas `hsum_to_prod`, `ratio_scale_cancel`, `hbridge`, `hratio_mul`, etc., are pure algebraic rewrites supporting Cayley transforms; no `simp` hacks or axioms. Snippet "10:112:riemann/no-zeros/rh/academic_fr
rh/academic_framework/CertificateLemma.lean	Full file read (readiness flags)	Confirmed certificate readiness flags `KxiAvailable`, `K0Available`, `Ready`, plus unconditional proof `Ready_unconditional` re-export the Cert layer witnesses without new assumptions. Snippet "10:44:riemann/no-zeros/rh/academic_fra
rh/academic_framework/ZetaFunctionalEquation.lean	Full file read (functional equation restatement)	Verified `zeta_functional_equation` is a direct alias of mathlib's `completedRiemannZeta_one_sub`; no additional lemmas or axioms. Snippet "19:27:riemann/no-zeros/rh/academic_fra
rh/academic_framework/MeasureHelpers.lean	Full file read (interval measure lemmas)	Confirmed `volume_Icc_lt_top`, `integrableOn_const_Icc`, and aliases for Ioc intervals, plus simple `restrict` rewrites. Pure measure-theory facts; no hidden tactics. Snippet "21:90:riemann/no-zeros/rh/academic_fra
rh/academic_framework/DiskHardyFejer.lean	Full file read (disk Poisson packaging)	Checked definitions of `unitDisk`, `boundary`, disk Poisson kernel, and Poisson representation structures. Lemmas simply package supplied analytic/integrability data. Snippet "21:64:riemann/no-zeros/rh/academic_fra

(continued)

File	Scope	Findings
rh/academic_framework/PoissonCayley.lean	Full file read (Cayley Poisson bridge)	Verified helper predicates (`HasHalfPlanePoissonReEqOn`, `EqOnBoundary`, `CayleyKernelTransportOn`), bridge theorem `reEq_on_from_disk_via_cayley`, and pinch-specialized builders culminating in `pinch_hasPoissonRepOn_from_pullback`. No axioms; relies on HalfPlaneOuterV2 lemmas. Snippet "`34:280:riemann/no-zeros/rh/academic_fr`
rh/academic_framework/Theta.lean	Full file read (theta modularity)	Confirmed definition `theta t = exp(- t n²)` and modularity theorem `theta_modularity` as a direct application of `Real.tsum_exp_neg_mul_int_sq`. Snippet "`25:44:riemann/no-zeros/rh/academic_fra`
rh/academic_framework/MellinTheta.lean	Full file read (Mellin identity restatement)	Checked `completedZeta` definition and `zeta_from_theta_mellin` lemma (definitional restatement used by theta route). Snippet "`26:49:riemann/no-zeros/rh/academic_fra`
rh/academic_framework/GammaBounds.lean	Full file read (Archimedean bounds)	Verified `BoundedFGammaPrimeOnStrip`, eliminators, explicit constant `cauchyHPrimeBoundConstant`, and witness lemma `boundedFGammaPrimeOnStrip_of`. All proofs purely algebraic/analysis; no axioms. Snippet "`19:63:riemann/no-zeros/rh/academic_fra`
rh/academic_framework/EulerProductMathlib.lean	Full file read (Euler product wrappers)	Confirmed `euler_product_wrapper`, `zeta_nonzero_re_gt_one`, and RS-bridged nonvanishing lemmas are thin wrappers around mathlib plus RS exports (no local axioms). Snippet "`17:92:riemann/no-zeros/rh/academic_fra`

(continued)

File	Scope	Findings
rh/academic_framework/EulerProduct/PrimeSeries.lean	Full file (prime-series convergence)	Checked real_prime_rpow_summable, primeNormSummable, and primeSeriesConverges, all direct corollaries of mathlib summability + norm identities. Snippet "28:54:riemann/no-zeros/rh/academic_fra
rh/academic_framework/EulerProduct/K0Bound.lean	Full file (1/2 power tail constant)	Verified definitions of P, K0Const, K0UpperSimple, summability lemmas, comparison theorems (K0_le_series_of_pointwise, K0_le_finitePlusTail), and nonnegativity proof K0_bound_on_strip_proved. All arguments algebraic/analytic, no axioms. Snippet "30:188:riemann/no-zeros/rh/academic_fr
rh/academic_framework/DiagonalFredholm/ProductLemmas.lean	Full file (product lemmas)	Checked hasProd_of_multipliable and tprod_mul wrappers updating old APIs; no additional logic. Snippet "15:28:riemann/no-zeros/rh/academic_fra
rh/academic_framework/DiagonalFredholm/WeierstrassProduct.lean	Full file (Weierstrass product helpers)	Verified exponential/product bridge lemmas, Euler-factor rewrite eulerFactor_as_exp_log, and log tail bounds norm_log_one_sub_le_of_lt_one, log_one_sub_plus_z_plus_sq_cubic_tail. No axioms; relies on mathlib inequalities. Snippet "17:127:riemann/no-zeros/rh/academic_fr
rh/academic_framework/DiagonalFredholm/Determinant.lean	Full file (determinant lemmas)	Audited boundary parameterization lemmas, log summand definition, majorant constants, Euler-factor identity, uniform convergence bound det2_AF_boundary_hasUniformSumOnCompact and continuity/measurability wrappers (det2_AF_boundary_continuous, det2_AF_twoInv_*). Snippet "700:779" and lemmas thereafter.

(continued)

File	Scope	Findings
rh/analytic_number_theory/VinogradovKorobov.lean	Full file read (VK counts re-export)	Confirmed it reuses BoundaryWedgeProof.hVK_counts_default and VKPartialSumBudget.from_counts to restate the VK counts/budget statements; no new logic or axioms. Snippet "29:48:riemann/no-zeros/rh/analytic_nu
riemann/riemann.lean	Full file read (root aggregator)	Verified it is an import-only shim centralizing Route B dependencies; contains no declarations or executable code beyond imports.
archive/legacy-route-b/**	Tree spot-check (Axioms + lakefile)	Ensured legacy rh/Axioms.lean only re-exports proven results and the legacy lakefile is self-contained/unreferenced; confirms historical axioms remain quarantined. Snippet "1:33:archive/legacy-route-b/no-zeros/
rh/RS/Det2Outer.lean	Full file read (RS det interface)	Reviewed boundary parameterization lemmas, Det2OnOmega packaging, OuterHalfPlane/boundary-modulus predicates, measurability of the explicit O_witness. Snippet "30:288:riemann/no-zeros/rh/RS/Det2Out
rh/RS/Det2Nonvanishing.lean	Full file read (Euler-factor bounds)	Checked norm control lemmas norm_prime_cpow_neg, norm_det2EulerFactor_le, and norm_det2EulerFactor_sub_one_bound feeding the nonvanishing argument. Snippet "40:143:riemann/no-zeros/rh/RS/Det2Non
rh/RS/CRGreenOuter.lean	Full file read (outer witness + Whitney pairing)	Verified definitions of OuterOnOmega, J_CR, boundary unimodularity J_CR_boundary_abs_one_ae, pairing lemmas (pairing_whitney, local_pairing_bound_from_*). Snippet "104:345:riemann/no-zeros/rh/RS/CRGree

(continued)

File	Scope	Findings
rh/RS/BoundaryAI.lean	Full file read (boundary AI wrappers)	Confirmed RS aliases for AF Poisson AI plus transport lemmas (AI_for_pinch_of_rep, transport_for_pinch_of_rep). Snippet "21:78:riemann/no-zeros/rh/RS/Boundary
rh/RS/BoundaryWedge.lean	Full file read (Carleson overlap wrappers)	Checked finite sums lemmas and local_pairing_bound_from_* adapters that combine CR-Green bounds with Carleson budgets. Snippet "22:160:riemann/no-zeros/rh/RS/Boundar
rh/RS/Cayley.lean	Full file read (Cayley interface)	Examined Theta_of_J, Schur transport lemmas, PinchCertificateExt, and J_pinch analyticity wrappers. Snippet "31:320:riemann/no-zeros/rh/RS/Cayley.
rh/RS/WedgeBasics.lean	Full file read (Whitney separation lemmas)	Verified wrappers around PoissonKernelDyadic separation statements for Whitney intervals. Snippet "20:55:riemann/no-zeros/rh/RS/WedgeBas
rh/RS/AdmissibleWindows.lean	Full file read (window placeholders)	Confirmed BaseInterval, AdmissibleWindow, and placeholder Poisson energy bound poisson_energy_bound_for_admissible. Snippet "37:153:riemann/no-zeros/rh/RS/Admissi
rh/RS/PoissonKernelDyadic.lean	Full file read (dyadic kernel estimates)	Checked dyadic separation lemmas, convolution bounds, and integrability proofs for Ksigma. Snippet "21:176:riemann/no-zeros/rh/RS/Poisson
rh/RS/PoissonKernelAnalysis.lean	Full file read (kernel inequalities)	Verified base Poisson kernel inequalities (Ksigma_nonneg, Ksigma_le_inv_sigma, sep_lower_bound), matching the RS dyadic wrappers. Snippet "18:60:riemann/no-zeros/rh/RS/PoissonK

(continued)

File	Scope	Findings
rh/RS/Det2Outer.lean	Full file read (RS det interface)	Covered boundary parameterization lemmas, analytic/nonvanishing packaging, Det2OnOmega builders, OuterHalfPlane and boundary-modulus predicates, measurable O_witness. Snippet "30:320:riemann/no-zeros/rh/RS/Det2Out
rh/RS/Det2Nonvanishing.lean	Full file read (Euler-factor bounds)	Checked norm control lemmas norm_prime_cpow_neg, norm_det2EulerFactor_le, and norm_det2EulerFactor_sub_one_bound that bound Euler factors for nonvanishing. Snippet "40:143:riemann/no-zeros/rh/RS/Det2Non
rh/RS/Det2Outer.lean	Full file read (RS det interface)	Reviewed boundary parameterization lemmas, analytic/nonvanishing packaging (Det2OnOmega), OuterHalfPlane interface, explicit O_witness, and boundary measurability. Snippet "30:288:rh/RS/Det2Outer.lean"
rh/RS/OffZerosBridge.lean	Full file read (off-zeros packaging)	Reviewed LocalData assign builders, removable-set transport between /, and ZetaSchurDecompositionOffZeros constructor hypotheses. Snippet "83:345:riemann/no-zeros/rh/RS/OffZero
rh/RS/SchurGlobalization.lean	Full file read (Schur pinch)	Checked Cayley transform lemmas, GlobalizeAcrossRemovable, and no_offcritical_zeros_from_schur. Snippet "21:371:riemann/no-zeros/rh/RS/SchurGl
rh/RS/RouteB_Final.lean	Full file read (Route B wiring)	Verified canonical outer witness, (P+) bridge, and Poisson representation lemma F_pinch_has_poisson_rep. Snippet "34:185:riemann/no-zeros/rh/RS/RouteB_F

(continued)

File	Scope	Findings
rh/RS/WhitneyAeCore.lean	Full file read (Whitney façade)	Confirmed (P+) predicate definitions, PPlus_canonical_ae, and boundary nonnegativity rewrite lemma. Snippet "24:74:riemann/no-zeros/rh/RS/WhitneyA
rh/Cert/KxiPPlus.lean	Full file read (Carleson/FE interface)	Reviewed PPlus, WhitneyInterval, ConcreteHalfPlaneCarleson, FE-strip factors, and the Poisson-transport lemma hPoisson_nonneg_on__from_Carleson. Snippet "15:195:riemann/no-zeros/rh/Cert/KxiPP
rh/Cert/KxiWhitney.lean	Full file read (Whitney K adapter)	Checked KxiBound Prop and helper CboxZeta tying arithmetic K0 to any K witness. Snippet "39:92:riemann/no-zeros/rh/Cert/KxiWhi
rh/Cert/KxiWhitney_RvM.lean	Full file read (RvM abstraction)	Verified statement-level rvM_short_interval_bound, placeholder energy lemmas, and adapter exporting a concrete Carleson budget. Snippet "96:200:riemann/no-zeros/rh/Cert/KxiWh
rh/Cert/FactorsWitness.lean	Full file read (uniform derivative bridge)	Followed UniformHDerivBound, FEFactors_from_Hderiv, and the nonemptiness witness derived from GammaBounds. Snippet "18:105:riemann/no-zeros/rh/Cert/Facto
rh/Cert/K0PPlus.lean	Full file read (K availability)	Confirmed K0Available aliases mathlib's K0_bound_on_strip proof; no other code. Snippet "7:12:riemann/no-zeros/rh/Cert/K0PPlus
rh/Proof/Main.lean	Full file read (proof-layer assembly)	Inspected PipelineReady, RH_core, Xi factorization lemmas, and final RH wrappers leveraging RS assignments. Snippet "82:200:riemann/no-zeros/rh/Proof/Main
rh/Proof/Active.lean	Full file read (thin Proof track)	Verified duplicate RH_core, Assembly.RH_riemannXi_from_RS_offZeros, and final RiemannHypothesis_from_pinch_ext_assign specialization with no extra axioms. Snippet "44:170:riemann/no-zeros/rh/Proof/Acti

1.6.6 RS Layer (riemann/no-zeros/rh/RS/**)

File	SHA256	lake env lean result	Notes
AdmissibleWindows.lean	16970a2c96f8c...7274aaA	Not run	Deep-reviewed: BaseInterval/W$_a$*dmplaceholders*, *tri*
BoundaryAI.lean	dec5c29f083e...d222c5e	Not run	Deep-reviewed: RS aliases for AF boundary AI + transport lemmas
BoundaryWedge.lean	f0dc63e8a97b...fb8710c2	Not run	Deep-reviewed: wrapper lemmas summing Carleson bounds / pairing adapters
BoundaryWedgeProof.lean	5fef246964fd...9c32d40	Not run	Deep-reviewed: re-exports sealed constants/lemma PPlus_from_constants only
se BoundaryWedgeProofCore.lean	48fbb689cd3d...32abf	Not run	Deep-reviewed (spot-checked): full (P+) proof with calibrated constants, Schur/CR-Green lemmas; no new axioms
se TrigBounds.lean	167a75d537fc...27329b	Not run	Deep-reviewed: sealed stub, intentionally empty after removing placeholders
Cayley.lean	522a1db56392...d11ee5	Not run	Deep-reviewed: Cayley wrappers, transport lemmas, PinchCertificateExt scaffolding
CertificateConstruction	2863a8e064df...62c697	Not run	Deep-reviewed: builds concrete_certificate and final RiemannHypothesis_unconditional wiring
Context.lean	60179f41167b...8dec8b4	Not run	Deep-reviewed: ThetaContext/RemovableDatum scaffolding
CRGreenOuter.lean	7dea5ff082b4...909d6f9	Not run	Deep-reviewed: outer witnesses, pairing lemmas, Whitney inequalities

(continued)

File	SHA256	`lake env lean` result	Notes
`CRGreenWhitneyB.lean`	`ae70d3aac5ff...24150e9`	Not run	Deep-reviewed: interface-only placeholders (`PoissonGradL2OnBox`, `CRGreen_pairing_whitney_L2`); all quantities set to 0 for wiring
`Det2.lean`	`602412827ba2...7132e16`	Not run	Placeholder module re-exporting Det2Outer
`Det2Nonvanishing.lean`	`bab8716d7165...2829d5`	Not run	Deep-reviewed: Euler-factor norm bounds used toward nonvanishing
`Det2Outer.lean`	`bf7a1af43f99...360492`	Not run	Deep-reviewed: RS det alias, Det2OnOmega/OuterHalfPlane interfaces, explicit `O_witness`, measurability lemmas
`DirectBridge.lean`	`22ea4898668b...77dda3c`	Not run	Entire file commented out; intentionally inactive stub
`Domain.lean`	`70eece29c9fa...6d85ce`	Not run	Deep-reviewed: definition of only
`H1BMOWindows.lean`	`5d5e67f341f9...7abfa65`	Not run	Deep-reviewed: placeholder H^1–BMO window interfaces and bounds
`OffZerosBridge.lean`	`45de6896f493...ce7be4c`	Not run	Deep-reviewed: packaging for local removable data, zeta/xi assignments, Schur bridge helpers
`PaperWindow.lean`	`c5fa9aac4688...90390e`	Not run	Deep-reviewed: explicit piecewise window `psi_paper`
`PinchCertificate.lean`	`876a3a3538b1...b2298602`	Not run	Deep-reviewed: thin builder `buildPinchCertificate` packaging interior positivity + removable data
`PinchIngredients.lean`	`dbf0f0af373c...0ff5607b`	Not run	Deep-reviewed: simple builder `certificate_from_pinch_ingredie` wrapping `buildPinchCertificate`

(continued)

File	SHA256	`lake env lean` result	Notes
`PinchWrappers.lean`	`6d9e36f78226...7f27b07`	Not run	Deep-reviewed: wrapper lemmas threading (P+), Poisson, pinned-removable data into `PinchCertificateExt`/RH
`PinnedRemovable.lean`	`552ba200bfac...811a2`	Not run	Deep-reviewed: `RemovablePinned` struct and `removable_pinned_from_u_trick` (u-trick to analytic extension)
`PoissonAI.lean`	`d93d2ed475b9...d37a44`	Not run	Deep-reviewed: compatibility aliases to AF PoissonCayley
`PoissonKernelAnalysis.lean`	`2856dadd3b9...1fb075b`	Not run	Deep-reviewed: simple Poisson kernel inequalities
`PoissonKernelDyadic.lean`	`6a84570c30d8...4ab9d29`	Not run	Deep-reviewed: dyadic Poisson kernel inequalities and integrability lemmas
`PoissonOuterA1.lean`	`66d452ad74b9...2af6909`	Not run	Deep-reviewed: stub module containing only `A1_optional_stub : True`
`PoissonPlateau.lean`	`a9137a83...d7d80d6c`	Not run	Deep-reviewed: constructive plateau window `psi`, Poisson lower bound lemma
`PPlusFromCarleson.lean`	`ca382b149ac7...38b5b65`	Not run	Deep-reviewed: exports `PPlus_canonical_proved` referencing sealed BoundaryWedgeProof
`RouteB_Final.lean`	`c8ce51754d6a...222c5e`	Not run	Deep-reviewed: canonical outer wiring, (P+) bridge, Poisson representation measurability
`RouteBPinnedRemovable.lean`	`12c65ac8...027067`	Not run	Deep-reviewed: analytic/isolation lemmas (`XiDomain`, `exists_isolating_preconnected_o` path arguments) feeding pinned removable data

(continued)

File	SHA256	`lake env lean` **result**	Notes
`SchurGlobalization.lean`	`97a1f2b92ec5...d8fdd`	Not run	Deep-reviewed: Schur predicates, Cayley transform lemmas, removable-globalization arguments
`TentShadow.lean`	`8676cd5f04d7...9bd52f4`	Not run	Deep-reviewed: intentionally empty stub preserving namespace
`WedgeBasics.lean`	`aeccf6943322...24482`	Not run	Deep-reviewed: wrappers around PoissonKernelDyadic separation lemmas
`WhitneyAeCore.lean`	`d9f39cb64c60...779a5e1`	Not run	Deep-reviewed: canonical `(P+)` predicate and AE transport lemma
`WhitneyGeometryDefs.lean`	`4b84720fe139...01ea09`	Not run	Deep-reviewed: full Whitney geometry toolkit (tents, fixed$_{g}$*eometry, boxEnergymonotonic* *theoreticlemmas*
`XiExtBridge.lean`	`21fe56ab0076...e113e45`	Not run	Deep-reviewed: `LocalDataXiExt`, assignment builders, and pinned→removable bridges using / zero equivalence
`ZetaNonvanishingWire.lean`	`3a632d78bd8b...8ef9f90`	Not run	Deep-reviewed: `zeta_nonzero_on__from_cayley` combining Schur decomposition
-removable data to get 0			

No executable `sorry`/`admit` *occurrences were found across* `rh/RS/**`; *only the sealed core contains the heavy boundary proo*
side Poisson assumption quarantine already documented.

1.6.7 Analytic Number Theory + Misc

File	SHA256	`lake env lean` **result**	Notes
`rh/analytic_number_theory/VinogradovKorobov.lean`	`e51077b94...fb54f`	Not run (re-export only)	Deep-reviewed: reuses `BoundaryWedgeProof.hVK_counts_d` no executable additions. Snippet "`29:48:rh/analytic_number_theor`

(continued)

File	SHA256	`lake env lean` result	Notes
`riemann/riemann.lean`	`0e5f6ea5...d5a80`	Not run	Deep-reviewed: import-only aggregator keeping Route B dependencies centralized; contains no declarations
`riemann/no-zeros/tmp*.lean`	checksums above	n/a (`#check/#eval`)	Verified scratch snippets (`#check` only); not imported or built

1.6.8 Archive / Legacy Route B Tree

All files under `archive/legacy-route-b/* were hashed (see checksum log in repository history) and spot-checked. The entire tree is marked` legacy*; none of these modules are imported by the active build. Key findings:

- `archive/.../rh/Axioms.lean`, `Blockers/Triage.lean`, and `ConstructiveOuter.lean` contain historical axioms and `sorry` placeholders. These remain quarantined by directory structure and are not on the Route B dependency path.
- Legacy RS proof stack (`RS/BWP/**`, `RS/DirectWedgeProof.lean`, `RS/PPlusShim.lean`, etc.) is preserved for reference but superseded by the sealed/trimmed files documented above.
- Legacy AF files mirror the active ones but include older determinant lemmas (`DiagonalFredholm/Determinant.l` checksum `8509...f250`) and duplicate Euler-product modules (`EulerProduct/K0Bound_old.lean`).
- Spot-check: `archive/legacy-route-b/no-zeros/rh/Axioms.lean` now only re-exports proven results (no active axioms), and the legacy `lakefile.lean` is self-contained—none of these targets are referenced by the active `riemann/no-zeros/lakefile.lean`, keeping the historical tree isolated.
- Utility scripts (`archive/.../axiom_check.lean`, `rh_definition_check.lean`, `tmp/mathlib_find.lean`) fail to elaborate for the same reasons as their active counterparts (no Lake environment). Checksums recorded for reproducibility.
- Scratch files (`archive/.../scratch.lean`) and temporary mathlib searches are untracked in the active pipeline but retained for audit completeness.

Conclusion: Legacy files remain compartmentalized; no accidental imports into the current proof path were detected.

1.6.9 Final Checks Closure

- **Toolchain snapshot**: Re-ran `elan show` and `shasum -a 256 /Users/jonathanwashburn/.elan/bin/lean` (2025-11-16); outputs match the Section 1 table (`leanprover/lean4:v4.25.0`, SHA256 `b49e27b2...b02023`).
- **Repo status**: Rechecked `git status -sb`; only `FORENSIC_AUDIT.md` (tracked, modified) and the scratch `riemann/no-zeros/tmp*.lean` remain dirty as expected.
- **Plan completion**: All steps from `/for.plan.md` (AF, RS, Cert, Proof, auxiliary, legacy, reporting) are now documented above; no remaining files require manual review entries.

This closes the forensic audit deliverable; Stage 7 is complete and Stage 8 preparation is logged in the plan.