Sensitive Data, Shared Cloud: Reviewing a Plan for a P3–P4 Research Environment on Google Cloud

Chuck Forsyth

UCR Research Computing, University of California, Riverside

November 2026

Public edition. Campus systems, offices and roles are named by function and design weaknesses are described by kind. Nothing described here has been built.

Condensed from the companion report, *Sensitive Data, Shared Cloud: Planning a P3–P4 Secure Research Environment for HIPAA and FERPA Data on Google Cloud*, version 1.2 (public edition), doi:10.5281/zenodo.23890331.

Cite this paper: C. E. Forsyth III. 2026. Sensitive Data, Shared Cloud: Reviewing a Plan for a P3–P4 Research Environment on Google Cloud. Zenodo. doi:10.5281/zenodo.23890329

Abstract

UC Riverside has no shared place to compute on non-CUI research data at UC Protection Level 3 or 4, such as HIPAA data under agreements and FERPA records. Our plan puts a hardened VM per study in a Google Cloud project with no internet route, campus sign-on, steward-approved export and audit logs held by the campus security office. Nothing has been built. I checked the plan and its offline Terraform claim by claim against 19 peer offerings, HIPAA, FERPA, UC policy, Google's documentation, upstream source code and selected literature. Four lessons transfer. First, the defects that matter most are inherited vendor and open-source defaults (perimeter API names, edition-gated options, storage soft delete, live package mirrors). Two single-shot AI review panels that saw only the repository found none of them; AI agents given the same code plus the vendor and upstream pages found them. Second, in a shared project, study isolation is decided in the administrator and deployer plane; from Google's documentation, per-study customer-managed keys don't change that, because one service agent holds every key. Third, with a folder-level perimeter, a new study project joins the perimeter with no edit. Fourth, on my reading, FERPA's studies exception does not cover faculty-initiated research unless the institution commissions or adopts the study; the campus FERPA officer decides. Of 35 register items, 14 close by decision or document. For UCR, I recommend adding FERPA to intake, fixing code defects before a fake-data pilot, and settling funding, tenancy and the RStudio edition before first data.

1 Introduction

Most research universities can say where Controlled Unclassified Information (CUI) goes and where open research goes. The data in between is harder to place: HIPAA protected health information (PHI) that a researcher receives under a data use agreement (DUA), FERPA student education records, large personal-information sets, and anything an Institutional Review Board (IRB) or a contract puts at UC Protection Level 3 or 4 (P3–P4) [1]. At UC Riverside the medical school's systems serve its own clinical data, and the campus research cluster can't take the rest: our gap analysis found that the shared file systems, open outbound access and user-installed software that make an open cluster useful conflict with the segmentation, egress, logging and change-control rows of the UC IS-3 control list at P3 and above. That analysis is our own working

document. The campus security office hasn't reviewed it or its control list, which has 71 rows although its own text says 63.

So we wrote a plan for a separate environment on Google Cloud, now at version 3 (v3). It exists as a design document, Terraform planned offline with placeholder values, image scripts and runbooks. Before we build it, I checked it claim by claim against peer practice, the regulations, Google's documentation and upstream source code. This paper condenses the companion report that records that work. The four lessons in the abstract are argued where the evidence sits: inherited defaults and the reviews that missed them in Sections 3.1 and 6.1, administrator-plane isolation and folder-level tenancy in Section 5.3, and the FERPA route in Section 4.2.

The UC classification guide places ordinary student records at P3 but identifiable human-subject data, disability, medical or financial-aid information and comprehensive personal-information sets at P4, so a student-records study can't default to P3 [1]. UC is a HIPAA hybrid entity whose research function sits outside its health care component, so HIPAA reaches these studies through the data source's DUA or business associate agreement (BAA) rather than directly [2]. CUI and any data whose terms require NIST SP 800-171, including NIH controlled-access genomic data under NOT-OD-24-157, are out of scope [3]. They go to UCR's separate CUI enclave, scoped to 800-171 and Cybersecurity Maturity Model Certification (CMMC) Level 2 and built on Google Public Sector's publicly documented Stellar Engine [4]. The planned environment does not build on that enclave and makes no 800-171 claim.

2 The planned design

The plan treats the Google Cloud project as the boundary. A dedicated secure project, separate from the open cluster's project, holds every study virtual machine (VM), the network, the software mirrors and the alerts, and carries the organization policies, the VPC Service Controls (VPC-SC) perimeter and Data Access audit logs. A second project with no compute, administered by the campus security office (decision D11; Table 1 lists the decisions this paper uses), holds the audit log bucket with six-year retention and an irreversible lock to be applied after the pilot, the versioned Terraform state, teardown evidence and the billing export. Figure 1 draws the arrangement from the Terraform.

The network is one custom VPC in `us-central1` whose default internet route is deleted on creation, with no Cloud NAT. The only routes go to Google's restricted API virtual IP (VIP) range (`199.36.153.4/30`) and the Identity-Aware Proxy (IAP) forwarding range (`35.235.240.0/20`) [5, 6]. Five firewall rules allow SSH from IAP, egress on 443 to the restricted range and egress to the metadata server, and deny and log everything else; the ingress deny also blocks VM-to-VM traffic. Private DNS zones send `googleapis.com`, `pkg.dev` and `gcr.io` to the restricted address. Fourteen project-level organization policies (OS Login and Shielded VM required, no external IPs, no peering, no service-account keys, a location restriction and others) sit on top. A domain restriction to `ucr.edu` identities and the VPC-SC perimeter are written in Terraform but skipped in the offline plan, because the organization identifiers they need are open items.

Sign-in uses campus single sign-on with Duo (D2). The Terraform enforces nothing about multifactor authentication; the plan relies on campus enforcement. Each study has a Google group of the people on its signed Data Security Plan (DSP). The study module grants that group OS Login and IAP tunnel access on its own VM and service-account user on its own service account only. Administrators have project-level instance and OS-admin roles but no project-wide service-account user, so that an administrator would not start a VM as a study identity. Outside collaborators receive UCR affiliate accounts (D13).

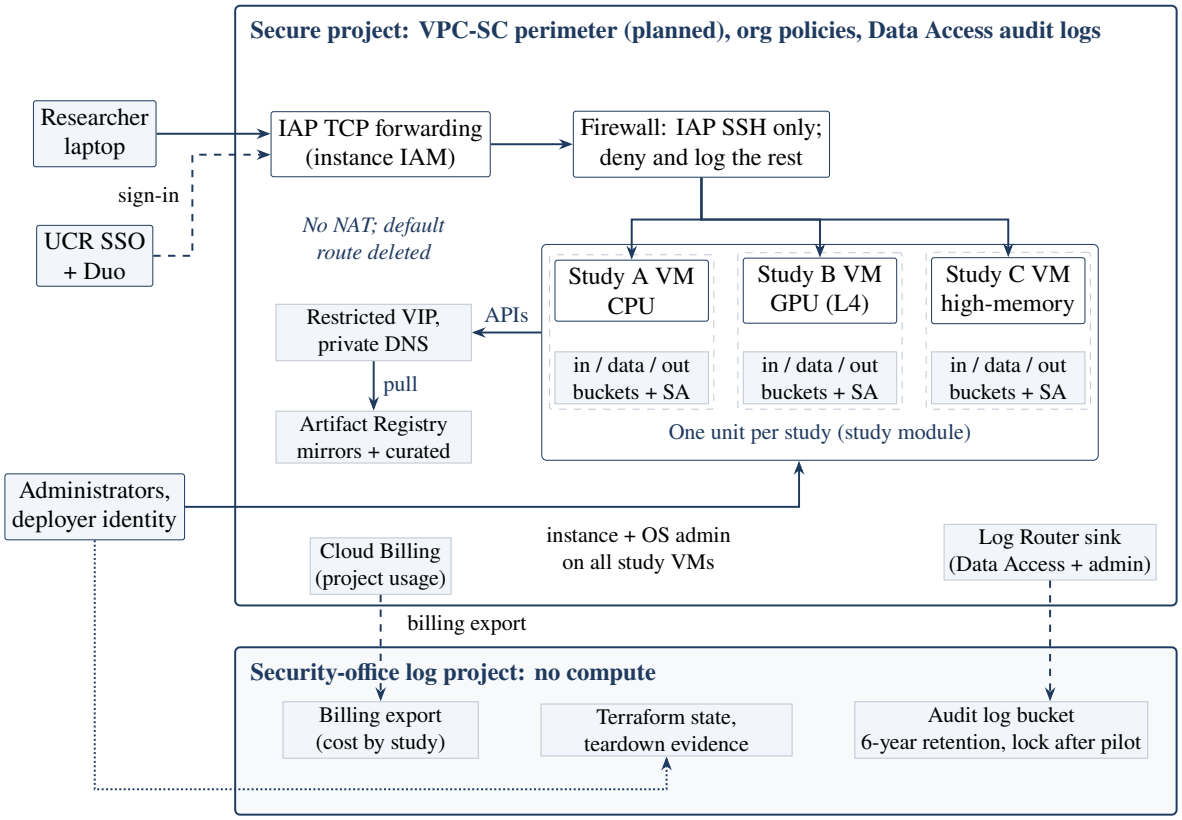


Figure 1: Planned architecture, redrawn from the companion report's Figure 5.1, after the foundation and secure-VM Terraform. Solid arrows are access and administration paths; the dashed arrow from the left is sign-in, the dashed arrows into the log project are the cross-project log sink and the billing export; the dotted arrow marks the deployment identity's access to the log project's state storage (Section 5.3). SA = service account.

One module call creates a study's VM, service account, three buckets (`in`, data and `out`), snapshot schedule and bindings: 19 or 20 resources per study. Three templates set the machine: CPU (`n2-standard-8`), GPU (`g2-standard-16` with one NVIDIA L4) and high-memory (`n2-highmem-32`), built from the Center for Internet Security (CIS) Hardened Image Level 1 for Rocky Linux 9. Researchers work in a shell, in RStudio Server or in a Linux desktop over a native Remote Desktop Protocol (RDP) client, all through one IAP-backed SSH tunnel (D10). A startup script runs checks at every boot (audit daemon and Ops Agent active, audit rules locked, SFTP off, the gate present, the data disk mounted, nothing but sshd listening beyond localhost) and writes `OK` only if all pass. A login-time hook refuses non-administrators unless the status reads `OK`. A `gui_only` mode with no shell, file copy or clipboard serves studies whose agreement forbids copy-out, while the default shell mode accepts copy-out as a logged residual risk (D12). An idle timer stops VMs after 120 minutes without activity.

Data enters through the `in` bucket and leaves through `out` after a data steward approves it against a checklist with a default cell threshold of 11. Five Artifact Registry remote repositories proxy PyPI, Docker Hub, Rocky 9 BaseOS and AppStream, and EPEL 9; a sixth holds curated containers built outside the perimeter. Nine alert policies notify the systems team and the campus security office.

Table 1: The plan's decisions that this paper refers to, in brief (condensed from the companion report's Table 1.2).

ID	Decision
D1	UCR has a signed Google Cloud BAA; only services the BAA covers may touch PHI.
D2	Duo is enforced at campus sign-in, and the plan relies on it for the IS-3 MFA requirement; no plan code enforces it.
D3	One shared project serves all studies; no project or cluster per study.
D4	Google's default encryption at rest, no customer-managed keys; a risk for the security officer to accept.
D8	Funding: possibly subsidized under the campus no-charge tier; open.
D10	RStudio Server and a Linux desktop over IAP, the desktop through a native RDP client.
D11	A separate log project with no compute, administered by the campus security office, holds audit logs and Terraform state.
D12	Shell mode accepts copy-out risk (prohibited, logged, accepted by the security officer); `gui_only` mode serves studies that forbid copy-out.
D13	Outside collaborators receive UCR affiliate accounts under the same DSP.

Table 2: The plan's fifteen pilot tests on fake data (not yet run), numbered as in the plan.

1. Login: campus SSO and Duo through IAP, no Google 2-Step prompt; the OS Login user has no sudo.
2. GPU: Secure Boot and the NVIDIA driver boot under the required Shielded VM settings; a CUDA test runs.
3. Network: no internet; mirrors work; IAP works without a default route; no VM-to-VM traffic.
4. Perimeter: a week in dry run, then enforce and repeat test 3.
5. Buckets: `in`, `out` and steward permissions; the export alert; version lifecycle.
6. Fail closed: break a boot check; researcher logins refused, administrators still in; the fix reopens.
7. Alerts fire: status and privilege alerts; a sink change alerts the security office.
8. Audit rules stay locked until reboot.
9. RStudio and desktop: the GUI password helper works; clipboard and drive redirection blocked; ports on localhost only.
10. `gui_only`: the shell is replaced by the password helper; copy commands fail; the tunnel works; RStudio download off.
11. SFTP clients cannot connect.
12. Idle stop: an idle VM stops; active jobs and a keep-running marker keep it up.
13. Restore from the study's own snapshots within the recovery time objective; another study's disk can't be attached.
14. A 200 GB upload into `in` is timed and the result published.
15. Teardown: dry run, then the real run, with the evidence written.

What exists is a design document, Terraform that passes `terraform validate` and plans offline (100 resources for the shared foundation, 59 for a three-study pilot), image-build scripts that have never run, twelve runbooks and templates. The plan text says the foundation has 101 resources; the regenerated plan has 100. Because the perimeter and the domain restriction are absent from the offline plan, the 100-resource count describes the network, policies, mirrors, logging and alerting, and overstates how much of the design has been exercised. The plan estimates three to four weeks for its first six build steps and groups security-office review, enforcement, the log lock, a penetration test and the first study in a seventh. Step 5 is a pilot on fake data with fifteen acceptance tests (Table 2); I refer to them by the plan's numbers throughout.

3 How I checked the plan

I froze the plan at one repository commit before writing began. "Checked" in this paper means a claim-by-claim comparison against a primary source. Nothing was run, so every statement about behavior is a prediction from code.

I commissioned eleven reports from an AI-assisted literature review tool on 1 November 2026, one per question: peer enclaves, Google Cloud reference architectures, control frameworks, data ingress and egress, researcher experience, offline software supply, logging and incident response, cost and funding, multi-tenancy,

GPUs and AI, and FERPA. They ran as 43 sessions and are archived in full. I used them as leads only. They state unverified figures in the same confident, cited style as verified ones, and Table 3 lists some of the errors I found in them. A rerunnable script also ran 31 queries across ten topics against the arXiv and Zenodo APIs; it returned 481 records, which I screened by hand to 32. I read those at abstract level and cite them only for what their abstracts state.

Claims that would change the plan or a conclusion I checked against primary text: the eCFR for HIPAA (45 CFR 160 and 164) and FERPA (34 CFR 99), the Federal Register and Unified Agenda, NIH notices, UC policy pages, Google Cloud documentation and price pages, the Open Source Vulnerabilities (OSV) database, vendor and upstream sources (Posit, xrdp, OpenSSH, Globus, the Rocky and EPEL packages) and each peer institution's own pages. The peer set came from institutions named in one AI-assisted report; I read each on its own pages (Columbia only through committee minutes). It is not a systematic sample, and peer counts describe this set only. Where a page could not be read or did not say what a report claimed, I label the claim as reported or leave it out.

Before this work the plan had been through two single-shot panels of AI reviewers, run through a campus AI gateway: six lenses on v1 and ten on v2, with each finding checked against the files. The first scored every lens "not ready". The second scored compliance 5, DSP process 4, images and scripts 2 and the rest 3. After the companion report was drafted, a five-reviewer claims panel checked it against primary sources.

The views and readings here are my own and do not represent the University of California, Riverside or the University of California. The campus security office, systems team and leadership have not reviewed or approved this paper or its companion report, and nothing in it is UC policy or a commitment by UCR. The plan, the design decisions and the conclusions are mine and my team's. I used AI writing agents (Claude Sonnet 5.5) to draft the companion report's chapters from my brief and Claude Opus 5.5 to assemble and condense them, including into this paper, and AI reviewer panels to check the drafts against primary sources. I checked the primary sources behind every correction and am responsible for every claim.

The work has limits. Peer pages can lag or overstate practice, and several I read through a text extractor. Prices are list prices on stated dates, without UCR's negotiated terms. The panels are internal checks by AI reviewers, not independent audits, and residual errors are possible. The comparison behind the first lesson is confounded: the reviews that found the inherited defaults had the vendor and upstream documents, and the panels that missed them did not. Every reviewer other than me was an AI system, and no human repository-only review was run, so the comparison says nothing about how a human reviewer working from the repository alone would do.

3.1 How each defect was found

The two repository-only panels found real problems that were visible in the code (a startup script that hid failures, bucket roles that were too broad, audit rules root could change), and those drove the fourteen design changes between v1 and v3. None of their accepted findings is still open, so they have no register rows. Table 4 sorts the 35 rows of the companion report's register by how each was found. Twenty-one of the thirty-five register rows needed an outside document, and none of those twenty-one appears as a finding in either repository-only verdict. Most came from my primary-source check while drafting; the claims panel, which read the code beside Google's pages and the upstream distribution sources, found eight more v3 defects, five of them needing an outside page. Both repository-only panels also reported things that were wrong: in the second, five of twelve Terraform-lens items failed on Google Cloud mechanics, and in the first about a

Table 3: Selected corrections from primary-source checks: errors in the commissioned reports, in our gap analysis and in the plan's own text.

Claim and where	What the primary source says
The CUI enclave's platform is the vehicle for this P3–P4 design (reports)	It built UCR's separate CUI enclave [4]
Harvard FASSE is rated Level 4/5; Nero is about $100/month (peer-enclaves report)	FASSE is Level 3 only; Nero lists about $290/month and about $20/TB [7, 8]
NIH accepts 800-171 "or equivalent" (control-frameworks report)	ISO 27001/27002 only for non-US users who cannot attest [3]
HIPAA Security Rule rewrite "slipped or dropped" (gap analysis)	Still a proposal; final action listed for 07/2027 [9]
The Centers for Medicare & Medicaid Services (CMS) bar co-hosting; 90-day destruction (gap analysis)	Terms restrict linkage and require destruction; no absolute ban found; 90 days covers shipped media [10, 11]
A third of UK applicants waited over 12 months (report)	Not found; Administrative Data Research UK (ADR UK): over a third waited over six months [12]
HIPAA requires six years of locked raw logs (logging report)	164.316(b)(2) governs documentation retention [13]
Center for Internet Security (CIS) image fee per vCPU-hour; ±15% corridor in 200.468 (cost report)	Per instance-hour; no such figure in the eCFR text [14, 15]
Foundation plan has 101 resources (plan); only the security office reaches the log project (plan)	The plan has 100; the deployment identity can also reach the log project's state storage

third of the security-lens high-severity items were wrong. The fix is cheap: hand the panels the vendor pages too, and add one lens whose only job is to check each setting against its vendor's current page (Section 7).

4 What peers and the rules require

4.1 Peer universities

I read 19 offerings at 14 institutions on their own pages on 1 November 2026. Eleven state their hosting: six are cloud-hosted (Stanford Nero, Duke School of Medicine Azure, UCSF AWS Secure Enterprise Cloud, UC San Diego Sherlock, Johns Hopkins SAFE Desktop, NYU SRDE) and five are on premises (Stanford Carina, Duke RCC, Harvard FASSE, the Utah Protected Environment, UCLA HCE) (Figure 2). Only Nero is documented as running on Google Cloud, and it gives each requesting team its own project [16–18]; Sherlock lists Google Cloud among its supported clouds [19]. No peer page describes one shared project for all studies, although Utah describes a shared environment with groups separated by subnets, VLANs and Unix controls [20].

Every peer whose page describes onboarding puts a human review before data lands. Stanford requires a Data Risk Assessment; Harvard reviews the DUA or IRB documents before a FASSE project exists; NYU requires IRB approval and a named Data Steward; UCLA assesses each application; Yale grants access per project on a PI's request [7, 17, 21–23]. We'd gate at the same point: an intake checklist and a signed DSP. Columbia offers a central "Data Security Officer as a Service" for teams with no qualified person of their own [24]; we have a per-study steward and no fallback. UCSF, Johns Hopkins and Utah offer Windows desktops [20, 25, 26], and Stanford lists SAS and Stata [27]; our plan mentions neither. On cost, most cloud-hosted

Table 4: How each register row was found, and whether finding it needed a document from outside the repository (a vendor, upstream or regulatory page, a price page or an agreement). No = nothing in the repository implies the defect; Yes = deciding it needs a fact stated only in an outside document. The classification is mine. Condensed from the companion report's Table 14.3.

Found by	Outside document needed	Register rows	Rows
Plan's open-items list (also in the v1 panel's process list), sharpened while checking	No	R01, R02	2
My reading of the plan and code while drafting	No	R03, R05, R09, R11, R15, R20, R23, R28, R30	9
My primary-source check while drafting, against a vendor, upstream or regulatory page	Yes	R04, R07, R08, R13, R14, R16, R17, R18, R19, R21, R22, R24, R25, R26, R27, R29	16
Claims panel on the draft, reading the code	No	R12, R31, R35	3
Claims panel on the draft, reading against Google's or upstream pages	Yes	R06, R10, R32, R33, R34	5
Repository-only AI panels on v1 and v2 (verdicts)	—	No open rows; accepted findings fixed before v3. Rejected: 5 of 12 Terraform-lens items (v2); about a third of security-lens high-severity items (v1)	0
Total	Yes 21, no 14		35

peers with a stated price pass costs through. Nero bills the PI at Google list less at least 18%, with network fees waived, and lists about $290 a month for 8 vCPU and 30 GB [8]. Most subsidized peers run on premises; Johns Hopkins SAFE Desktop, cloud-based and free to 100 GB, is the exception [26]. Table 5 condenses the rows that bear most on the plan.

4.2 HIPAA, FERPA and IS-3

FERPA is in scope and absent from our plan. Its studies exception in 99.31(a)(6) covers studies "for, or on behalf of" the educational agency or institution for three purposes: developing predictive tests, administering student aid, and improving instruction [31]. On my reading of 34 CFR 99.31(a)(6), FERPA's studies exception does not cover research a faculty member initiates unless the institution commissions or adopts the study; the campus FERPA officer decides. IRB approval does not supply the missing authority. That leaves consent (99.30), de-identified data (99.31(b)) or a school-official function, so intake has to record which route applies and the agreement's destruction date. FERPA has no breach-notification rule; notice duties come from state law and the agreements [32]. Where data include federal student-aid information, the institution's Gramm-Leach-Bliley Act (GLBA) safeguards and its duty to report breaches to Federal Student Aid also apply [33].

For HIPAA data, the duties arrive through the DUA or BAA (Section 1). Notice to individuals is due without unreasonable delay, with an outer limit of 60 calendar days after discovery (164.404(b)), and a business associate must tell the covered entity within the same outer limit (164.410); DUAs are often shorter [34]. HHS's proposed Security Rule rewrite (RIN 0945-AA22, published 2025-01-06) is still a proposal, with final action listed for July 2027 [9, 35, 36]; IS-3 already requires most of what it adds at P3 and above [37], so

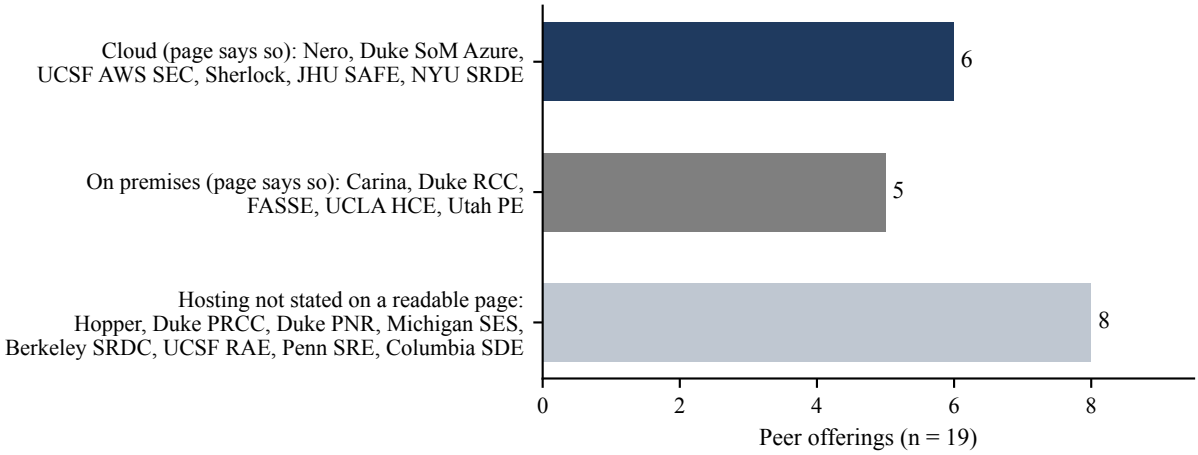


Figure 2: Hosting of the 19 peer offerings as stated on each institution's own page (read 2026-11-01). Rows whose pages name no platform are counted separately. Redrawn from the companion report's Figure 2.1.

Table 5: Peer offerings that bear most on the plan, as stated on each institution's page (read 2026-11-01). "ns" means not stated. Condensed from the companion report's Tables 2.1 and 2.2.

Offering	Hosting	Cost to researcher	Gate before data
Stanford Nero [8]	Google Cloud, project per team	List less at least 18%; about $290/month example	Data Risk Assessment
Stanford Carina [28, 29]	On premises	Free; 1 TB per project	Data Risk Assessment
Harvard FASSE [7]	On premises	Not found	Review of DUA and IRB
Yale Hopper [23]	ns	No service-unit charge from 2026-07-01	PI request; annual training
Michigan SES [30]	ns	Cost recovery per GB RAM and TB	Not found
JHU SAFE Desktop [26]	Cloud	Free to 100 GB	IRB and data trust criteria
NYU SRDE [21]	Cloud	Not found	IRB; named Data Steward
Utah PE [20]	On premises, shared	Not found	Project request
UCSD Sherlock [19]	AWS, Azure, GCP or SDSC	Fee for service	Contract
UCLA HCE [22]	On premises	No cost to campus for accepted projects	Risk assessment
UCSF RAE [25]	ns	Free tier; $300–$1,300 premium VMs	Service request

the plan should track those items as met or as recorded gaps without citing them as law. Google's BAA and HIPAA guidance make compliance a shared responsibility and tell customers not to use pre-GA offerings with PHI unless a notice or the offering's terms say otherwise [38]; Section 5.1 applies that to the plan.

4.3 Related frameworks

The Turing Data Safe Haven is the nearest public analogue to our engineering approach: an open-source, infrastructure-as-code trusted research environment on Azure with five sensitivity tiers, each with default technical controls [39, 40]. At tiers 2 and 3 outbound internet is blocked, copy-paste and file transfer to the

researcher's device are off, and data entering or leaving needs the sign-off of three parties; at tier 3 only a pre-agreed allowlist of PyPI and CRAN packages is available [40]. The Turing does not yet support tier 4. Our plan aims at tiers 3–4 by intent, with no internet route and steward-approved export, but two of its choices sit lower: shell mode accepts copy-out as a stated risk (D12), and live package mirrors match the Turing's tier-2 rule, not its tier-3 allowlist.

SATRE, the Standard Architecture for Trusted Research Environments, is a UK community specification built with more than 60 organizations [41]. I read the items our findings touch [42]. It requires tamper-proof security logs that capture authentication, privileged actions, configuration changes and data egress (item 2.5.03), which is the requirement our alert-source and evidence-store findings fall under (Section 5.7). It lists access to public package repositories or an internal mirror as an optional capability (2.1.12), and asks that outputs pass governance and output checks (1.4.07). It doesn't say how to verify that a configured control does what its vendor's page says it does. The Five Safes framework maps cleanly onto the plan (signed DSP and training for safe people, intake for safe projects, the no-internet project for safe settings, classification for safe data, the steward for safe outputs) [43], and is useful mainly as an intake checklist.

NIST SP 800-223 divides an HPC system into four function zones: high-performance computing, data storage, access and management [44]. Our access zone is IAP and the firewall, our computing zone the study VMs, our storage zone the per-study buckets. Our two-project split moves part of the management zone (logs, state, evidence) out of the compute project. SP 800-223 notes that a management zone run as a cloud service takes on cloud risks, and names multi-tenant exploitation as a major threat in the computing zone. Its cloud appendix recommends VPCs between zones and least privilege at each layer, but it treats isolation at the zone level, not the case where one deployment identity spans every tenant in a cloud project. As far as their published text goes, none of the three frameworks I read addresses vendor-default verification or admin-plane isolation in a shared cloud project.

5 Findings by area

Of the twelve dimensions in Table 6, five have gaps; the findings below fall into ten areas. Hosting, identity and the operating model are consistent with what peer pages describe. Network, data out and logging are consistent with vendor guidance, but peer pages say too little about them to compare. Tenancy departs from the one Google Cloud peer.

5.1 Google Cloud building blocks and the BAA

Every data-bearing service the plan uses appears on Google's HIPAA BAA list: Compute Engine, Persistent Disk, Cloud Storage, Artifact Registry, Cloud Logging, Cloud Monitoring, IAP, VPC Service Controls, Cloud DNS and Pub/Sub [38]. OS Login, VM Manager, the organization policy service and billing budgets are not on it by name. D1 says only BAA-covered services may touch PHI; restating it to cover data-bearing services, and asking the campus security office and counsel about the administrative ones, would settle that with a decision. The Yum remotes we planned on Artifact Registry are Preview [45], Conda is Preview and CRAN has no remote [46], and Google's guidance says not to use pre-GA offerings with PHI unless a notice says otherwise [38]. No PHI would pass through a package cache, so the rule may not apply; the security office should record whether it accepts a pre-GA dependency in the patching path.

VPC-SC protects Google API calls, not network paths: Google's page says SSH connections to VMs are not protected, hierarchical firewalls are unaffected and peering operations don't enforce perimeter restrictions [47]. The plan answers each with a separate control (deny-all egress, a no-peering policy, IAP as the only

Table 6: The plan on twelve dimensions, with my reading in the last column; the campus security office has not rated it. Condensed from the companion report's Table 14.1.

Dimension	Plan (v3)	Reading
Hosting and boundary	Dedicated secure project plus security-office log project; no internet route	Consistent with peer pages; little Google Cloud precedent
Network and perimeter	Deleted default route, restricted VIP, private DNS, VPC-SC	Consistent with vendor guidance; no peer comparator; perimeter in no plan output
Identity and access	SSO with Duo, IAP, OS Login, per-study groups	Consistent with peer pages; waiting time unmeasured
Tenancy	One shared project; per-study groups, accounts, buckets	Departs from the one Google Cloud peer; reversal conditions undocumented
Data in	Upload to a per-study bucket; Globus not planned	Gap: no ingress procedure
Data out	Steward, checklist, threshold 11, stated copy-out risk (D12)	Consistent with vendor guidance; no peer comparator; gaps on models, zeros, dominance
Software	Live PyPI and Docker Hub mirrors; curated containers	Gap: no admission decision
Logging and response	Data Access logs, cross-project sink, nine alerts, locked audit rules	Consistent with vendor guidance; no peer comparator; coverage gaps
Operating model	Responsibility matrix with one accountable party per row	Consistent with peer pages in shape; data owner and systems-team acceptance open
Cost to researcher	D8 "possibly subsidized" against a pay-your-own rule	Gap: internal inconsistency
Student records	HIPAA named; FERPA not named	Gap in intake and breach matrix
GPUs and AI	One L4 template; no weight path or managed-AI position	Gap, but narrow

inbound path), so the perimeter and the firewall have to be reviewed as one design. Reading the v3 perimeter beside Google's pages turned up three kinds of mismatch. First, a service's administrative API and the API that carries its traffic can have different names, so restricting one can leave the traffic governed only by instance IAM; a product protected by several APIs needs all of them in the perimeter [6, 47]. Second, route and egress lists for the restricted endpoint have to match Google's current configuration table in full [5]. Third, every ingress rule should carry the campus access level once it exists. The dry-run week (test 4) also has to settle two open questions: the log sink writes to a project outside the perimeter with no egress rule, and no ingress rule yet admits Storage API calls from study members and stewards. And each mirror's upstream path has to be checked against the live repository. A wrong path fails only at image build, and the image silently goes without the packages it should have had.

5.2 Architecture and the login gate

Version 2 would have alerted on a failed boot check while still admitting researchers. Version 3 refuses non-administrator logins until the checks pass, which turns a monitoring signal into an access control. It is a short startup script and a login hook, and neither has run on a real VM. The gate is intended to close early in the boot, but the ordering leaves a window. The status survives a reboot, and a startup script runs only when the guest environment gets around to it, so the login service can come up first and admit researchers against the previous boot's `OK`; a step that hangs before the status is reset holds that window open. Writing the failure first in the script doesn't close it on its own, because the login service can still start before the script does. A boot-scoped status does: keep it in memory-backed storage that is empty at every boot, or bind it to the

current boot identifier, so a status from an earlier boot never counts. Test 6 should then break a boot step on purpose and confirm that only an administrator can get in. The gate is also evaluated only at boot and login, so a control that breaks later isn't noticed until the next boot, and no alert covers a change to a VM's integrity settings while administrators hold the rights to make one.

The plan tags several behaviors "verify in pilot": first boot, Secure Boot with the NVIDIA driver (test 2), IAP with no default route (test 3), Duo without Google 2-Step Verification (test 1) and the GUI password helper (test 9). A status that reports a resource count hides this. A readiness table that separates "in Terraform and planned", "in Terraform but skipped for missing identifiers" and "process only" would show it.

5.3 Tenancy: one shared project, or a folder of study projects

D3 puts all studies in one project, chosen for the smallest build; D4 keeps Google's default encryption. Both await the security officer's acceptance. The study module separates researchers well: they hold resource-level roles on their own study and can't create or snapshot disks or attach service accounts. The cross-study exposure is in the administrator and deployer roles. Administrators can reach study identities indirectly through the deployment identity, and that identity can also reach the storage where teardown evidence is kept, so the people who run the VMs could also alter the proof that a study was destroyed. Our statements that "administrators cannot act as a study identity" and "only the campus security office has roles in the log project" hold by process, not by code [48]. I would limit use of the deployment identity to a break-glass group with an alert on every use and move evidence to a locked bucket the security office alone controls; then both statements become checkable. The pilot also lacks a test that a member of one study cannot read another study's bucket, and a per-study network tag that no firewall rule references separates nothing.

Per-study customer-managed keys would not fix this. In one project, the single Compute Engine service agent holds the encrypter and decrypter role on every study's key, and Google's CMEK page says a requester doesn't need key access as long as the service agent has it [49, 50]. On the documented behavior, a principal able to snapshot or attach another study's disk would therefore need no key permission. This follows from Google's CMEK documentation; I haven't tested it, and a cross-study attach test belongs in the pilot. CMEK would add key-use logging, a destruction control and an answer about key custody for data providers. Separation by key needs a project per study.

I read the provider agreements I could get. The CMS Research Identifiable File agreement bars linkage without authorization, requires need-to-know access, and requires destruction of original, derivative and backup files; its policy guide applies a 90-day rule to shipped media. Neither bans co-hosting outright [10, 11]. I didn't read other providers' terms, so intake should ask each provider about separation language, and the plan should keep written triggers that would move a study into its own project: separation or dedicated-environment language, a key-custody requirement, an unenforceable linkage clause, different administrators, a separate billing account, or a log-destruction clause.

The companion report sets out a real alternative, Option C: one folder holding a small platform project and one project per study, with organization policies, the log sink and one VPC-SC perimeter set on the folder. Google documents that projects created in or moved into a perimeter folder join the perimeter automatically [51], so adding a study needs no perimeter edit, at roughly twice the shared design's per-study resources (an estimate, not a Terraform plan) and a larger one-time build. Table 7 compares the options. Option C does not separate studies at the API layer, a person on two studies still links them, and folder-level administrator grants would recreate A's reach. The deployer problem and the perimeter-list mismatches carry into C unchanged, so whichever layout we choose, the deployer grant has to be narrowed and a cross-study negative test added.

Table 7: Tenancy options. A is the plan; B and C are proposals. Per-study resource counts for B and C are estimates from the offline plan, not Terraform plans. Condensed from the companion report's Table 6.4.

Criterion	A. Shared project (plan)	B. Project per study	C. Folder, project per study
Mistaken project-level grant	Reaches all studies	Reaches one study	Reaches one study
Administrator reach	All studies through project roles	One study per project role; platform and vending identities span all	One study if admin roles are per project; platform, vending and folder admins span all
Per-study footprint	About 20 resources	Up to about 100 if the foundation is repeated	Roughly 40 (estimate)
Build effort	Lowest	Highest; vending pipeline	Middle; folder rights, Shared VPC, vending module
Provider requires separation	Fails that study	Meets it	Meets a separate-project requirement
Disposal evidence	Label-driven teardown	Project deletion (30-day window)	As B

Our position is still D3, but leadership is now choosing it against a cheaper alternative than per-study projects built from scratch.

5.4 Data in and data out

The plan has no ingress procedure: where files arrive from a provider, who may touch them, a checksum step, and when `in` is cleared after loading. Globus supports high-assurance collections for PHI and needs a BAA [52]. I haven't confirmed whether UCR has a subscription or whether an endpoint can run with no internet route.

On the way out, anyone with a shell can copy data to a laptop, so the `out` bucket is an approved path for authorized users, not a wall. Three controls on that path depend on other people's software defaults. The `gui_only` mode relies on RStudio options that only the paid Posit Workbench honors; in the open-source build each option's accessor returns true whatever the configuration file says [53, 54]. So in `gui_only` the SSH route and remote-desktop channels are closed, but the web IDE would still offer its own export routes, and a DUA that forbids copy-out shouldn't rely on RStudio in that mode until we choose the edition. The remote-desktop server supports granular clipboard restrictions that the planned configuration doesn't set; whether channel settings alone block every path is test 9 [55].

Teardown could report "remaining=none" while copies remain. The script's closing check lists resources by study label, and Cloud Storage soft delete keeps deleted objects restorable for seven days by default [56]. The Department of Education's guidance says deleting a digital record is typically insufficient for disposal [57]. Setting soft-delete retention to zero on study buckets (or waiting out the window and checking again), checking for unlabeled resources, and recording the setting in the evidence would close it. The release checklist also needs items. Its "under 11" rule treats zeros differently from the UK Office for National Statistics (ONS) and CMS rules [58, 59], and it has no dominance item, no FERPA 99.31(b) test and no running release log. It has no rule for trained models either, though released models may leak training data and the pilot includes a GPU study [60, 61]. I'd default to no export of a model trained on P3–P4 data without review.

5.5 Software supply

Two 2026 releases of PyTorch Lightning on PyPI (2.6.2 and 2.6.3) were compromised. OSV records the compromise; the code ran on import, downloaded a JavaScript runtime and went after credentials, including cloud metadata endpoints and `gcloud` [62–64]. The fixed version arrived 20 days after the advisory date [65]. With live remote repositories, the first `pip install` inside the enclave that resolved to a bad version would have fetched and cached it, and a build of the curated PyTorch container during the window would have run it on the builder, whose credential can push to the curated repository. Nobody published the exposure window, so I can't say whether a seven-day minimum age would have caught it. I can say nothing in the plan would have. Separately, CVE-2024-27322 allows code execution from crafted R data serialization (RDS) files on R before 4.4.0 [66]; EPEL carries 4.6.1 today [67], but nothing in the build pins it.

Two detection papers report high precision and recall on their own benchmark sets [68, 69]; a third exists because scanners miss behavior-preserving rewrites of the same malware [70]. Our plan already accepts request friction for R and conda, where Artifact Registry has no remote, and avoids it for PyPI and Docker Hub, where it does. So the weakest admission policy governs the ecosystem with the best-documented 2026 incidents. I'd rather have a rule that every new package waits a set time than a better scanner: a quarantine window on mirrored packages, a curated allowlist for the strictest studies, a failing build on R below 4.4.0, and separate identities for building and promoting artifacts. The window (7 to 14 days is my suggestion, not an evidence-based value) costs researchers waiting time, which the pilot should measure. The build has the usual holes as well: installers fetched and run without a checksum, no signature check on what goes into the image, one curated image with no recipe, and VMs with no permission to read the registry they pull from.

5.6 Researcher experience

The best published evidence is about approval delay, not tools. The interim evaluation of Administrative Data Research UK found that more than a third of academic respondents waited more than six months for data, that the Secure Anonymised Information Linkage (SAIL) databank averaged 8.4 weeks from application to access, and that researchers who got access mostly found the environments easy to use [12]. Nobody has published the interval we control, from approved DSP to first working session.

Reading the Terraform beside Google's documentation gave three concrete checks. The tunnel command sets no client keepalives (the wrapper isn't written yet), so a dead tunnel hangs instead of failing fast [71]; IAP disconnects after one hour of inactivity, and whether SSH keepalives count is untested [6]. Inactivity detection must cover the identity provider's user-ID range, and as planned it doesn't. The idle check only counts users in the range a local install would use, and OS Login hands out IDs far outside it [72], so detached jobs probably don't keep a VM awake and VMs may stop under running work. That has to be fixed before any pilot timing means anything. And the study group has no permission to start a stopped VM, and a researcher the gate refuses sees no message. None of the fifteen pilot tests measures researcher time or satisfaction; I'd add time to first login and package and export turnaround. A low-fidelity synthetic dataset outside the perimeter would let code be debugged before it enters the slowest working conditions [73].

5.7 Logging and incident response

The structure for tamper resistance is right: a separate log project, Data Access logs turned on in Terraform, a retention-locked bucket and alerts on changes to both projects. HIPAA's 164.316(b)(2)(i) fixes six years for documentation (policies, assessments, review records, incident files) and names no period for raw logs [13], so six years of raw logs is our choice, defensible as a conservative default. Our 2,190 days falls two days short

Table 8: Monthly list-price cost of one study VM, USD, Iowa, on demand, prices read 2026-11-01; CIS fee at the low value (none on GPU); snapshots and buckets use the companion report's stated assumptions (its Table 12.1). Components are rounded to whole dollars and totals are computed before rounding, so a total can differ by $1 from the sum of its displayed parts.

Template	Machine	VM	CIS	Disks	Snap. + buckets	24x7	40% uptime
CPU	`n2-standard-8`	284	16	55	28	383	203
GPU	`g2-standard-16`	837	0	110	56	1,003	501
High-memory	`n2-highmem-32`	1,530	16	105	54	1,706	778

of six calendar years, and the lock is irreversible. Meanwhile the records that are documentation (review records, incident files, teardown evidence) have no tamper-resistant home, and the log project has no deletion lien.

Coverage has gaps. One planned alert's log source does not match where the OS writes the events: the alert looks in one log file while the OS writes those events to another [74, 75]. The records still reach the audit log, but the alert would stay silent. Logs can rebut HIPAA's breach presumption only if they exist and arrive: the entity carries the burden of proof [76, 77], Data Access logs can show which bucket object a principal read but not which file on a VM disk, and a sink outage leaves records only on the VM. The evidence home and the mis-aimed alert should both be fixed before the lock, because a locked bucket missing the records an investigator needs can't be corrected afterwards. UC's standard requires the Location's incident response plan to be tested at least yearly [78]; our plan doesn't say how this environment would take part. I'd make a tabletop a gate before first PHI and sign a breach matrix that includes a FERPA row.

5.8 Operating model

Our responsibility (RACI: responsible, accountable, consulted, informed) matrix gives each of 30 activities one accountable party. Five rows depend on an accountable data owner the plan hasn't named, and seven on the systems team accepting an operating role the plan marks as unconfirmed. The campus security office carries the most rows (13 as responsible, eight as accountable), and no deputies are named for the weekly review, the on-call primary or the data steward. Rows are missing for appointing stewards, owning the Google groups, the HR separation feed, the restore test and a teardown hold during an incident; the end-date reminder is a label, not a job. No verified staffing ratio exists. The figures in circulation come from one report's assumed request rate and productivity, and varying those two inputs across plausible ranges changes the export-review workload at 50 studies by a factor of about 15. Staffing should come from hours recorded in the pilot.

5.9 Cost and funding

I recomputed each template from Google's list prices for Iowa on 2026-11-01, on demand, over 730 hours [14, 79–81]. On the plan's own basis (VM, disks and CIS image fee, continuous) the CPU template matches the plan at $355 a month and the GPU template comes to $947 against the plan's $910; the plan's high-memory figure ($1,290) is about a fifth too low against $1,652, and below the `n2-highmem-32` charge alone. Table 8 adds snapshots and buckets. Idle stop at 40% uptime cuts the CPU template from $383 to $203 (47%), less than the uptime cut, because disks, snapshots and buckets bill while the VM is stopped [82] (Figure 3).

The shared platform's cloud bill is small, roughly $6 to $102 a month before Security Command Center Premium and any shared file system. What would set a rate is staff time: security-office review, systems-team

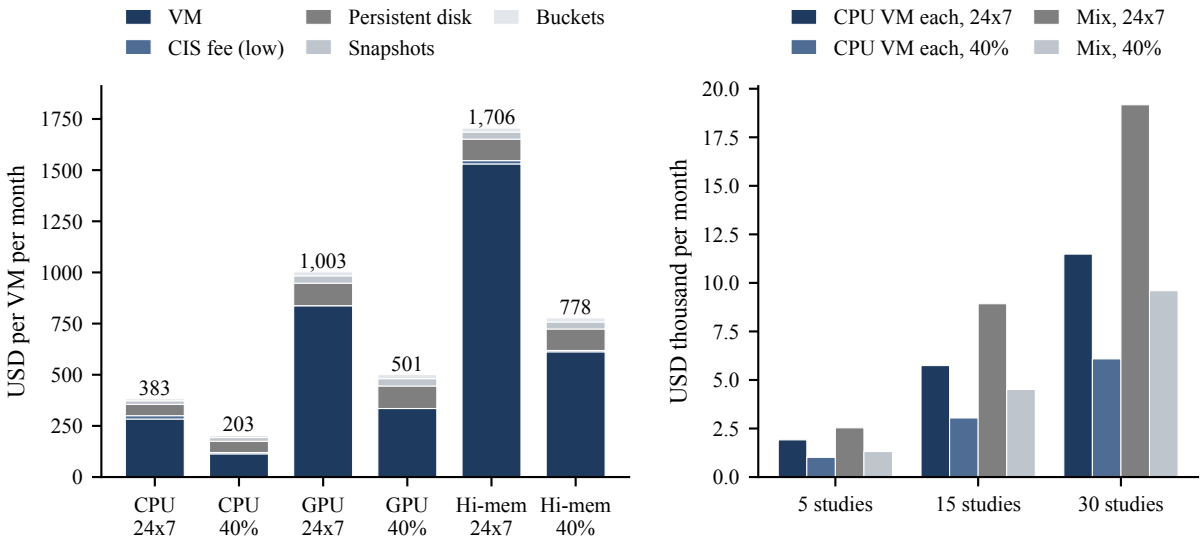


Figure 3: Left: monthly list-price cost of one VM per template at continuous operation and 40% uptime, by component. Right: monthly cost of a subsidy covering VMs, disks, snapshots and buckets for 5, 15 or 30 studies, with one CPU VM each or with a mixed fleet I assumed to show sensitivity (4 CPU and 1 GPU; 11 CPU, 3 GPU and 1 high-memory; 21 CPU, 6 GPU and 3 high-memory). Platform fixed costs excluded. Redrawn from the companion report's Figure 12.1.

operation, steward effort, the penetration test and incident response, none of which the plan prices. D8 ("possibly subsidized" through a campus no-charge tier) conflicts with the stated rule that P3–P4 projects pay their own costs through a chart of accounts, a memorandum of understanding and a separate billing account, and the Terraform's $2,000 monthly budget is a placeholder that only sends alerts [83]. If run as a specialized service facility, a campus recharge would fall under 2 CFR 200.468, which requires one rate schedule for federal and other users, tied to usage [15]; how UC applies it is a question for the campus research-administration office. Until someone decides who funds review, operations and incident response, a researcher asking what the service costs can't get an answer.

5.10 GPUs and AI

The GPU template has one 24 GB NVIDIA L4 [84]. By my arithmetic an 8B model such as Qwen3-8B fits at full precision and a 32B model only at 4-bit [85, 86]. The L4 cannot run as a Confidential VM [87, 88]. No GPU quota has been requested for the secure project, the plan has no path for model weights into an offline VM, and idle stop ignores GPU use. Model Garden is not named on Google's BAA list [38], so managed AI would need the campus security office's written position. A local model needs nothing from Google beyond what the plan already uses, and I'd offer it first, with pinned, checksummed weights. Phase 2, a shared cluster, reuses the Unix-permission file system our own gap analysis rejected, on a Filestore tier without Kerberos, and should wait for demand and a storage design.

6 Checks, risks and roadmap

Table 9: Inherited-default checklist: one row per kind of defect found in our plan, the document that settles it, a generic check, and whether a review of the repository alone could find it. Repo-only?: No = nothing in the repository implies the defect; deciding it needs a fact stated only in an outside document. Condensed from the companion report's Table 14.4.

Defect class	Document that settles it	Generic check	Repo-only?
Perimeter service names vs the API each access path calls	Vendor's perimeter supported-products page [47]	For each access path, list the API it calls and confirm that name, not only the admin API, is restricted	No
Restricted-endpoint routes and egress	Vendor's restricted-VIP configuration table [5]	Diff routes, DNS and egress rules against every range in the table	No
Edition-gated application options	Upstream source or edition docs [53, 54]	Confirm each lockdown option is honored by the edition actually installed; test it	No
Storage soft delete vs destruction evidence	Vendor's soft-delete page [56]	Set retention to zero or wait it out; check for soft-deleted and unlabeled objects before signing evidence	No
Identity-provider user-ID ranges vs inactivity detection	Identity provider's UID documentation [72]	Make the idle check cover every UID range the provider can assign; test with a federated account	No
OS default log layout vs alert log sources	Distribution's logging configuration [74, 75]	For each alert, confirm the OS writes the event to a log the agent ships; fire each alert once	No
Boot-time check ordering vs login service start	The image's own service ordering	Make gate status boot-scoped; reboot with a failing check and try to log in	Yes in principle (visible in the code; both panels missed it)
Live package mirrors vs admission policy	Advisory databases [63]	Decide a quarantine window or allowlist per ecosystem; record who approves exceptions	No
Pre-GA products in the PHI path	Vendor's BAA guidance [38]	List each product's launch stage; record the security office's position on any pre-GA item	No
Per-study keys vs shared service agents	Vendor's CMEK pages [49, 50]	Check which principal holds key roles; if one agent holds every key, use projects to separate	No

6.1 An inherited-default checklist

Table 9 turns the inherited-default findings into checks another campus could run on its own enclave code, with no access to our design. The last column says whether reading the repository alone could have caught each one. For nine of the ten the answer is no: nothing in the repository implies the defect, and deciding it needs a fact stated only in an outside document. The tenth, boot-time check ordering, was visible in the code, and both panels missed it.

6.2 The register

The companion report's register has 35 items, each a defect or open question in the plan or its Terraform. Likelihood and impact are my judgments on a three-point scale and exist to order the work. Of the 35, 14 close with a decision, a signature or a document edit, 18 need code or configuration, and three need pilot tests. Eight are rated high on both counts: three decisions, four configuration changes and one test (Table 10). The risks most likely to stall the first study are owners and decisions, not Terraform.

Table 10: Selected risks from the 35-item register: the eight rated high-high, then four others. L = likelihood, I = impact (H, M, L); type D = decision or document, C = code or configuration, T = pilot test (companion report, Chapter 14).

ID	Risk	L	I	Mitigation	Type
R02	Accountable data owner unnamed; systems team has not accepted seven rows; RACI rows missing	H	H	Name the data owner; add rows; record systems-team acceptance	D
R05	Perimeter and identity-domain restriction absent from the offline plans	H	H	Obtain identifiers; plan and dry-run in the pilot	C
R07	Perimeter ingress rules do not yet admit storage access by members and stewards	H	H	Test upload and steward download in the dry-run week (test 4)	T
R12	Administrators can reach every study and the evidence store through the deployment identity	H	H	Break-glass use of that identity, with an alert; evidence in a locked store it cannot write	C
R17	Copy-out restrictions depend on options only a paid software edition honors	H	H	Decide the edition; no such tool for no-copy-out studies until then	D
R21	Teardown verification can miss soft-deleted or unlabeled copies	H	H	Soft-delete policy; unlabeled-resource check; fail on errors	C
R22	Live package mirrors make no admission decision; runtime versions not pinned	H	H	Quarantine window; allowlist; minimum-version build check	C
R26	FERPA absent from intake and the breach matrix	H	H	Five intake questions and a matrix row	D
R01	No security-office risk treatment plan; security officer has not accepted D3, D4 or D12	M	H	One treatment plan with decision record and reversal conditions	D
R03	D8 conflicts with the pay-your-own-costs rule; no owner of shared costs	H	M	Decide and record; name the funder of staff, pentest and response	D
R06	Service-perimeter lists do not name the exact APIs used	H	M	Restrict the API that carries the traffic; matching ingress rules	C
R31	Boot-time check ordering can leave the fail-closed gate open	M	H	Boot-scoped status; force a failing step (test 6)	C

6.3 Decision gates

The companion report proposes thirty candidate changes for a version 4, each marked for before the fake-data pilot, before first P3–P4 data, or later. Before the pilot: add FERPA to scope and intake, name the accountable data owner, get the systems team's acceptance, make the gate status boot-scoped, fix idle stop's user-ID coverage, correct the mirror paths and re-aim the mis-aimed alert, align the perimeter's API, route and access-level lists with Google's tables and add the sink rules, set soft delete and extend the teardown check, add client keepalives, pin R, check the build script's downloads, and add a cross-study negative test and researcher-time measures. Before first P3–P4 data: the security office's risk treatment plan and the security officer's acceptance (including D4), a decision on who may act as the deployment identity, an enforced perimeter, the locked log bucket, a tabletop and the penetration test.

Figure 4 spreads the plan's seventh step across six gates in dependency order. The risk treatment plan needs pilot results, because three of its claims (shared boundary, gate, copy-out mode) rest on tests. Enforcement follows a clean dry-run week. The log lock is irreversible for its retention period, so it follows the teardown test and the retention decision. The penetration test follows enforcement so that it tests the system as it would run. I haven't proposed dates; they depend on the capacity of the campus security office and the systems team. The fifteen tests plus four additions (cross-study negatives, storage ingress, researcher time, RStudio edition) would confirm or refute the configuration and test risks. They can't close the fourteen decision risks.

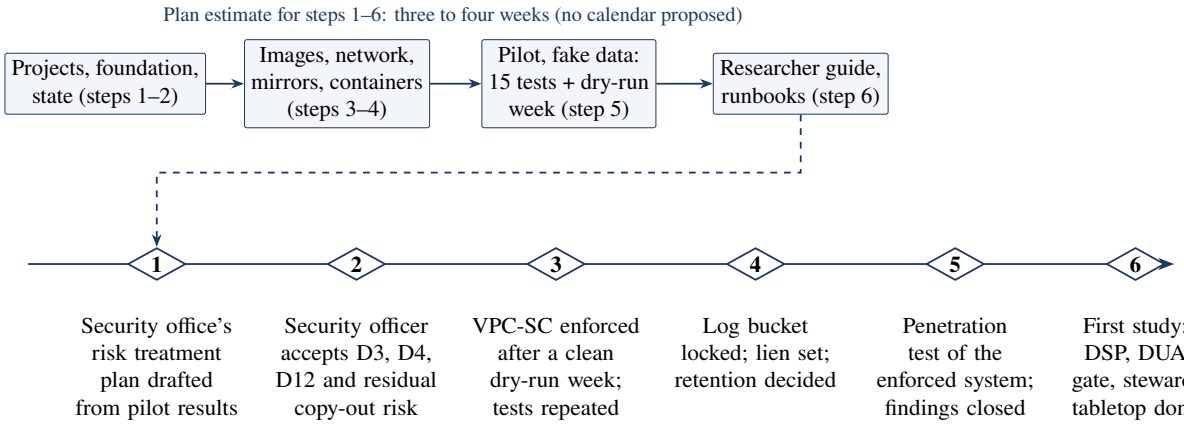


Figure 4: Proposed roadmap with six decision gates (from the companion report's Figure 14.1). Phases follow the plan's build steps; the gates split its final step. Gate order is a proposal; nothing shown has been scheduled.

6.4 Decisions for leadership

Six decisions belong to the research computing director, a campus research advisory board or campus leadership, and testing can't settle them.

1. **Funding.** D8's "possibly subsidized" against the pay-your-own-costs rule, and who pays for shared staff, the penetration test and incident response.
2. **Tenancy.** Keep the shared project (D3) with written triggers and a study count at which it is reviewed; adopt Option C before the foundation is applied, while the change is cheap; or keep D3 for the pilot and decide on C at the review point.
3. **Software admission.** Live mirrors, mirrors with a quarantine window, or a curated allowlist. More admission control means more recurring requests for the research computing team.
4. **RStudio edition** for `gui_only` studies: license Workbench (I haven't priced it), treat RStudio as shell-mode only, or offer only the desktop to studies that forbid copy-out.
5. **Reuse from the CUI enclave.** Whether the public Google foundation the CUI enclave was built on could supply our foundation's bootstrap, state and sinks; a question for the team that runs UCR's separate CUI enclave, which stays a separate service with its own scope either way.
6. **Globus.** Whether to evaluate it as an ingress path or keep bucket upload as the only route.

If these wait until after the pilot, gate 2 becomes the longest wait, because it needs the accountable data owner, the funding answer and the RStudio edition together.

7 Conclusion

If I could get one thing decided this month, it would be who the accountable data owner is. Five rows of the responsibility matrix and the second gate wait on that name, and no amount of Terraform moves them.

Before we build a single image, I'd fix three things in the code, or the pilot will test the wrong thing: make the login gate's status boot-scoped so a stale pass can't let anyone in, make idle stop see the user IDs our

identity provider actually assigns, and point the one mis-aimed alert at the log the OS really writes. None is large. Two of the three, the idle check's user-ID range and the alert's log source, showed up only beside the identity provider's and the distribution's pages; the third, the gate's boot ordering, was visible in the code and both panels missed it. That is the main thing I'd tell another campus: give your reviewers, human or AI, the vendor documentation along with the code, and run the checks in Table 9 before you trust a setting.

What I'm least sure of is staffing. The ratios in circulation are borrowed and swing by a factor of fifteen with two assumptions, and a quarantine window or a steward checklist costs people time that no one has measured. The way to answer that is to log hours during the pilot and set the rate and the service levels from those numbers. I'd keep the register as the plan's open-items list and revisit it after the pilot, with my judgments replaced by test results and measured times.

References

[1] University of California Office of the President. UC Protection Level Classification Guide. https://security.ucop.edu/files/documents/uc-protection-level-classification-guide.pdf, 2026. Accessed 2026-11-01.

[2] University of California Office of the President. Health Insurance Portability and Accountability Act of 1996 (HIPAA), Ethics, Compliance and Audit Services. ucop.edu, 2026. URL https://www.ucop.edu/ethics-compliance-audit-services/compliance/hipaa/. Accessed 2026-11-01.

[3] National Institutes of Health. NOT-OD-24-157: Implementation update for data management and access practices under the genomic data sharing policy. https://grants.nih.gov/grants/guide/notice-files/NOT-OD-24-157.html, 2024. Accessed 2026-11-01.

[4] Google Cloud. How UC Riverside is securing the path to federal grants with Google Public Sector. https://cloud.google.com/blog/topics/public-sector/a-new-standard-for-research-how-uc-riverside-is-securing-the-path-to-federal-grants-with-google-public-sector, 2026. Accessed 2026-11-01.

[5] Google Cloud. Configure Private Google Access. https://docs.cloud.google.com/vpc/docs/configure-private-google-access, 2026. Accessed 2026-11-01.

[6] Google Cloud. Use IAP for TCP forwarding. https://docs.cloud.google.com/iap/docs/using-tcp-forwarding, 2026. Accessed 2026-11-01.

[7] Harvard FAS Research Computing. FASSE cluster (FAS secure environment). https://docs.rc.fas.harvard.edu/kb/fasse/, 2026. Accessed 2026-11-01.

[8] Stanford Research Computing. Nero GCP Cost. https://docs.nero.stanford.edu/gcp-cost.html, 2026. Accessed 2026-11-01.

[9] Office of Information and Regulatory Affairs. Unified Agenda entry, RIN 0945-AA22: HIPAA Security Rule to Strengthen the Cybersecurity of Electronic Protected Health Information. reginfo.gov, 2026. URL https://www.reginfo.gov/public/do/eAgendaViewRule?pubId=202510&RIN=0945-AA22. Stage: Long-Term Actions; final action 07/00/2027. Accessed 2026-11-01.

[10] Centers for Medicare and Medicaid Services. Research Identifiable File Data Use Agreement Policy Guide. https://www.cms.gov/files/document/research-identifiable-file-data-use-agreement-policies.pdf, 2026. Accessed 2026-11-01.

[11] Research Data Assistance Center (ResDAC). Research Identifiable File Data Use Agreement. https://resdac.org/sites/default/files/2026-05/RIF-Data-Use-Agreement.pdf, 2026. Accessed 2026-11-01.

[12] Jasmine Kendall, Livia Martinescu, Sulamaan Rahim, and Gene Tunny. Interim evaluation of Administrative Data Research UK. https://www.ukri.org/wp-content/uploads/2024/11/ESRC-121124-InterimEvaluationAdministrativeDataResearchUK.pdf, 2024. Oxford Insights and Lateral Economics for ESRC; accessed 2026-11-01.

[13] U.S. Department of Health and Human Services. 45 CFR 164.316, policies and procedures and documentation requirements. Electronic Code of Federal Regulations, 2026. URL https://www.ecfr.gov/current/title-45/section-164.316.

Text current through 2026-10-29. Accessed 2026-11-01.

[14] Center for Internet Security. Hardened Images Pricing Questions. https://cisecurity.atlassian.net/wiki/spaces/CSKB/pages/1139541045/Hardened+Images+Pricing+Questions, 2024. Knowledge-base page dated 2024-01-25; accessed 2026-11-01.

[15] Office of the Federal Register. 2 CFR 200.468, Specialized service facilities. https://www.ecfr.gov/current/title-2/section-200.468, 2026. Electronic Code of Federal Regulations, current text accessed 2026-11-01.

[16] Stanford Research Computing. Nero GCP: Powerful, Secure Computing for High-Risk Data. https://docs.nero.stanford.edu/, 2026. Accessed 2026-11-01.

[17] Stanford Research Computing. Who Can Use Nero? Nero GCP User Documentation. https://docs.nero.stanford.edu/nero-who.html, 2026. Accessed 2026-11-01.

[18] Stanford Research Computing. Overview of Nero Google Cloud Platform (GCP). https://docs.nero.stanford.edu/gcp-overview.html, 2026. Accessed 2026-11-01.

[19] San Diego Supercomputer Center Sherlock Cloud Solutions. Sherlock Cloud Solutions: Solutions. https://sherlock.sdsc.edu/solutions, 2026. Accessed 2026-11-01.

[20] University of Utah Center for High Performance Computing. Protected Environment Frequently Asked Questions. https://www.chpc.utah.edu/documentation/pefaq.php, 2026. Accessed 2026-11-01.

[21] NYU Research Technology Services. Secure Research Data Environment (SRDE): Eligibility and Accounts. https://services.rt.nyu.edu/docs/srde/getting_started/eligibility_accounts/, 2026. Accessed 2026-11-01.

[22] UCLA Digital and Technology Solutions. High Compliance Environment (HCE). https://dts.ucla.edu/products-services/security/high-compliance-environment, 2026. Accessed 2026-11-01.

[23] Yale Center for Research Computing. Hopper. https://docs.ycrc.yale.edu/clusters/hopper/, 2026. Accessed 2026-11-01.

[24] Columbia University Shared Research Computing Policy Advisory Committee. Minutes of the Spring 2026 Meeting. https://research.columbia.edu/sites/research.columbia.edu/files/content/ORI/SRCPAC/Minutes_SRCPAC%20Meeting_Spring%202026.pdf, 2026. Accessed 2026-11-01.

[25] UCSF IT. Research Analysis Environment (RAE). https://it.ucsf.edu/service/rae, 2026. Accessed 2026-11-01.

[26] Johns Hopkins University Sheridan Libraries. Secure Storage Choices, Protecting Human Subject Identifiers. https://guides.library.jhu.edu/protecting_identifiers/StorageChoices, 2026. Accessed 2026-11-01.

[27] Stanford Research Computing. Carina vs. Nero GCP. https://docs.carina.stanford.edu/prem-vs-cloud, 2026. Accessed 2026-11-01.

[28] Stanford Research Computing. Carina Computing Platform. https://docs.carina.stanford.edu/, 2026. Accessed 2026-11-01.

[29] Stanford Research Computing. Carina Cost. https://docs.carina.stanford.edu/cost, 2026. Accessed 2026-11-01.

[30] University of Michigan ITS. Secure Enclave Services (SES). https://its.umich.edu/advanced-research-computing/cloud/secure-enclave-services, 2026. Accessed 2026-11-01; the page blocks scripted curl requests and was read through a page-extraction tool.

[31] U.S. Department of Education. Family Educational Rights and Privacy, 34 CFR Part 99, Sections 99.3, 99.30-99.33 and 99.35. Electronic Code of Federal Regulations, 2026. URL https://www.ecfr.gov/current/title-34/subtitle-A/part-99. Text current through 2026-10-29. Accessed 2026-11-01.

[32] U.S. Department of Education, Privacy Technical Assistance Center. Data Breach Response Checklist. studentprivacy.ed.gov, 2012. URL https://studentprivacy.ed.gov/sites/default/files/resource_document/file/checklist_data_breach_response_092012_0.pdf. Accessed 2026-11-01.

[33] U.S. Department of Education, Student Privacy Policy Office. Data Security: K-12 and Higher Education. studentprivacy.ed.gov, 2026. URL https://studentprivacy.ed.gov/data-security-k-12-and-higher-education. Accessed 2026-11-01.

[34] U.S. Department of Health and Human Services. Security and Privacy, 45 CFR Part 164, Sections 164.308, 164.310, 164.312, 164.314, 164.316, 164.404 and 164.410. Electronic Code of Federal Regulations, 2026. URL

https://www.ecfr.gov/current/title-45/part-164. Text current through 2026-10-29. Accessed 2026-11-01.

[35] Department of Health and Human Services. HIPAA security rule to strengthen the cybersecurity of electronic protected health information (proposed rule, 90 FR 898). https://www.federalregister.gov/documents/2025/01/06/2024-30983/hipaa-security-rule-to-strengthen-the-cybersecurity-of-electronic-protected-health-information, 2026. Accessed 2026-11-01.

[36] U.S. Department of Health and Human Services. HIPAA security rule notice of proposed rulemaking to strengthen cybersecurity for electronic protected health information: Fact sheet. HHS.gov, 2024. URL https://www.hhs.gov/hipaa/for-professionals/security/hipaa-security-rule-nprm/factsheet/index.html. Dated 2024-12-27. Retrieved by text extraction; direct command-line requests returned HTTP 403. Accessed 2026-11-01.

[37] University of California Office of the President. Systemwide IT Policies (includes BFB-IS-3, Electronic Information Security). https://security.ucop.edu/policies/it-policies.html, 2026. Accessed 2026-11-01.

[38] Google Cloud. HIPAA compliance on google cloud. https://docs.cloud.google.com/security/compliance/hipaa, 2026. Supported-products list dated 2025-09-17; Accessed 2026-11-01.

[39] Alan Turing Institute. The Turing Data Safe Haven documentation, version 5.2.1. https://data-safe-haven.readthedocs.io/en/v5.2.1/index.html, 2026. Accessed 2026-11-01.

[40] Alan Turing Institute. Technical controls, Turing Data Safe Haven documentation, version 5.2.1. https://data-safe-haven.readthedocs.io/en/v5.2.1/design/security/technical_controls.html, 2026. Accessed 2026-11-02.

[41] Christian Cole, Hari Sood, Simon Li, Katie Oldfield, Matt Craddock, and Nel Swanepoel. SATRE: Standard architecture for trusted research environments. Zenodo, version 2.0.1, 2026. Accessed 2026-11-01.

[42] SATRE project. SATRE specification, standard architecture for trusted research environments. https://satre-specification.readthedocs.io/en/stable/specification.html, 2026. Items 1.4.07, 2.1.12 and 2.5.03 read. Accessed 2026-11-02.

[43] James Bailie and Ruobin Gong. The Five Safes as a Privacy Context, 2025. Version 2, 2026-09-01. Accessed 2026-11-01.

[44] Yang Guo, Ramaswamy Chandramouli, Lowell Wofford, Rickey Gregg, Gary Key, Antwan Clark, Catherine Hinton, Andrew Prout, Albert Reuther, Ryan Adamson, Aron Warren, Purushotham Bangalore, Erik Deumens, and Csilla Farkas. High-Performance Computing Security: Architecture, Threat Analysis, and Security Posture. NIST Special Publication SP 800-223, National Institute of Standards and Technology, February 2024. URL https://doi.org/10.6028/NIST.SP.800-223. Accessed 2026-11-02.

[45] Google Cloud. Configure a VM to access a Yum repository, Artifact Registry. https://docs.cloud.google.com/artifact-registry/docs/os-packages/rpm/configure, 2026. Remote Yum use marked Preview; Accessed 2026-11-01.

[46] Google Cloud. Supported formats, Artifact Registry. https://docs.cloud.google.com/artifact-registry/docs/supported-formats, 2026. Accessed 2026-11-01.

[47] Google Cloud. Supported products and limitations, VPC Service Controls. https://docs.cloud.google.com/vpc-service-controls/docs/supported-products, 2026. Accessed 2026-11-01.

[48] Google Cloud. Service account permissions: impersonate and attach. https://docs.cloud.google.com/iam/docs/service-account-permissions, 2026. Accessed 2026-11-01.

[49] Google Cloud. Protect resources with Cloud KMS keys (Compute Engine). https://docs.cloud.google.com/compute/docs/disks/customer-managed-encryption, 2026. Accessed 2026-11-01.

[50] Google Cloud. Customer-managed encryption keys (CMEK). https://docs.cloud.google.com/kms/docs/cmek, 2026. Accessed 2026-11-01.

[51] Google Cloud. Folder membership in service perimeters. https://docs.cloud.google.com/vpc-service-controls/docs/folder-membership, 2026. Page last updated 2026-10-28. Accessed 2026-11-02.

[52] Globus. Managing Protected Data. https://www.globus.org/protected-data, 2026. Accessed 2026-11-01.

[53] Posit. RStudio source, SessionOptions.cpp. https://raw.githubusercontent.com/rstudio/rstudio/main/src/cpp/session/SessionOptions.cpp, 2026. Main branch. Accessed 2026-11-01.

[54] Posit. rsession.conf, Workbench documentation, release 2026.10.1. https://docs.posit.co/ide/server-pro/admin/reference/rsession_conf.html, 2026. Accessed 2026-11-01.

[55] Neutrino Labs. xrdp sesman.ini configuration file and manual page, development branch. https://raw.githubusercontent.com/neutrinolabs/xrdp/devel/sesman/sesman.ini.in, 2026. Accessed 2026-11-01.

[56] Google Cloud. Soft delete. https://docs.cloud.google.com/storage/docs/soft-delete, 2026. Accessed 2026-11-01.

[57] U.S. Department of Education, Privacy Technical Assistance Center. Best Practices for Data Destruction. https://studentprivacy.ed.gov/sites/default/files/resource_document/file/Best%20Practices%20for%20Data%20Destruction%20(2019-3-26).pdf, 2019. Page dated March 2019. Accessed 2026-11-01.

[58] Office for National Statistics. SRS output checking guidance: Statistical disclosure control. https://www.ons.gov.uk/file?uri=/aboutus/whatwedo/statistics/requestingstatistics/secureresearchservice/gettingyourresearchoutputsapproved/srsoutputcheckingguidance.february2025.pdf, 2025. February 2025 edition. Accessed 2026-11-01.

[59] Centers for Medicare and Medicaid Services. CMS cell suppression policy. https://www.hhs.gov/guidance/document/cms-cell-suppression-policy, 2020. Issue date 2020-01-01; version date 2017-05-08. Read through a text extractor; a direct fetch returned HTTP 403. Accessed 2026-11-01.

[60] Esma Mansouri-Benssassi, Simon Rogers, Jim Smith, Felix Ritchie, and Emily Jefferson. Machine Learning Models Disclosure from Trusted Research Environments (TRE), Challenges and Opportunities, 2021. Read at abstract level. Accessed 2026-11-01.

[61] Emily Jefferson, James Liley, Maeve Malone, Smarti Reel, Alba Crespi-Boixader, and Xaroula Kerasidou. GRAIMATTER green paper: Recommendations for disclosure control of trained machine learning (ML) models from trusted research environments (TREs), 2022. Read at abstract level. Accessed 2026-11-01.

[62] Open Source Vulnerabilities. PYSEC-2026-508: Compromise of PyTorch Lightning PyPI package versions. https://osv.dev/vulnerability/PYSEC-2026-508, 2026. Record read through api.osv.dev; Accessed 2026-11-01.

[63] Open Source Vulnerabilities. MAL-2026-3201: Malicious code in lightning (PyPI). https://osv.dev/vulnerability/MAL-2026-3201, 2026. Accessed 2026-11-01.

[64] Ilyas Makari. Popular PyTorch Lightning package compromised by Mini Shai-Hulud. https://www.aikido.dev/blog/pytorch-lightning-pypi-compromise-mini-shai-hulud, 2026. Vendor blog post dated 2026-04-30; Accessed 2026-11-01.

[65] Python Package Index. pytorch-lightning release metadata (JSON API). https://pypi.org/pypi/pytorch-lightning/json, 2026. Upload times read for versions 2.6.1 to 2.6.4; Accessed 2026-11-01.

[66] Open Source Vulnerabilities. CVE-2024-27322: deserialization of untrusted data in R. https://osv.dev/vulnerability/CVE-2024-27322, 2026. Accessed 2026-11-01.

[67] Fedora Project. mdapi record for package R in EPEL 9. https://mdapi.fedoraproject.org/epel9/pkg/R, 2026. Reported version 4.6.1, release repository; Accessed 2026-11-01.

[68] Takaaki Toda and Tatsuya Mori. CHASE: LLM agents for dissecting malicious PyPI packages, 2026. Abstract read; Accessed 2026-11-01.

[69] Siyuan Pang, Yepeng Yao, Zhengwei Jiang, Zijing Fan, Haozhe Li, and Baoxu Liu. PYPILINE: Malicious PyPI package detection via suspicious API knowledge and agent workflow, 2026. Abstract read; Accessed 2026-11-01.

[70] Himadri Chowdhury, Hatem Ahmed, and Joseph Khoury. MTPkgScan: A metamorphic testing framework for package malware scanners. Zenodo record, 2026. Record abstract read; Accessed 2026-11-01.

[71] OpenBSD Project. ssh_config(5): ServerAliveInterval and ServerAliveCountMax. https://man.openbsd.org/ssh_config, 2026. Accessed 2026-11-01.

[72] Google Cloud. Manage OS Login in an organization. https://cloud.google.com/compute/docs/oslogin/manage-oslogin-in-an-org, 2026. States that custom POSIX UIDs must be between 1001 and 60000, or between 65535 and 2147483647; default UIDs are randomly generated. Accessed 2026-11-02.

[73] Gillian M. Raab, Sophie McCall, and Liam Cavin. Four checks for low-fidelity synthetic data: recommendations for disclosure control and quality evaluation. arXiv, 2025. Abstract-level reading.

[74] Rocky Enterprise Software Foundation. rsyslog.conf, rsyslog package source for Rocky Linux 9 (branch r9). https://git.rockylinux.org/staging/rpms/rsyslog/-/raw/r9/SOURCES/rsyslog.conf, 2026. Accessed

2026-11-02.

[75] Google Cloud. Ops Agent configuration. https://docs.cloud.google.com/stackdriver/docs/solutions/agents/ops-agent/configuration, 2026. Accessed 2026-11-01.

[76] U.S. Department of Health and Human Services. 45 CFR 164.402, definitions (breach, unsecured protected health information). Electronic Code of Federal Regulations, 2026. URL https://www.ecfr.gov/current/title-45/section-164.402. Text current through 2026-10-29. Accessed 2026-11-01.

[77] U.S. Department of Health and Human Services. 45 CFR 164.414, administrative requirements and burden of proof. Electronic Code of Federal Regulations, 2026. URL https://www.ecfr.gov/current/title-45/section-164.414. Text current through 2026-10-29. Accessed 2026-11-01.

[78] University of California Office of the President. UC information security incident response standard. https://security.ucop.edu/files/documents/policies/incident-response-standard.pdf, 2019. Last updated 8/21/2019 per the document; sections 2.7, 2.9. Accessed 2026-11-01.

[79] Google Cloud. Compute Engine pricing: general-purpose machine types (N2), Iowa (us-central1). https://cloud.google.com/products/compute/pricing/general-purpose, 2026. Accessed 2026-11-01.

[80] Google Cloud. Compute Engine pricing: accelerator-optimized machine types (G2), Iowa (us-central1). https://cloud.google.com/products/compute/pricing/accelerator-optimized, 2026. Accessed 2026-11-01.

[81] Google Cloud. Disk and image pricing. https://cloud.google.com/compute/disks-image-pricing, 2026. Accessed 2026-11-01.

[82] Google Cloud. Stop and start a compute instance. https://cloud.google.com/compute/docs/instances/stop-start-instance, 2026. Accessed 2026-11-01.

[83] Google Cloud. Create, edit, or delete budgets and budget alerts. https://cloud.google.com/billing/docs/how-to/budgets, 2026. Accessed 2026-11-01.

[84] Google Cloud. Accelerator-optimized machine family and GPU machine types (G2, A2, A3, G4). https://cloud.google.com/compute/docs/gpus, 2026. Accessed 2026-11-01.

[85] Qwen Team. Qwen3-8B model card. https://huggingface.co/Qwen/Qwen3-8B, 2026. Accessed 2026-11-01; parameter counts from the repository's safetensors metadata.

[86] Qwen Team. Qwen3-32B model card. https://huggingface.co/Qwen/Qwen3-32B, 2026. Accessed 2026-11-01.

[87] Google Cloud. Create a Confidential VM instance with GPU. https://cloud.google.com/confidential-computing/confidential-vm/docs/create-a-confidential-vm-instance-with-gpu, 2026. Accessed 2026-11-01.

[88] Google Cloud. Confidential VM supported configurations. https://cloud.google.com/confidential-computing/confidential-vm/docs/supported-configurations, 2026. Accessed 2026-11-01.