

An Author's Taxonomy of Customer-Observable Security Controls in SaaS Platforms

 **Sergei Beliachkov**¹

¹ Head of Department, Platform Cybersecurity Center, JSC Sberbank-Technologies, Moscow, Russia

Abstract: The software-as-a-service model places most technical security mechanisms in the provider's hands, while the customer remains accountable for data, identities, and regulatory outcomes. Certification reports describe what a provider has implemented, yet reveal little about what a tenant can monitor and verify on its own. This article introduces a taxonomy that classifies security controls for SaaS platforms by the degree to which they are observable to the tenant. The proposed scale contains four levels: directly observable controls, controls verifiable on request, attestation-only controls, and non-observable controls. The classification procedure draws on the SaaS Security Capability Framework of the Cloud Security Alliance, the Cloud Controls Matrix, NIST SP 800-53, and ISO/IEC 27001, and the scale is applied to three control domains: logging and monitoring, security incident management, together with e-discovery and cloud forensics, and identity and access management. The selection of domains mirrors the framework sections shaped through first-hand work within the Cloud Security Alliance working group. Observations from a corporate program that hardened seven SaaS platforms serving more than 100,000 active users illustrate how the scale behaves in practice. The taxonomy gives tenants a shared language for procurement, contractual negotiations, and continuous assurance, and gives providers a roadmap for increasing visibility into their controls. A systematic classification of SaaS controls along the observability axis has not been proposed before.

Keywords: SaaS security, customer-observable controls, security transparency, cloud assurance, shared responsibility model, SaaS Security Capability Framework.

1. Introduction

Software as a service has become the dominant delivery model for business applications, and security remains the main obstacle to its adoption. A multivocal review of 93 studies identified data leakage, identity and access management, regulatory compliance, and malicious insiders as the SaaS challenges most frequently reported in academic and practitioner literature alike (Humayun et al., 2022). Each of these challenges shares a structural root. The tenant entrusts code, infrastructure, and operational processes to an external party and loses the technical means to inspect them. What the tenant retains is accountability, since regulators, customers, and courts hold the organization whose data was leaked accountable, regardless of where the failure occurred.

The shared responsibility model splits security duties between the provider and the customer, and the split depends on the service model. In SaaS, the provider

operates the application, platform, and infrastructure, leaving the tenant responsible for configuration, identity governance, and data handling. Research on cloud infrastructure security shows that transparency between the user and the service provider plays a decisive role in this arrangement, as the customer needs awareness of storage locations, applied policies, and protection methods to manage residual risk (Alghofaili et al., 2021). Established control catalogs organize safeguards by function. Comparative analyses of the COBIT, NIST, ISO, and Cloud Security Alliance frameworks assess their focus, scope, and limitations, and none of the schemes ranked controls by the customer's ability to observe their operation (Chauhan & Shiaeles, 2023). NIST SP 800-53 and ISO/IEC 27001 follow the same functional logic and remain silent on who can see a control at work (International Organization for Standardization [ISO], 2022; Joint Task Force, 2020).

The economics of the relationship amplifies the problem. Certification, the dominant assurance instrument, operates on a timescale of years, whereas a SaaS platform changes on a timescale of days. A certificate, therefore, reports the past state of a moving system, and the tenant that relies on it buys confidence with a built-in delay. Evidence the tenant can gather itself carries no such delay, which is why the question of observability precedes the question of control quality in any practical assessment of a platform. A provider may run an excellent control whose excellence the customer has no way to confirm, and for the purposes of the customer risk register, such a control contributes no verifiable assurance.

In 2025, the Cloud Security Alliance published the SaaS Security Capability Framework, hereinafter SSCF, the first industry standard devoted to customer-facing security controls of SaaS platforms (Cloud Security Alliance [CSA], 2025). SSCF version 1.0 defines configurable and consumable controls across six domains whose names follow the Cloud Controls Matrix convention (CSA, 2021, 2025). The framework asks providers to implement these controls and expose them to customers, who are responsible for using them, so the document reshapes the shared responsibility model around concrete product capabilities. Membership in the Cloud Security Alliance working group that produced the framework included direct work on two of its sections, logging and monitoring, together with security incident management, e-discovery, and cloud forensics, and later on the identity and access management section of the SSCF-to-CCM v4.1 mapping. This first-hand participation shaped the scope of the present article.

The framework also changes the vocabulary of vendor assessment. Where earlier questionnaires asked a provider to describe its practices in free text, the SSCF names concrete capabilities, from log delivery to enforced single sign-on, that either exist in the product or do not. This shift from narrative to capability invites the next analytical step: ranking the capabilities by the strength of the position they give the customer. That step is the subject of the present article.

Prior research laid the groundwork for this move. Cloud security surveys registered a lack of user control and loss of visibility among the recurring complaints of tenants across a decade of studies (Alghofaili et al., 2021; Humayun et al., 2022), and framework comparisons recorded the response of the industry, a growing family of

catalogs whose coverage widens while the position of the customer inside them stays fixed (Chauhan & Shiaeles, 2023). The taxonomy proposed below reads this literature from the tenant's perspective and asks every control one question that the catalogs leave unasked: what can the customer see.

A gap remains between what the SSCF asks providers to disclose and how tenants' reason about assurance. Existing catalogs answer the question of what a control does. They leave open the question of how a customer learns that the control works. The difference matters in negotiations, audits, and incidents, when the value of a control collapses to the value of the evidence it leaves behind. No published classification arranges SaaS security controls along that axis. The article addresses the gap in three steps. First, it formulates an observability criterion and derives a four-level scale from it. Second, it applies the scale to the three SSCF domains named above and maps representative controls to established catalogs. Third, it examines the practical consequences of the scale through the lens of a multi-year corporate security program and derives recommendations for tenants and providers.

2. Materials and Methods

The study belongs to the review genre and rests on document analysis. The source corpus consists of the published SSCF version 1.0 (CSA, 2025), the Cloud Controls Matrix version 4 (CSA, 2021), NIST SP 800-53 revision 5 (Joint Task Force, 2020), ISO/IEC 27001:2022 (ISO, 2022), public documentation of widespread SaaS mechanisms such as audit log streaming, single sign-on, and cross-domain identity provisioning, and peer-reviewed literature on cloud and SaaS security published between 2021 and 2026. Internal working materials of the Cloud Security Alliance and corporate documents of the organizations mentioned below were excluded from the corpus, so every claim about control content can be traced to an open source.

The classification procedure unfolded in five steps. The first step extracted customer-facing controls from the three examined SSCF domains and their matching families in the reference catalogs. The second step formulated the observability criterion: a control counts as observable to the extent that the tenant can obtain evidence of its operation through standard product interfaces without provider assistance. The third step derived discrete levels from the criterion by grouping

three attributes: the verifier's identity, the cadence of verification, and the type of supporting evidence. The fourth step assigned each extracted control to a level and checked the assignment against the product documentation for widely used SaaS platforms. The fifth step resolved boundary cases with a conservative rule: when a control spans two levels, it receives the lower of the two confirmable levels, because assurance claims should not exceed available evidence.

The three attributes behind the levels deserve a short justification. The verifier's identity distinguishes self-service evidence from mediated evidence, and mediation always introduces the mediator's interests. The cadence of verification separates continuous confidence from periodic confidence, and the gap between audits is exactly the window in which silent control degradation accumulates. The type of evidence separates machine-readable records, which feed automation, from documents, which feed archives. Together, the attributes describe how far a tenant can push its own assurance process before it must rely on others' words.

Three measures guarded the procedure's validity. Extraction and assignment relied solely on published texts, making every step reproducible by a reader using the same documents. Assignments were rechecked against documentation for several unrelated SaaS products, reducing the risk of anchoring to one vendor's habits. The conservative boundary rule removes the main source of inflation, the temptation to credit a control with the strongest level it touches under ideal conditions.

The resulting scale distinguishes four levels. Directly observable controls leave evidence in the product itself, in dashboards, journals, or APIs that the tenant reads at will. Controls verifiable on request produce artifacts that the provider hands over when asked, such as an incident report or an evidence package. Attestation-only controls remain hidden from the tenant, and an external auditor

vouches for them through certificates and periodic reports. Non-observable controls operate inside the provider without any evidence channel to the customer.

The scale measures evidence and stays silent about a control's internal quality. A weak control can be observable and a strong one hidden, and the two judgments should be kept apart. A tenant needs both of them, since the observability level tells it whether independent judgment about quality is possible at all. This separation also protects the scale from vendor marketing, because a claim about quality requires proof at the corresponding level before it enters a risk decision.

Observations that illustrate the scale come from a security program executed between 2022 and 2025 across corporate SaaS platforms of SberTech, including `gitverse.ru`, `saas.works.sbertech.ru`, `web.senat.cloud`, `demo.navplatform.ru`, `pmdemo.sbertech.ru`, `datatracker.sbertech.ru`, and `sberdatascience.ru`, which jointly serve more than 100,000 active users. These observations enter the article as first-hand practitioner experience, without the status of a controlled experiment, and the text marks their epistemic weight wherever they appear.

3. Results

3.1. The four-level observability scale

The scale ranks the controls of the examined domains, customer-facing capabilities, and provider-side safeguards alike, by the strength of the evidence a tenant can gather about them. Three attributes define each level: who performs the verification, how frequently the verification can be, and what type of evidence supports it. Figure 1 presents the scale structure, including the defining attributes and an example control for each level.

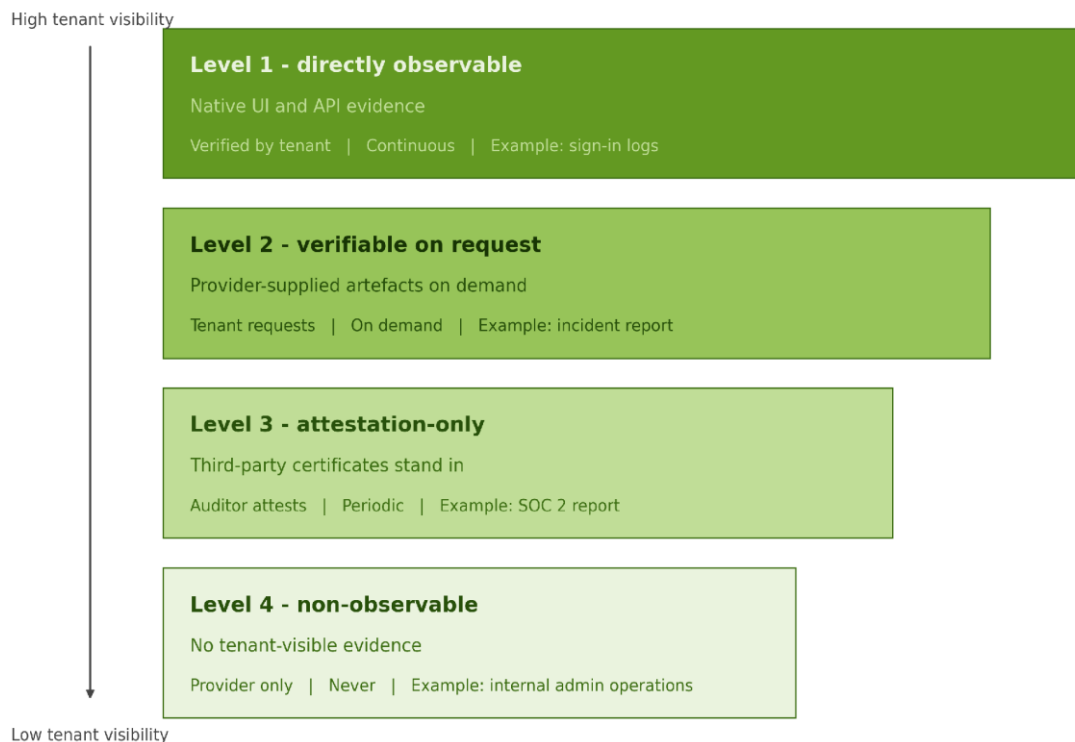


Fig. 1. The four-level scale of customer observability of security controls in SaaS platforms

The first level covers directly observable controls. The tenant reads their evidence in native interfaces, from sign-in journals to log streaming endpoints, and the reading can proceed continuously and without provider involvement. The second level covers controls verifiable on request. Their evidence exists on the provider side, and the tenant obtains it on demand through support channels, contractual clauses, or trust portals. The third level covers attestation-only controls. The tenant sees no evidence of the control itself, and confidence rests on periodic third-party examinations such as SOC 2 reports or ISO/IEC 27001 certificates (ISO, 2022). The fourth level covers non-observable controls that produce no tenant-visible traces, even through attestation scoping, such as day-to-day administrative operations within the provider.

The scale differs from maturity models in its subject. A maturity model grades an organization's internal processes, while the observability scale grades the boundary between two organizations. The distinction shows in the direction of improvement. A provider raises

its maturity by refining what it does, and it raises observability by exposing what it does. The second movement can lag the first by years, even in well-run companies, because exposure demands product interfaces, documentation, and support commitments that internal refinement never requires.

Two properties make the scale usable in analysis. The scale is ordinal, since each step downward weakens the tenant's position as a verifier, lengthens the verification cycle, and replaces machine-readable evidence with documents or silence. The scale is also exhaustive, since every control falls into exactly one level once the conservative boundary rule from the second section has been applied. Table 1 shows the assignment of six representative controls drawn from the three examined domains, including one provider-side safeguard on which the tenant depends without seeing it, together with the evidence a tenant can expect and the corresponding families of the reference catalogs.

Table 1. Classification of representative SaaS security controls by observability level

Control	Domain	Level	Evidence available to the tenant	Framework references
Audit log export to tenant systems	Logging and monitoring	1	Streaming API and documented event schema	LOG / AU / A.8.15
Tenant-configurable log retention	Logging and monitoring	1	Retention settings in the administrative console	LOG / AU / A.8.15
Customer incident notification	Incident management and forensics	2	Notification with scope and timeline	SEF / IR / A.5.24–A.5.26
Forensic evidence and e-discovery support	Incident management and forensics	2	Evidence package delivered on request	SEF / IR / A.5.28
Single sign-on and multi-factor policy enforcement	Identity and access management	1	Tenant-set policies and sign-in journals	IAM / IA, AC / A.5.16–A.5.17
Provider-side privileged access management	Identity and access management	3	Attestation reports only	IAM / AC / A.8.2

The framework references in Table 1 remain at the family level. The LOG, SEF, and IAM identifiers follow the Cloud Controls Matrix naming convention (CSA, 2021), the AU, IR, IA, and AC families belong to NIST SP 800-53 (Joint Task Force, 2020), and the A.5 and A.8 clauses belong to ISO/IEC 27001:2022 (ISO, 2022). Family-level anchoring keeps the taxonomy stable across future catalog revisions, since individual control numbers change more often than family names.

Reading the table by columns clarifies the taxonomy's intent. The level column compresses the assessment into a single ordinal value, the evidence column explains that value, and the reference column ties the customer-facing capability to the catalog language that governance teams speak. In vendor assessment practice, these three columns answer three different questions: how strong the tenant's position is, what the tenant should ask to see, and where the requirement lives within the organization's compliance universe.

3.2. Logging and monitoring

Logging and monitoring form the natural anchor of the taxonomy, because evidence itself is the product of this domain. The SSCF assigns seven controls to it and requires

providers to deliver audit-relevant events to the customer in a consumable form (CSA, 2025). Log export to the tenant's external system, configurable retention, and documented event schemas belong to the first level of the scale. Once a platform streams its audit trail outward, the tenant gains a continuous, machine-readable window into authentication activity, configuration changes, and administrative actions, and this window survives disputes, because exported records live outside the reach of the party they describe.

Retention and integrity deserve separate mention within the first level. A stream of events answers the question of what happens now, and retention settings answer the question of what can be reconstructed later. When the tenant controls retention and stores exported records within its own perimeter, the evidentiary value of the journal no longer depends on the platform's goodwill. Documented schemas complete the arrangement, since an undocumented event stream forces the tenant to reverse-engineer the meaning of its own evidence, and reverse engineering breaks down each time the provider ships a release.

For the tenant, first-level telemetry also changes the detection posture. Exported authentication and

administrative events allow the customer to run its own correlation rules against SaaS activity, join platform events with signals from the rest of its estate, and detect account compromise patterns that the provider, which sees one tenant at a time from the inside, lacks context to recognize. Detection thus becomes a shared capability, and the quality of the share available to the tenant depends on the observability level of the underlying controls.

Practice confirms the weight of this channel. An implementation study of a security information and event management deployment for cloud resources demonstrates automated visibility of the protected environment and, at the same time, warns that log collection and event investigation demand mature skills on the consumer side (Tuyishime et al., 2023). The warning matters for the taxonomy: a first-level control raises the ceiling on achievable assurance, and the tenant still has to build the capability to reach this ceiling. Recent forensic research moves in the same direction, constructing investigative knowledge bases by inspecting and tracking the RESTful APIs of cloud services, thereby turning exported telemetry into material for event reconstruction (Jeong et al., 2026).

The upper boundary of the domain lies where provider-internal monitoring begins. Detection rules, alert thresholds, and the coverage of the provider's security operations center remain invisible to the tenant, and their existence surfaces only through attestation reports at the third level, or not at all at the fourth. The logging and monitoring section of the SSCF took shape with direct participation in the Cloud Security Alliance working group, and the section reflects the position that the boundary between exported and internal telemetry should shift toward the tenant, on a control-by-control basis, as export interfaces mature.

3.3. Security incident management, e-discovery, and cloud forensics

The incident domain sits lower on the scale than any other examined area. A systematic review of cloud forensic investigation identifies loss of physical access, multi-tenant storage, jurisdictional fragmentation, and dependence on provider cooperation as persistent obstacles in the field (Alshabibi et al., 2024). Each obstacle translates into an evidence gap for the tenant. During an incident on a SaaS platform, the customer cannot image a disk, capture memory, or interview an administrator, and

every artifact of the investigation arrives through the provider.

The SSCF answers this asymmetry by concentrating on the interface between the parties (CSA, 2025). Customer notification about security incidents with defined timelines and scope belongs to the second level: the control operates on the provider side, and the tenant verifies it through the artifacts it produces. Provisioning forensic evidence packages and supporting e-discovery requests follow the same pattern: the tenant's request triggers delivery, and the delivered artifact carries the proof. Internal response runbooks, escalation chains, and containment tooling of the provider remain at the third and fourth levels, where the tenant relies on attestation or receives no signal at all.

E-discovery adds a legal dimension to the same mechanics. Litigation and regulatory inquiries require the tenant to produce records whose physical custody lies with the provider, spanning storage layers and sometimes jurisdictions. A second-level control that fixes formats, timelines, and scope of such production in advance turns an improvised negotiation under deadline pressure into a routine procedure. The observability scale makes the value of this preparation explicit, since without contractual anchoring, the same capability slides toward the third level, where the tenant can point to a certificate and to nothing else.

Notification timelines carry regulatory weight of their own. Personal data legislation across jurisdictions binds breach reporting of the tenant to fixed windows measured in days or hours, and the clock starts regardless of where the breach occurred. A tenant whose provider commits to second-level notification with defined deadlines inherits a chance to meet its own statutory window. A tenant without such a commitment discovers breaches late and from outsiders, and the observability scale states this difference in advance, at contract time, when it can still be repaired.

The practical consequence concerns preparation. A tenant that treats second-level controls as contractual objects, with response deadlines, artifact formats, and escalation contacts fixed in agreements, converts a weak observability position into an enforceable one. The SSCF section on security incident management, e-discovery, and cloud forensics, shaped through direct work in the working group, pursues exactly this conversion by describing the artifacts a customer should be able to

request from a provider before, during, and after an incident.

3.4. Identity and access management

Identity and access management occupies the opposite pole of the scale. With 21 controls, the domain is the largest in the SSCF, and its opening control requires visibility into user access for the customer (CSA, 2025). Enforcement of single sign-on, multi-factor authentication policies, session management, and automated provisioning through cross-domain identity management protocols leaves first-level evidence: the tenant configures these mechanisms itself and reads their operation in sign-in and provisioning journals. Open federation protocols made this maturity possible, since a control built on a public standard exposes its behavior through the standard itself.

The research literature explains why this domain matured earlier than others. A survey of identity and access management for cross-domain dynamic users traces federation, delegation, and revocation mechanisms that presuppose an active, configuring customer (Badirova et al., 2023). Work on decentralized access control in cloud environments extends the same principle and studies architectures in which policy evaluation itself becomes transparent to the resource owner (Punia et al., 2024). In terms of the proposed scale, the IAM domain demonstrates that first-level observability is achievable when the industry converges on open protocols and the customer holds the configuration.

Provisioning deserves the same attention as authentication. Automated account lifecycle management through cross-domain identity management protocols leaves a first-level trail of creations, updates, and deactivations, closing one of the oldest gaps in enterprise security: orphaned accounts that survive employee departures. A tenant that reads provisioning journals

verifies the deactivation of every leaver on its own schedule, and the same journals feed access recertification campaigns, which regulators in financial and public sectors expect as a standing process.

Session and token handling round out the first level of the domain. Configurable session lifetimes, forced re-authentication for sensitive operations, and revocation of active tokens give the tenant levers whose effect appears in the same journals that record ordinary sign-ins. The combination matters during containment, since the response to a compromised account in SaaS consists of exactly these actions, and a tenant that has rehearsed them against observable evidence executes them in minutes.

Residual opacity persists at the provider edge. Privileged access by provider personnel to underlying systems remains an attestation-only matter, and the tenant learns about it only from audit reports. The identity and access management section of the SSCF-to-CCM v4.1 mapping, completed with the same first-hand engagement, ties customer-facing IAM capabilities to the corresponding Cloud Controls Matrix requirements (CSA, 2021), thereby anchoring the most observable domain of the framework to the general catalog.

3.5. Cross-domain observability profile

Aggregation of the assignments from the three preceding subsections yields a compact profile of the examined domains. The profile is presented in Figure 2. The shares express a qualitative expert assessment of the control corpus assembled in the second section, which joins customer-facing SSCF capabilities with provider-side safeguards from the reference catalogs, and they serve as an illustration, since none of the source documents publish per-level statistics, and independent coding of the control texts remains a task for future work.

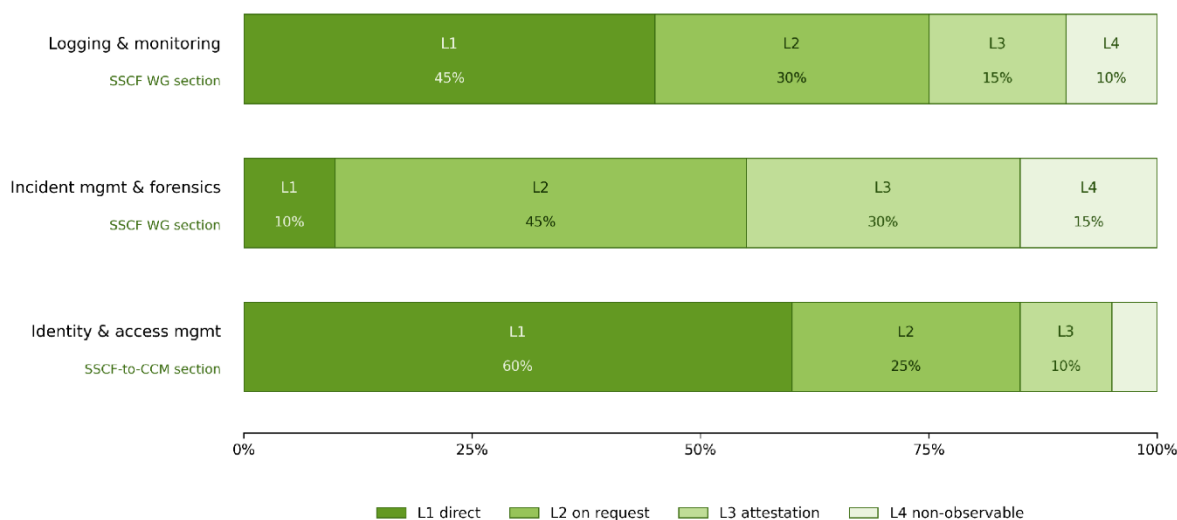


Fig. 2. Distribution of controls across observability levels in the three examined domains, per expert assessment

The profile makes the asymmetry of the SaaS assurance landscape visible. Identity and access management concentrates its mass at the first level; logging and monitoring spread between the first and second levels; and the incident domain shifts toward the second and third levels. The asymmetry has a practical reading. A tenant assessing a SaaS platform should expect strong self-service evidence for identity, partial evidence for telemetry, and contract-mediated evidence for incidents; deviations from this pattern warrant questions during procurement, as they signal either an advanced platform or a gap in the offering.

The profile also guides the sequencing of tenant investments. Capability building should start where the evidence already flows, in identity journals and exported logs, because analytic skills mature fastest on continuous data. Contractual work should concentrate on the incident domain, where the scale shows that agreements create most of the achievable assurance. Attestation review remains necessary across all three domains, and the profile shows its proper place, coverage of the residue that neither self-service evidence nor contracts reach.

3.6. Practitioner observations

Between 2022 and 2025, direct leadership of a security program across the SberTech SaaS portfolio named in the second section, with more than 100,000 active users in scope, generated the observations reported below. The program predates the publication of the SSCF and of the present taxonomy, and it pursued the direction that both documents later codified, so its results read here as a retrospective application of the scale. In these terms, the program moved controls upward. Audit log delivery and

schema documentation reached the first level; incident notification and evidence provision were formalized at the second; and secure development lifecycle checks, with automated code and vulnerability scanning, were wired into pipelines whose results the platform teams could read continuously.

Multi-year portfolio observations ground the assertion that the number of platform vulnerabilities declined by close to 30% over the period, alongside measurable progress in service-level readiness for incident response. The figure describes a long-term trend across the whole portfolio; no single scanner export can reproduce it, and the article therefore presents it as practitioner testimony. Attribution requires the same caution, since secure coding, automation, and awareness activities advanced in parallel, and the observability agenda formed one strand among several.

Service-level readiness improved through the same mechanism. Formalized notification and evidence procedures gave incident response a measurable external contour, from first message to final report, and internal drills began to rehearse the production of customer-facing artifacts in addition to technical containment. The teams learned to treat the customer as a consumer of evidence, and this framing outlived individual incidents and settled into the platforms' operating rhythm.

The qualitative lesson survives these caveats. Teams change behavior when evidence becomes visible to those who depend on it. Exported logs invited platform teams to monitor their own authentication anomalies,

published schemas turned security telemetry into a shared artifact between security and engineering, and formalized incident artifacts gave customer-facing units a concrete promise to defend against. Observability served as an organizational instrument in addition to a technical one, and this dual role explains why moving a control up the scale pays off beyond the audit context.

4. Discussion

For tenants, the taxonomy converts a diffuse demand for transparency into a checklist with a clear order. Procurement questionnaires can ask which controls on a candidate platform sit at the first and second levels; contracts can specify evidence formats and deadlines for second-level controls; and internal risk registers can mark residual exposure at the third and fourth levels. The scale also connects to the continuous assurance agenda. Research on continuous auditing and certification of cloud services argues that point-in-time certificates age quickly in dynamic environments and that automated evidence collection enhances the reliability of certification for the auditor and the customer alike (Suhonen & Martínez, 2024). First-level controls supply the raw material that such auditing consumes, so a tenant that negotiates log export today prepares its continuous audit pipeline for tomorrow.

For providers, the scale describes a migration path. Raising a control from the third level to the first requires product work, from event schemas and export endpoints to administrator-facing dashboards, and the SSCF demonstrates that the industry now expects this work (CSA, 2025). Competitive dynamics reinforce the expectation. A platform that controls a customer can shorten security review cycles, reduce the load of bespoke questionnaires on both sides, and turn transparency into a sales argument in regulated segments, where the customer must demonstrate oversight of its processors.

Regulated industries sharpen every one of these considerations. Financial supervision, health data protection, and personal data legislation oblige the customer of a processor to demonstrate oversight, and demonstration requires evidence in the literal sense. An observability profile for a platform translates supervisory expectations into product requirements that procurement can price and legal teams can enforce. In this translation, the taxonomy serves as an interface between compliance language and engineering language, two vocabularies

whose mutual opacity costs enterprises real money during every audit cycle.

A note on incentives completes the picture for providers. Exposure of controls carries costs beyond engineering, since visible evidence invites questions and binds the provider to the disclosed behavior. The taxonomy praises this honesty. Each control moved to the first level removes one item from every future questionnaire for every customer, and the aggregate savings grow with the customer base, which is why the largest platforms moved earliest and why the SSCF captures a direction the market had already chosen (CSA, 2025).

Within the broader assurance landscape, the observability axis stays orthogonal to functional classifications. The Cloud Controls Matrix and NIST SP 800-53 indicate which safeguards exist (CSA, 2021; Joint Task Force, 2020), and the proposed scale indicates which of them a tenant can witness. The zero trust paradigm offers a complementary lens. Comprehensive surveys of zero trust architecture place continuous verification at the center of the model (Syed et al., 2022), NIST SP 800-207 formalizes the rejection of implicit trust inside a perimeter (Rose et al., 2020), and a recent systematic review documents implementation of these principles across application domains (Mushtaq et al., 2025). The taxonomy extends the same philosophy across the organizational boundary: a tenant should withhold implicit trust in a provider wherever explicit evidence channels exist. A multidimensional survey of zero-trust security draws the corresponding conclusion for device ecosystems under the verify-and-trust formula (Azad et al., 2024), and the observability scale specifies what verification means for each class of SaaS controls.

Auditors also gain a planning instrument. An engagement scoped with the observability scale in hand can allocate scarce on-site effort to third-level controls, where the auditor substitutes for the customer's missing eyes, and verify first-level controls through the same interfaces the tenant uses, at a fraction of the cost. Continuous audit programs push this logic to its end and consume first-level evidence around the clock (Suhonen & Martínez, 2024). The scale, therefore, aligns the interests of three parties that assurance practice treats in isolation, the tenant, the provider, and the

auditor, around one question: the question of where evidence lives.

The relation between the scale and service-level agreements deserves a separate remark. An agreement defines obligations, and the scale describes their verifiability, so the two instruments complement each other. A service-level clause about incident notification without a second-level evidence channel remains a promise the tenant cannot test, and a first-level control without contractual status remains a courtesy the provider can withdraw. Mature procurement binds the two together, level by level, and the taxonomy supplies the ordering in which this binding pays off fastest.

Four limitations bound the results. The taxonomy covers three of the six SSCF domains, the two shaped by direct work within the framework and the one covered by the matching contribution to the SSCF-to-CCM mapping, and the remaining domains await the same treatment. The distribution in Figure 2 rests on expert assessment of the published control set and awaits formal coding by independent raters. The reduction of vulnerabilities described in the third section has the status of a practitioner assertion about one corporate environment and admits alternative explanations. The SSCF itself is young, version 1.0 appeared in 2025, and its adoption trajectory will test the assumptions built into the scale (CSA, 2025).

Future work follows from these limits. Extension of the scale to the remaining SSCF domains, formalization of a numeric observability index that would aggregate per-control levels into a platform score, and empirical validation of the link between observability and security outcomes across many tenants form the immediate agenda. Integration of such an index into continuous audit pipelines would close the loop between the taxonomy and the assurance practice it serves (Suhonen & Martínez, 2024), and comparative application of the index to competing platforms would give the market a public measure of transparency.

5. Conclusion

The article introduced a four-level scale of customer observability for SaaS security controls and applied it to three domains of the SaaS Security Capability Framework: logging and monitoring, security incident management with e-discovery and cloud forensics, and identity and access management. The scale distinguishes directly

observable controls, controls verifiable on request, attestation-only controls, and non-observable controls, and it grounds each level in the verifier's identity, the cadence of verification, and the type of available evidence. The application showed a stable asymmetry of the SaaS assurance landscape. Identity management offers the strongest self-service evidence, telemetry stands in the middle, and incident handling depends on provider-mediated artifacts. Practitioner observations from a multi-year corporate program illustrated how deliberate movement of controls up the scale reshapes engineering practice and organizational behavior around security evidence. A search of the published literature revealed no earlier systematic classification of SaaS controls along the observability axis. The taxonomy gives tenants, providers, and auditors a common coordinate system in which requirements for transparency can be stated, priced, and verified. Movement of controls along this axis turns the shared responsibility model from a declaration into a working agreement between the parties.

References

1. Alghofaili, Y., Albattah, A., Alrajeh, N., Rassam, M. A., & Al-rimy, B. A. S. (2021). Secure cloud infrastructure: A survey on issues, current solutions, and open challenges. *Applied Sciences*, 11(19), 9005. <https://doi.org/10.3390/app11199005>
2. Alshabibi, M. M., Bu dookhi, A. K., & Hafizur Rahman, M. M. (2024). Forensic investigation, challenges, and issues of cloud data: A systematic literature review. *Computers*, 13(8), 213. <https://doi.org/10.3390/computers13080213>
3. Azad, M. A., Abdullah, S., Arshad, J., Lallie, H., & Ahmed, Y. H. (2024). Verify and trust: A multidimensional survey of zero-trust security in the age of IoT. *Internet of Things*, 27, 101227. <https://doi.org/10.1016/j.iot.2024.101227>
4. Badirova, A., Dabbaghi Varnosfaderani, S., Fatemi Moghaddam, F., Wieder, P., & Yahyapour, R. (2023). A survey on identity and access management for cross-domain dynamic users: Issues, solutions, and challenges. *IEEE Access*, 11, 61660–61679. <https://doi.org/10.1109/ACCESS.2023.3279492>
5. Chauhan, M., & Shiaeles, S. (2023). An analysis of cloud security frameworks, problems and proposed solutions. *Network*, 3(3), 422–450. <https://doi.org/10.3390/network3030018>
6. Cloud Security Alliance. (2021). *Cloud Controls Matrix and CAIQ v4.0*. Cloud Security Alliance.

7. Cloud Security Alliance. (2025). *SaaS Security Capability Framework (SSCF), version 1.0*. Cloud Security Alliance.
8. Humayun, M., Niazi, M., Almufareh, M. F., Jhanjhi, N. Z., Mahmood, S., & Alshayeb, M. (2022). Software-as-a-service security challenges and best practices: A multivocal literature review. *Applied Sciences*, 12(8), 3953. <https://doi.org/10.3390/app12083953>
9. International Organization for Standardization. (2022). *ISO/IEC 27001:2022: Information security, cybersecurity and privacy protection – Information security management systems – Requirements*. ISO.
10. Jeong, B., Kim, J., Lee, S., & Park, J. (2026). FOREST: Inspecting and tracking RESTful APIs for constructing a cloud forensic knowledge base. *Forensic Science International: Digital Investigation*, 56, 302070. <https://doi.org/10.1016/j.fsidi.2026.302070>
11. Joint Task Force. (2020). *Security and privacy controls for information systems and organizations* (NIST Special Publication 800-53, Rev. 5). National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.SP.800-53r5>
12. Mushtaq, S., Mohsin, M., & Mushtaq, M. M. (2025). A systematic literature review on the implementation and challenges of zero trust architecture across domains. *Sensors*, 25(19), 6118. <https://doi.org/10.3390/s25196118>
13. Punia, A., Gulia, P., Gill, N. S., Ibeke, E., Iwendi, C., & Shukla, P. K. (2024). A systematic review on blockchain-based access control systems in cloud environment. *Journal of Cloud Computing*, 13, 146. <https://doi.org/10.1186/s13677-024-00697-7>
14. Rose, S., Borchert, O., Mitchell, S., & Connelly, S. (2020). *Zero trust architecture* (NIST Special Publication 800-207). National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.SP.800-207>
15. Suhonen, T., & Martínez, C. (2024). Continuous auditing and continuous certification of cloud services in MEDINA: Security auditor's view. *Open Research Europe*, 3, 208. <https://doi.org/10.12688/openreseurope.16703.2>
16. Syed, N. F., Shah, S. W., Shaghaghi, A., Anwar, A., Baig, Z., & Doss, R. (2022). Zero Trust Architecture (ZTA): A comprehensive survey. *IEEE Access*, 10, 57143–57179. <https://doi.org/10.1109/ACCESS.2022.3174679>
17. Tuyishime, E., Balan, T. C., Cotfas, P. A., Cotfas, D. T., & Rekeraho, A. (2023). Enhancing cloud security: Proactive threat monitoring and detection using a SIEM-based approach. *Applied Sciences*, 13(22), 12359. <https://doi.org/10.3390/app132212359>