S

Security as a Service (SECaaS): Models, Risks, and Selection

Security as a Service delivers security capabilities through a hosted subscription. The provider operates technology, but the customer retains important decisions and duties.

PUBLISHER
Gridinsoft Support

UPDATED
Oct 8, 2026

READ TIME
3 min

Security as a Service (SECaaS) delivers security capabilities through a hosted, subscription, or usage-based service. Examples include identity and access management, email and web security, data loss prevention, security monitoring, encryption, vulnerability management, network controls, and disaster-recovery capabilities.

SECaaS can reduce infrastructure and maintenance work, but it does not transfer all security accountability to a provider. Effectiveness depends on correct integration, customer policy, reliable data, alert ownership, and a documented shared-responsibility model.

Common SECaaS categories

- identity, MFA, privileged access, and access-policy services;
- email, web, DNS, API, and cloud-access security;

- endpoint detection, vulnerability scanning, and exposure management;
- SIEM, log analytics, threat intelligence, and managed detection;
- data protection, key management, backup, and recovery services;
- network firewall, DDoS protection, secure access, and segmentation controls.

The Cloud Security Alliance's SECaaS guidance groups services into ten foundational categories. Current products often combine several categories or deliver them through cloud-native platforms.

SECaaS vs. SaaS, MSSP, and MDR

- **SaaS** is a general software delivery model; SECaaS is specifically a security capability delivered as a service.
- **MSSP** describes a provider operating or monitoring security controls for a customer, including tools hosted by either party.
- **MDR** emphasizes human-led detection, investigation, and response, often using endpoint, identity, and cloud telemetry.

A vendor may use several labels. The contract and operating procedure should specify who monitors, who decides, who can contain systems, and when the customer is contacted.

Shared responsibility

The provider is typically responsible for service infrastructure, platform availability, tenant isolation, service updates, and protection of its administrative plane. The customer commonly remains responsible for account lifecycle, privileged access, configuration, data classification, endpoint coverage, legal basis for processing, alert response, and recovery decisions. Responsibilities vary by service and must be written down.

Ask how subcontractors, support engineers, APIs, customer-managed keys, and emergency access affect that boundary. A certification or audit report supports due diligence but does not prove that your tenant is configured correctly.

Provider evaluation checklist

1. **Scope:** Which assets, users, regions, data types, and threats are covered or excluded?
2. **Identity:** Are SSO, phishing-resistant MFA, role separation, approval, and administrative audit logs supported?
3. **Data:** Where are content and telemetry processed, encrypted, retained, backed up, and deleted?
4. **Detection and response:** Who reviews alerts around the clock, what evidence is supplied, and which containment actions require approval?
5. **Assurance:** Review architecture, penetration-test handling, audit reports, vulnerability disclosure, incident history, and notification commitments.
6. **Reliability:** Validate rate limits, regional dependencies, degraded modes, recovery targets, and support escalation.
7. **Portability:** Confirm export formats for logs, rules, cases, keys, and configurations, plus deletion evidence at termination.

Pilot and operate the service

Start with measurable use cases and a representative pilot. Test onboarding, policy errors, alert delivery, containment approval, provider outage, lost connectivity, and restoration. Compare detections with existing controls and calculate total cost including ingestion, retention, integration, training, incident hours, and exit work.

Assign internal owners even for a fully managed service. Review access and configurations regularly, reconcile licensed assets with actual coverage, and exercise incident contacts. Maintain independent copies of evidence needed during a provider outage or contract dispute.

Risks and tradeoffs

- provider concentration can make one outage or compromise widely disruptive;
- limited telemetry or opaque detections can slow investigation;

- misconfiguration and uncovered assets create false confidence;
- data residency, retention, and cross-border support may create compliance obligations;
- proprietary rules and case data can make migration expensive.

SECaaS FAQ

Does SECaaS replace an internal security team?

No. Someone inside the organization must own risk, business context, access decisions, response authority, and provider oversight.

Is a larger feature list always better?

No. Prefer verified coverage, usable evidence, reliable operations, and clear responsibilities for the risks that matter.

NEED MORE HELP?

We are here when the guide is not enough.

Visit the Gridinsoft Support Center for current product guidance or submit a support request with the details of your case.

Open this guide online at support.gridinsoft.com