

Table of Contents

About This Report	01
Executive Summary	02
The Rise of SaaS Breaches: Attackers Are Targeting the Weakest Link	04
The High Cost of Low Visibility: SaaS Misconfigurations	08
CASBs Are Inadequate to Address Modern SaaS Misconfigurations	10
Distributed SaaS Management: Balancing Productivity with Security	13
The Scope of Sensitive Data in SaaS: Wider Than You Might Think	16
Non-Human Identities Outnumber Humans and Are Prime Targets	18
The “Write” Stuff: Security Concerns in The Rise of Generative AI	20
Leaving the Door Open: Ineffective Lifecycle Management and Offboarding	22
Key Recommendations Moving Forward	24
About Valence Security	26

About This Report

Software as a Service (SaaS) is used today by every organization to power almost every aspect of their business. SaaS applications like Microsoft 365, Google Workspace, Salesforce, GitHub, Slack, and Atlassian have become ingrained in business operations, elevating productivity and efficiency. The SaaS cost-effective subscription model makes it easy to scale business systems and services from any web browser on any device, anywhere in the world to fuel their efficiency and growth.

What makes SaaS so great also makes it challenging to secure. Recent high-profile breaches demonstrate a critical truth: SaaS applications have become a prime target, but many security programs lack critical capabilities to properly protect and secure SaaS. These incidents exposed source code, sensitive data and customer data, disrupted operations, and led to reputation damage and lawsuits, highlighting the potential impact of SaaS security misconfigurations and weak points.

Because an organization's security is only as strong as its weakest link, it's imperative that SaaS applications don't give attackers an easy opening to exploit. In SaaS, security is a shared responsibility between the SaaS provider who secures the underlying infrastructure and the customer who manages their user access, data security within the application, and overall secure configuration. This shared responsibility model presents numerous challenges:

- **Limited Standardization:** Lack of standardized security configurations and best practices, making it difficult to achieve consistent security across SaaS applications.
- **Third-Party Integrations:** SaaS-to-SaaS interconnectivity leverages non-human identities to automate processes, but some security methods such as MFA cannot be applied.
- **Configuration Complexity:** Each SaaS application has its own unique configuration, terminology, and permissions, requiring specialized knowledge to enforce security controls.
- **Data Everywhere:** Organizations' highly sensitive data is stored in SaaS applications. While this makes it easy to share the data externally, it introduces a sprawl of overshared data.
- **Distributed Management:** SaaS applications are often administered by business units, leading to potential misconfigurations with limited visibility and control for security teams.

These inherent challenges, coupled with recent high-profile SaaS security breaches, highlight the critical need for robust SaaS security practices. This report explores these challenges, potential risks, and best practices to navigate the modern SaaS security landscape.

Research Methodology

In this report, we will review the recent trends, threats and breaches in the SaaS security landscape. This report was compiled with primary research that includes results of a survey conducted by Enterprise Management Associates (EMA) from 125 security executives as well as anonymous data from 2024 collected from hundreds of real enterprise SaaS applications by the Valence SaaS Security Platform.

Executive Summary

SaaS Security: Confidence Gap and Critical Risks

This report analyzes current SaaS security trends and exposes a concerning gap between security leaders' confidence in existing processes and their ability to fully secure SaaS applications. Our findings highlight the most prevalent SaaS security risks based on a survey of senior cybersecurity executives and data analysis from the Valence SaaS Security Platform.



Key Findings:

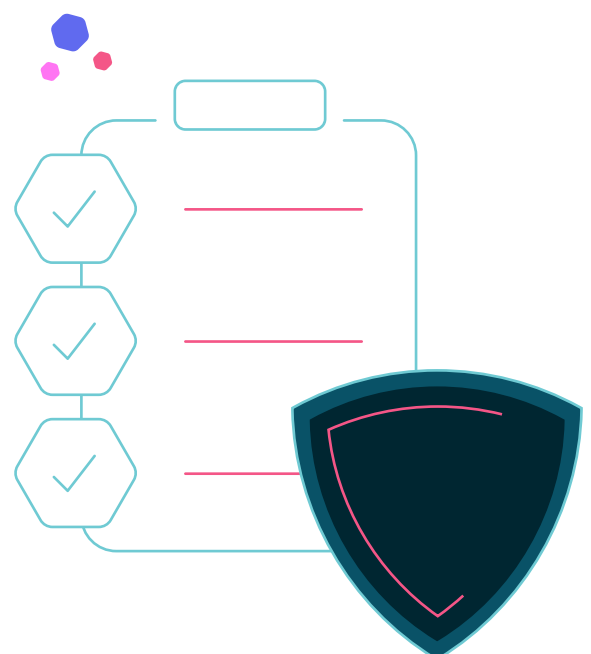
- **High Priority, High Confidence:** Security executives are prioritizing SaaS security with **96%** identifying it as a high or top priority in their organization. In fact, **93%** of survey respondents said their organization's budget for SaaS security had increased as compared to previous budget years. In addition, confidence in current SaaS security programs or processes is high (**84%** are either "extremely" confident or "very" confident).
- **Rise of SaaS Breaches:** There is a notable difference between perception and reality when it comes to SaaS security. Even with the very high confidence levels and high prioritization of SaaS security, more than half (**58%**) of organizations experienced a SaaS security incident within the past 12-18 months.
- **High Profile SaaS Breaches:** Historically, SaaS applications have been overlooked in terms of security programs which leads to increased misconfigurations and exposure of sensitive data. Recent high profile breaches, such as the Microsoft Midnight Blizzard breach and the Cloudflare breach following the Okta attack campaign, highlight that malicious actors have identified SaaS as a new prime target.
- **Top SaaS Security Challenges:** Security executives noted the following security challenges as their most difficult in securing SaaS applications:
 - ⊗ Half (**50%**) identified distributed management of SaaS applications outside of IT/security teams as one of their top 3 challenges
 - ⊗ Half (**50%**) indicated that governing Generative AI adoption is a top challenge
 - ⊗ Nearly half (**43%**) indicated the complexity of SaaS configurations as one of their top challenges
- **Shift in Tools to Secure SaaS:** When asked which tools their organizations are using to protect their SaaS applications **52%** said CASB solutions while **48%** said SSPM solutions. While CASBs have been around for over a decade longer, the fact that SSPMs have a similar adoption rate highlights a significant shift in how organizations address SaaS security.

Executive Summary

- **Inactive SaaS Data Shares:** SaaS makes it easy for users to share data with external collaborators - but when was the last time somebody unshared a file or resource? Our research shows that the vast majority of external data shares (**94%**) are inactive, meaning that people have access to these files, folders, recordings, records, etc. when they don't need it anymore. This unnecessary exposure indicates the importance of proper lifecycle management of shared resources and access to them.
- **Overprivileged Third-Party SaaS Integrations:** All organizations (**100%**) grant access via API to at least one of their third-party vendors. This includes access to files, emails, admin privileges and more - **33%** of these integrations are granted access to sensitive permissions and data. SaaS-to-SaaS integrations are increasingly targeted by attackers since traditional access controls are less effective when it comes to these non-human identities and organizations typically don't have the same monitoring capabilities as they have for human identities.
- **Emerging GenAI Threats:** Security concerns around GenAI include limited visibility, data exposure risks during AI training, and potential for unintended data output. Nearly all (**90%**) of security executives said their organizations have implemented a GenAI governance policy, highlighting a growing awareness of the need for proactive security measures in this evolving space. Most GenAI tools are offered as SaaS applications and they are often integrated with business-critical SaaS applications to gain access to sensitive data.

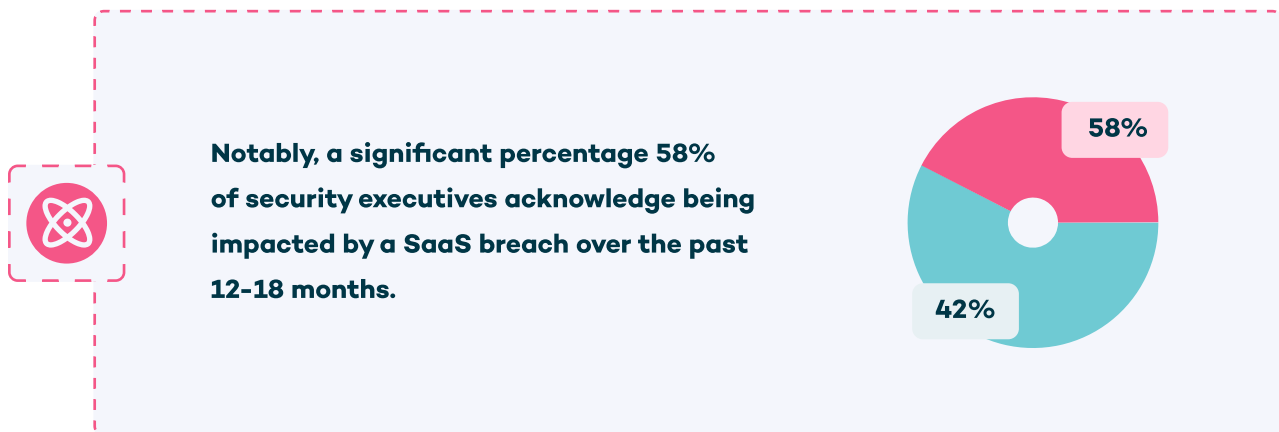


Understanding these inherent challenges is crucial, especially in light of the recent high-profile SaaS security incidents detailed in the following section. These breaches and misconfigurations illustrate the real-world consequences of inadequate SaaS security practices.



The Rise of SaaS Breaches: Attackers Are Targeting the Weakest Link

In recent years, we have witnessed a widening series of attacks targeting SaaS applications, which often impacted the customers of SaaS vendors. Notably, a significant percentage (**58%**) of security executives acknowledge being impacted by a SaaS breach over the past 12-18 months. The breaches detailed below illustrate how a compromise in one organization can have cascading effects, impacting other interconnected organizations. The breaches highlight the critical need for robust SaaS security practices, as misconfigurations, excessive privileges and access, unsecure non-human identities, and other security risks can have devastating consequences.



Let's take a closer look at some of the most prominent recent SaaS security incidents:

1 CircleCI Breach (December 2022)



Hackers stole OAuth tokens from CircleCI production systems and leveraged these non-human identities to gain unauthorized access to GitHub tenants of CircleCI's customers.

CircleCI, a popular continuous integration and delivery (CI/CD) platform, experienced a data breach that resulted in the theft of customer data. Malware on a compromised employee's laptop led to attackers stealing a session cookie, enabling them to impersonate the employee, bypass multi-factor authentication (MFA) protection, and gain access to CircleCI's internal production systems. This unauthorized access allowed the attackers to steal customer data, including environment variables, tokens, and API keys. CircleCI became aware of these events by a customer reporting suspicious GitHub OAuth activity their security team discovered.

The breach highlights the risk associated with stolen OAuth tokens, emphasizing the need for robust management practices of non-human identities, including tokens and API keys, in addition to regular rotation of these credentials. Additionally, the incident underscores the importance of enforcing strong endpoint security measures and implementing strict multi-factor authentication (MFA) policies with minimal exceptions, especially for privileged accounts.

2 Microsoft-Storm-0558 Breach (June 2023)



Using a stolen signing key to forge Azure AD tokens, attackers compromised Microsoft 365 and breached customer emails, highlighting the importance of tracking access tokens.

A China-based threat actor (Storm-0558) compromised Microsoft 365 and stole email data from 25 of its customers, specifically US government agencies and private companies. The attackers used some novel exploits to achieve this attack by abusing signing and access keys. The attacker used a stolen signing key to forge Azure AD tokens. The term forge is important here, because a validation issue in Microsoft's code made it possible to create and use invalid tokens. As long as these tokens were properly signed, Microsoft would allow their use. The forged Azure AD tokens were then used to generate access tokens and steal emails via the Outlook Mail API.

This breach emphasizes the importance of tracking access tokens, particularly newly generated ones, which helps to ensure that your environment is not accessed by parties in ways that you did not intend, potentially preventing data loss. It also illustrates the importance of maintaining and utilizing security controls that are accessible to you, as the customer of a SaaS platform. and the dangers of a single compromised account impacting multiple entities.

3 MGM and Caesars (September 2023)



Social engineering attack led to compromised Okta super admin accounts, which allowed attackers to launch a ransomware attack on the Las Vegas casino giants.

Two Las Vegas casino giants, MGM Resorts and Caesars, suffered a devastating cyberattack, leading to extensive outages and disruptions across their internal networks, including: ATMs, slot machines, digital room key cards, and electronic payment systems. MGM estimates the costs of the breach exceeded \$100 million. Voice phishing compromised an MGM employee's Okta account, granting attackers super admin access to launch a ransomware attack that crippled the Las Vegas casinos' operations. MGM Resorts' IT team shut down its systems after detecting attackers had compromised its Okta servers. This supposedly resulted in MGM Resorts being locked out of its Okta tenant, while the attackers were able to retain super administrator privileges as well as global admin rights to MGM Resorts' Microsoft Azure tenant.

The key to preventing this incident was preventing the initial access. According to reports, attackers were able to vish (phishing via voice call) a service desk agent without being forced to authenticate themselves via another factor. Implementing strict Multi-Factor Authentication (MFA) for all access points within your SaaS environment adds a crucial layer of security to prevent unauthorized access.

4 Okta Support System Breach (October 2023)



Attackers abused stolen session cookies to gain admin access to Okta customers such as Cloudflare, 1Password, and BeyondTrust, following a breach into Okta's internal systems.

Attackers utilized stolen credentials to breach Okta's support case management system. This system houses HAR files containing session cookies, which were accessed by the attackers.

This unauthorized access prompted the attackers to shift their focus towards targeting Okta's customer base. Cloudflare (see below), 1Password, and BeyondTrust confirmed that hackers used stolen session cookies from the Okta HAR files—used to impersonate in-house Okta administrator accounts, and bypass multi-factor authentication (MFA)— to target their systems as a result of the breach.

This incident underscores the critical role of robust access management in securing SaaS applications. Furthermore, it's recommended that threat detection and monitoring controls include notifications when new accounts are created, or any significant changes to administrative access/authorization, as well as any strange behavior coming from accounts with admin-level privileges.

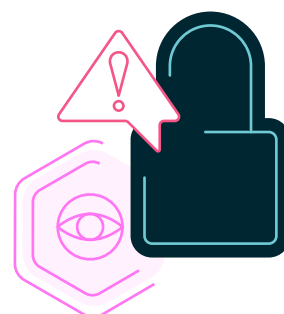
5 Cloudflare (Post-Okta Breach) (January 2024)



Leveraging compromised service account credentials from the Okta breach, attackers breached Cloudflare's internal systems, accessing sensitive data like source code.

Leveraging compromised credentials from Okta, nation-state attackers breached Cloudflare's Atlassian Bitbucket, Confluence and Jira platforms, accessing sensitive data like source code. The attackers leveraged a service token and service account credentials that leaked during the Okta breach, one of which was granted to allow the SaaS application Smartsheet to have administrative access to Cloudflare's Atlassian systems. Following the Okta compromise, the Cloudflare security team performed an in depth forensics analysis and rotation of more than 5,000 production credentials. Unfortunately, the team missed 4 credentials of a service token/service accounts that belonged to SaaS applications and other services.

This breach highlights the importance of SaaS security for both primary and third-party integrations, and the need to manage non-human identities and service accounts carefully, revoking any privileges that are not currently needed.



6 The Microsoft Midnight Blizzard Breach (January 2024)

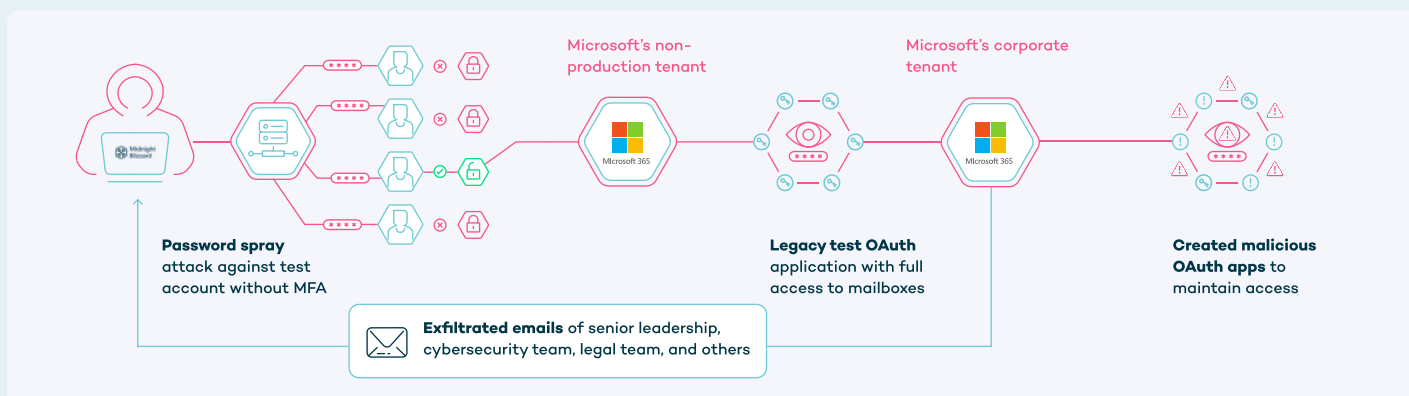


One account without MFA in a test environment led to abused OAuth tokens and unauthorized access to Microsoft's senior leadership's corporate email.

In January 2024, a nation-state actor was able to access corporate email accounts of Microsoft's senior leadership. This is a classic example of a SaaS breach that doesn't leverage a vulnerability exploitation or zero days, but rather exploited misconfigurations in a SaaS platform.

According to Microsoft, the Russian state-sponsored threat actor Midnight Blizzard was able to ultimately gain access to corporate email accounts, including members of senior leadership, the cybersecurity team, the legal team, and others, and exfiltrated some emails and attached documents.

The attackers initially gained access to Microsoft's non-production Microsoft 365 test tenant via a password spray attack targeting a human account that didn't have multi-factor authentication (MFA) enabled. From there, the attackers leveraged a legacy test OAuth application (non-human identity) that had full access to Microsoft's corporate production Microsoft 365 tenant with privileges to read emails. The threat actor also created additional malicious OAuth applications and granted them access to Microsoft's corporate environment using newly created user accounts. This helped them to authenticate to Microsoft Exchange Online and target Microsoft corporate email accounts. The attackers also used residential proxy networks to obfuscate the source of their attack and leverage IP addresses of legitimate users.



The Midnight Blizzard breach underscores the importance of a holistic approach to SaaS security. Organizations must consider human and non-human identities, third-party integrations, and potential misconfigurations across their entire SaaS landscape to effectively mitigate the evolving threat landscape. It's crucial that security teams continuously audit the current privilege level of non-human identities to understand which identities are highly privileged. Adhering to least privilege principles is of course recommended as well.

The Midnight Blizzard breach also exemplifies the dangers of overlooking seemingly "low-risk" elements like neglected resources. These can include dormant accounts, legacy tokens, and inactive data shares. Security teams often prioritize active resources for protection, assuming attackers target them first, but neglected resources offer a potentially easier path for attackers. This emphasizes the importance of regularly reviewing and removing unnecessary resources like dormant accounts and inactive data shares to minimize potential attack surfaces.

The High Cost of Low Visibility: SaaS Misconfigurations

While the previous section explored notable SaaS breaches, there have been many published SaaS misconfigurations that could lead to similar consequences. In the SaaS shared responsibility model, customers have less control compared to traditional applications and infrastructure. Therefore, security programs focus more on how they can manage the security controls that were made available to them instead of on software vulnerabilities. These security controls can be misconfigured and expose business-critical data. Typical misconfigurations are lack of MFA/SSO enforcement, overprivileged third-party access, unsecure default sharing settings, publicly available data, and many more.

It's worth noting that **43%** of survey respondents listed complexity of SaaS configurations as one of their biggest challenges. This is understandable since over the years, SaaS applications have evolved to become complex platforms with a wide range of capabilities and functionalities, and therefore more configurations. Each SaaS application has its own terminology, configurations, permissions, and logs, requiring deep understanding for effective security control and monitoring.

High-impact misconfigurations in recent years include:

1 Salesforce Sites Leak Private Data (April 2023)



KrebsOnSecurity published that security researcher Charan Akiri discovered many public Salesforce sites that are leaking private data. These Salesforce Community websites were misconfigured to allow unauthenticated guest users access to private records. Brian Krebs reported that Vermont state and Washington D.C. exposed sensitive data such as names, SSNs, and bank account info. The research identified hundreds of organizations running misconfigured Salesforce pages.

2 ServiceNow Publicly Exposed Data (October 2023)



Another case of misconfigured unauthenticated (guest) user access was published in ServiceNow. A misconfiguration in ServiceNow's Access Control List (ACL) and a default setting in the Simple List widget, which makes data from tables available in dashboards, allowed public access to data stored in ServiceNow tables. This misconfiguration has existed since the introduction of the widget in 2015, but was only discovered in 2023 and ServiceNow quickly published a response to remediate the issue once it was discovered.

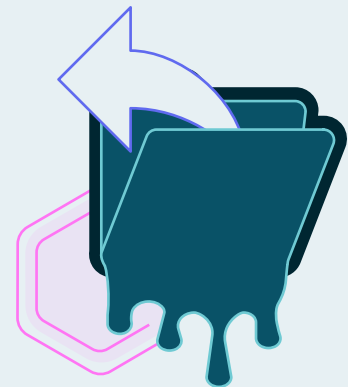


Data about one million people was exposed for more than six years due to a Google Drive “anyone with a link” anonymous sharing misconfiguration.

Japanese game developer Ateam exposed information about nearly one million people due to a misconfiguration in Google Drive. For over six years, a misconfigured Google Drive instance - which allowed files stored in it to be viewed by anyone with a link (anonymous access) - left the personal data of nearly one million individuals exposed, including customers, business partners, job applicants, and even current employees. This included sensitive information like full names, email addresses, phone numbers, and customer management numbers. The potential impact is significant - exposed data can be used for identity theft, fraud, targeted phishing attacks, or sold on the dark web.

This incident highlights the importance of managing external access to files and data stored in SaaS applications. In many applications such as Google Drive, but also in OneDrive, Box, Dropbox, and even in Zoom, Salesforce, and NetSuite, employees leverage the built in collaboration functionalities to share data with external third parties. Although it's easier to share with “anyone with a link” since the employee doesn't need to think in advance who needs access to the data, it's highly recommended to default to individual user access. By restricting access by default or with automated policy enforcement, organizations can significantly reduce the risk of accidental exposure. Furthermore, it underscores the need for granular access controls. Don't grant blanket access - carefully consider who needs access to specific data and for what purpose. The principle of least privilege should guide access control decisions: users should only have the access level necessary to perform their job functions. This also means that you should remove unnecessary shares when there is no longer business justification - for example, when the specific need or project has ended - so the shares don't linger forever.

Ateam's Google Drive misconfiguration is just one example of a larger trend. Organizations of all sizes and across various industries are susceptible to SaaS misconfigurations due to the complexity of SaaS environments and the dynamic nature of user permissions. These examples demonstrate that even minor misconfigurations can have real consequences. These misconfiguration examples illustrate that organizations must prioritize SaaS security to identify and remediate misconfigurations, safeguard data, and maintain user trust.



CASBs Are Inadequate to Address Modern SaaS Misconfigurations

Cloud Access Security Brokers (CASBs) have for years been considered a key component of cloud security and especially the go to solution for SaaS security, offering visibility and control over cloud application usage. CASBs primarily focus on user access control to these applications, while providing application discovery, data loss prevention (DLP), and threat protection capabilities.

While CASBs offer these valuable functionalities, they have limitations when it comes to operationalization and confronting modern SaaS security risks:

➤ SaaS Misconfigurations Blindspot



Since CASBs focus on user-to-SaaS access, they don't analyze SaaS configurations, leaving security teams blind to the complex modern SaaS platforms and their distinct set of security controls.

➤ Lack of SaaS-to-SaaS Visibility



SaaS interconnectivity is typically directly between SaaS apps and leverage non-human identities (OAuth, APIs, service accounts) that communicate outside of the purview of a CASB.

➤ Deployment Complexity



Many CASBs require an inline proxy component to offer their full capabilities, which adds significant complexity to the deployment process. In addition, most CASBs require time-consuming tuning, especially due to the DLP component.

➤ Limited Coverage



CASBs typically combine proxy and API based components, but most CASBs cover less than a couple dozen of SaaS applications, which means that they rely only on the proxy component to cover many business-critical SaaS, which has been proved to be partial coverage only with limited value.



SaaS misconfigurations in particular pose a unique challenge that traditional CASBs weren't designed to address. Each SaaS application operates in its own universe. They have distinct terminology, permission structures, logging formats, and security controls. As a result, this demands expertise in each platform to effectively monitor activity and identify misconfigurations.



In our survey, 43% of security executives identified the “Complexity of SaaS configurations” as one of their top SaaS security challenges.

Common misconfigurations include:

Overly Permissive Access Controls



Granting excessive access to sensitive data increases the attack surface and makes it easier for unauthorized users to gain a foothold.

Weak Authentication Practices



This includes a lack of enforced multi-factor authentication, Single Sign On (SSO), password complexity requirements, or other authentication controls.

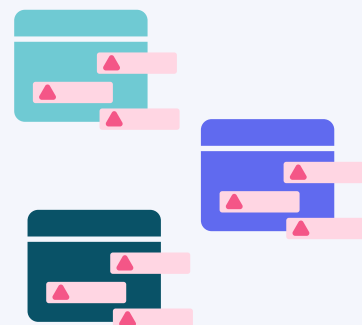
Dormant Accounts and Integrations



Dormant accounts, legacy API/OAuth tokens, and inactive data shares can create risks if they remain accessible. Attackers often target these "blind spots" because they are less likely to be monitored or secured.



In our survey, 48% of respondents indicated that “Tracking the changes in functionalities and risks in each SaaS application” was one of the top barriers for their teams to address SaaS security risks

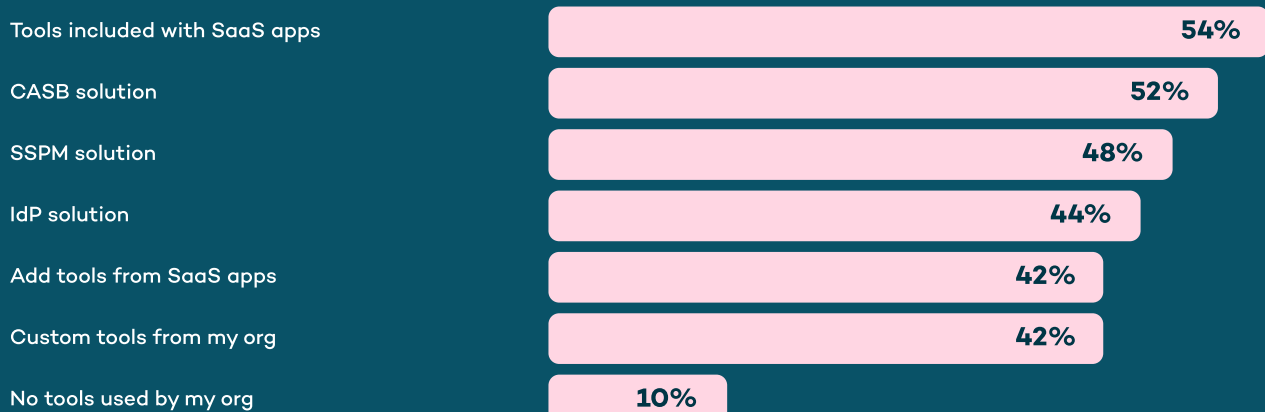


These misconfigurations, if left unaddressed, can have devastating consequences. Organizations need to prioritize a proactive approach to SaaS security, implementing automated tools and processes for continuous monitoring and configuration management. Regularly reviewing and updating access controls, enforcing strong password policies with MFA, and diligently managing unused accounts and integrations are all crucial steps towards reducing the risk of a security incident. Compounding these challenges is "configuration drift," the tendency for configurations to deviate from their secure state over time. This drift can introduce vulnerabilities that go unnoticed unless actively monitored and addressed.

The Tools Organizations Use to Secure Their SaaS Apps



In the survey, we asked: What tools/solutions are used by your organization to protect your SaaS applications (select all that apply)?



This survey highlights a significant shift in securing SaaS applications. While CASBs, a longstanding solution (12+ years), are still widely used (52%), the relatively new SSPM category (3-5 years old) has reached nearly the same adoption rate (48%). This rapid rise in SSPM adoption suggests security teams are recognizing limitations in traditional CASBs when it comes to securing SaaS applications.

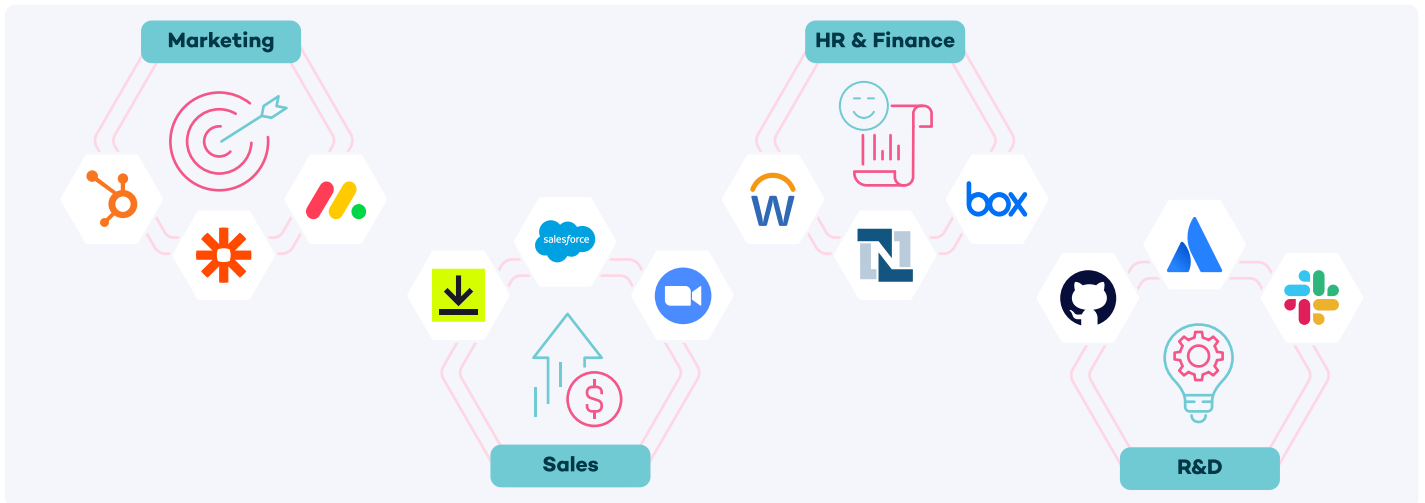
SSPM for SaaS Security

SaaS Security Posture Management (SSPM) solutions address the limitations of CASBs, particularly related to misconfigurations. SSPMs go beyond user access control by analyzing the security configurations within each SaaS application to identify and address security risks, and other deviations from best practices, before they are exploited. This proactive approach helps prevent misconfigurations from creating security risks in the first place, significantly reducing the time it takes to detect and remediate issues.



Distributed SaaS Management: Balancing Productivity with Security

One of the unique challenges for IT and security teams when it comes to SaaS security is that they often do not have visibility or governance over all SaaS platforms. They likely have ownership over SaaS applications like Microsoft 365 or Google Workspace, but what if they are not administrators of Workday, GitHub, Salesforce, or NetSuite? Those might be independently managed by the business teams like marketing, sales, finance, HR, or R&D teams. With increased adoption of business-critical SaaS applications for productivity and collaboration, non-security teams are likely to continue to have a primary role in owning and managing many of these SaaS applications.



Half (**50%**) of survey respondents identified distributed management of SaaS applications outside of IT/security teams as one of their top 3 challenges. The distributed management of SaaS offers efficiency benefits for business units, but it can also lead to security challenges. There are two main security challenges:

▶ Lack of Security Expertise for SaaS Admins



Business users managing SaaS applications typically lack the security expertise to configure them securely. This can lead to misconfigurations exposing sensitive data or granting unauthorized access. Business users, such as marketing or finance teams, in general prioritize functionality and ease of use over security best practices when managing SaaS applications. For example, a common misconfiguration by Salesforce admins is creating local user accounts that bypass the organization's SSO policy or the marketing onboarding a GenAI tool and granting it access to sensitive data.

▶ Limited Knowledge of SaaS Apps in the Security Team



The flip side is that since most security teams don't use or manage SaaS applications such as Salesforce, Workday, and NetSuite, they typically are less educated on their risks. This requires security teams to constantly educate themselves on these applications and limits their ability to guide the SaaS admins on best practices. In addition, security teams are less aware of the context of how each application is utilized within the organization, which makes it difficult to effectively prioritize risks and plan remediation steps.



Our survey found that 43% of security executives said that the largest barriers to remediating SaaS risks is the need to “understand the risk, the required fix, and the business implications.” Management of SaaS applications outside of security team governance makes this very difficult and requires close collaboration between security teams and business units.



Empowering Productivity With Secure SaaS Controls



The key to successful distributed SaaS management lies in striking a balance between empowering business units to be productive and agile while ensuring robust security practices across all SaaS applications.

1

The first step to create such balance is to ensure security teams have the required centralized visibility into SaaS security risks - without proper visibility, it's impossible to ensure security policies are enforced.

2

Next, is to define security policies across all different SaaS applications. It's crucial that these policies not only limit the SaaS functionalities due to security controls, but also define how to enable business users to securely leverage SaaS benefits. For example, defining how data can be shared externally or how to integrate third-party vendors. Just blocking functionalities by default without alternatives typically leads to creative bypasses which is an undesired result.

3

Lastly, partnership and collaboration between security teams, SaaS admins, and business users is critical. Empowering business users with self-served guardrails that help them understand when something is not properly secured and guides them on how to fix any risks.

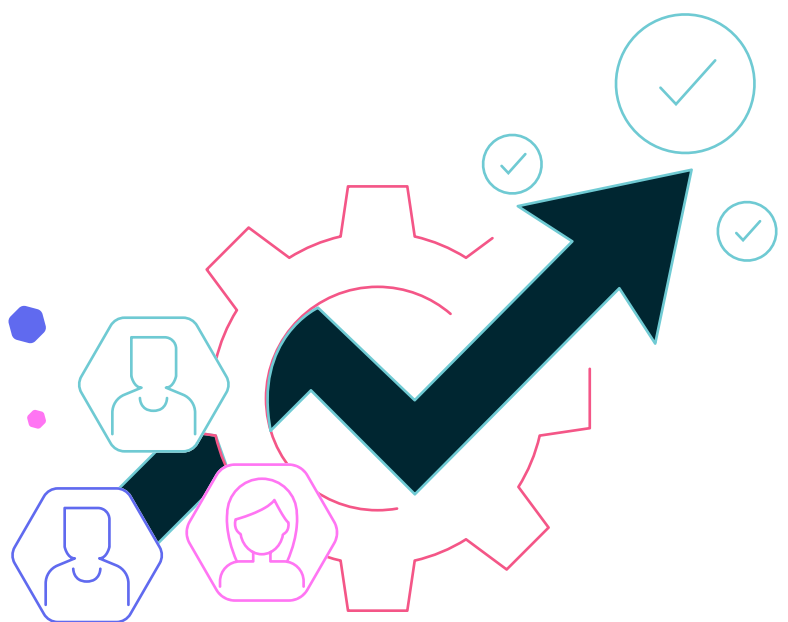
An interesting window into this collaboration is by looking at users of the Valence SaaS Security platform. Almost three-quarters (71%) of Valence customers grant non-security team SaaS administrators with access to the relevant SaaS applications they manage. In total, these SaaS administrators make up 31% of Valence users. Some of the success stories we've seen in the field include creating pre-configured security templates with best practices for common SaaS applications which are shared with business users. This reduces the risk of misconfigurations due to a lack of expertise.



Almost three-quarters (71%) of Valence customers grant non-security team SaaS administrators with access to the relevant SaaS applications they manage.


This transparency and shared understanding foster trust and collaboration between teams. Security teams can shift from a role of solely blocking actions to one of providing guidance and support. Business users can be empowered to make informed decisions about application usage while adhering to security best practices.

By focusing on empowerment and collaboration, organizations can cultivate a culture of security awareness across all departments. This not only strengthens overall security posture but also fosters business unit agility and innovation through the responsible use of SaaS applications.




The Scope of Sensitive Data in SaaS: Wider Than You Might Think




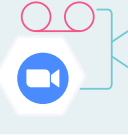



SaaS applications have transformed how organizations operate, offering features for managing, sharing, and collaborating using a wide range of data. However, this convenience comes with a critical responsibility: protecting sensitive data. Breaches of sensitive data can lead to identity theft, financial fraud, and reputational damage for both the customer and the organization. Data hosted on SaaS applications can be shared directly with external collaborators, anonymously with anyone with a link, via APIs and non-human identities (which we will examine more in the next section) or in other native ways.



Most security teams think about OneDrive, Google Drive, and Box when it comes to sharing data, but surprisingly, even platforms such as Salesforce, NetSuite, and Zoom have built in functionality to store and share sensitive data externally.



Here are some types of sensitive data processed, transmitted, and stored by SaaS applications:

Legal Documents in OneDrive 	Financial Reports in Box 	Source Code in GitHub 
Meeting Recordings in Zoom 	Confidential Messages in Slack 	Customer Transaction Data in Salesforce 
Login Credentials in Confluence/Atlassian 	Financial Documents in NetSuite 	PII (Personal Identifiable Information) in Workday 

Externally Shared Data

In SaaS applications, a data share refers to the functionality that allows you to grant access to specific files, folders, or records with other users (both internal and external). This is commonly achieved by sharing a link or granting specific user permissions within the SaaS platform.

The Rise of Personal Account Sharing



Nearly half (**46%**) of external data shares are shared with personal, non-corporate accounts such as Gmail. Two reasons for this are either sharing data with the employee's personal account so they can access it from personal devices, or that an external collaborator needs to leverage a personal account since their corporate account can't access the selected SaaS platform. This practice, however, bypasses organizational oversight and significantly increases the risk of unauthorized access. Personal accounts often have weaker security practices, may not utilize MFA or lack strong password policies. Furthermore, IT and security teams lose visibility and control over the shared data once it resides in a personal account.

Link settings

Who would you like this link to work for?

[Learn more](#)

- Anyone with the link
- People in Contoso with the link
- People with existing access
- Specific people

- Public**
Anyone can see this repository. You choose who can commit.
- Internal**
Octo Corp **enterprise members** can see this repository. You choose who can commit.
- Private**
You choose who can see and commit to this repository.

Anonymous Sharing With "Anyone With the Link"



As we detailed earlier in this report, the Google Drive misconfiguration that allowed a folder to be viewed by "anyone with the link," exposed the personal data of nearly one million individuals. But Ateam is not alone - based on the data from Valence tenants, a concerning **22%** of external data shares utilize open links, meaning anyone with the link can access the data. This essentially removes access controls and exposes sensitive information to anyone who stumbles upon the link or deliberately finds it for malicious purposes. Compounding this issue, **94%** of these open link shares are inactive. These inactive open link shares for sensitive data highlight a critical lack of proper data lifecycle management and oversight, creating a situation where sensitive information is both accessible and neglected.

Overly Permissive Access



In the case of both the ServiceNow and Salesforce misconfigurations detailed earlier in the report, misconfigurations resulting in overly permissive access controls can be a major security risk. Both cases had misconfigurations that mistakenly allowed unauthenticated guest users to access sensitive data, including passwords, bank account details, and customer PII. It's critical that when setting configuration settings in their SaaS platforms, security teams and SaaS admins minimize the data access levels as much as possible.

Non-Human Identities Outnumber Humans and Are Prime Targets

Modern SaaS environments rely heavily on automated systems and SaaS-to-SaaS integrations, each requiring non-human identities such as service accounts, third-party apps, API keys, and OAuth tokens. These integrations enhance functionality and streamline workflows, but also introduce a new layer of security complexity. While authentication tokens offer an efficient login experience for integrations, their ease of use can be a double-edged sword, as they vastly outnumber human identities. Take Google Workspace as an example. Based on our data, for every **1** human identity with access to Google Workspace, there are **8.6** non-human identities (tokens).



A stolen token can potentially grant an attacker complete access to a SaaS account, bypassing other credential requirements since the token itself functions as a single key, granting access upon verification, as the sole authentication mechanism. Furthermore, non-human identities often lack traditional security controls like multi-factor authentication (MFA), making them attractive targets for attackers.

Nearly all (**94%**) of survey respondents believe they have a dedicated process or tool in place to manage non-human identities within their SaaS applications. However, this perception of control often differs with reality. We've read about numerous examples of high-impact breaches stemming from (or at least providing one of the primary attack vectors from) compromised non-human identities. Here are a few recent cases highlighting the dangers:

1 Microsoft Midnight Blizzard Breach



This attack exploited a legacy test OAuth application to access sensitive data, and then created additional malicious OAuth applications enabling them to enter Microsoft's corporate environment.

2 Cloudflare Breach



Following an Okta breach, attackers leveraged a stolen service token and service account credentials to gain administrative access to Cloudflare's Atlassian instance.

3 CircleCI Breach



Attackers stole and abused a legitimate GitHub OAuth token that was granted to CircleCI to gain unauthorized access to GitHub tenants of CircleCI's customers.

Widespread Access, Limited Oversight

Sometimes, administrators give third-party integrations access to everything. While full access should be avoided whenever possible, there are some situations where SaaS administrators might grant broader permissions to third-party integrations:

Security Scanning Tools



Security scanning tools that analyze email for malware, phishing attempts, or data loss prevention (DLP) may require access to email content to function effectively.

Productivity Applications



Certain productivity apps, such as enterprise search tools, might require access to a wider range of data (including emails, files, and calendars) to provide comprehensive search functionality.

Data-Driven Licensing Management



Some integrations might require access to user data (such as roles or departments) to accurately assign licenses or features based on specific criteria.

It's crucial for security professionals to understand these use cases and implement security best practices to minimize the risk associated with overly permissive access. More notably, a concerning trend is emerging – the extent of over provisioning of access privileges to these integrations.

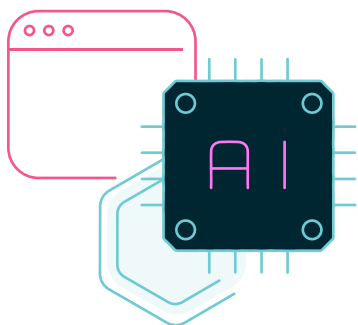
Our analysis revealed that **100%** of organizations grant full access to sensitive data (email, files, calendars, source code) to at least one of their third-party integrations. In fact, **33%** of integrations are granted sensitive data access or privileges. Considering the average organization utilizes **1,998** integrations, this highlights the potential for data exposure through compromised or misconfigured integrations.

If a third-party integration is compromised by attackers or misconfigured, the full access granted could allow them to steal or manipulate sensitive data. Most security teams don't have the ability to continuously review and right-size overly-broad permissions for their integrations that were set up incorrectly or drift over time. Organizations are granting broad access to sensitive data without a clear understanding or control over non-human identities.



The “Write” Stuff: Security Concerns in The Rise of Generative AI and SaaS Integrations

Half (**50%**) of the survey respondents indicated that governing Generative AI adoption is a top challenge.



Generative Artificial Intelligence (GenAI) tools are a growing trend. Since most GenAI tools are relatively new, they are delivered as SaaS - which means security teams need to address this increasing risk surface as part of their SaaS security program. GenAI usage can be net new applications such as ChatGPT, additions to existing applications such as Microsoft 365 Copilot or GitHub Copilot, or integrations into existing business-critical SaaS applications such as Slack or Google Workspace.

The 5 Most Common GenAI Integrations:

Here's a breakdown of the 5 most common GenAI integrations with SaaS platforms found by Valence, including a brief description of their functionality and typical access to SaaS data:

OpenAI



Generates creative text based on questions or requests - with native integrations to create suggestions based on data in SaaS apps.

Grammarly and Wordtune



Writing enhancement service that accesses all the content your employees write.

Fireflies.ai and Otter.ai



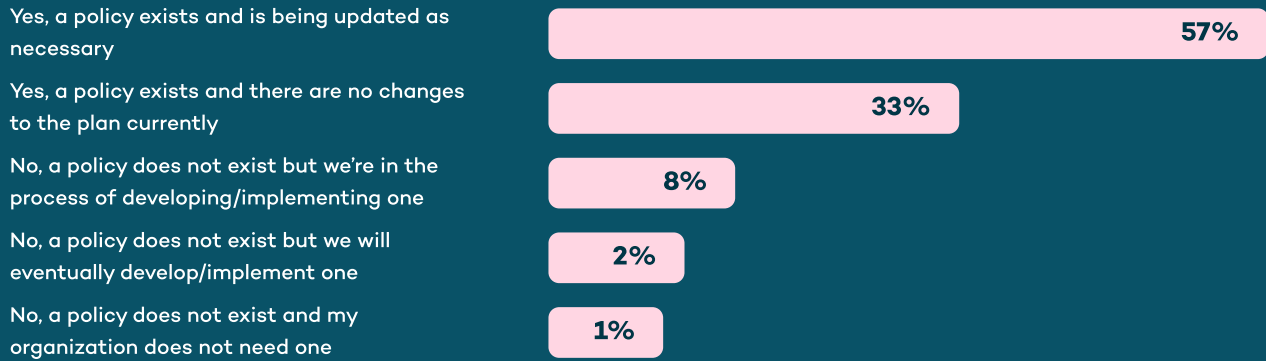
Summary and transcriptions service with access to your employees' online meetings.

GenAI adoption unlocks powerful functionality, but also introduces unique security challenges. Unlike traditional SaaS applications and integrations, GenAI tools often require access to vast amounts of data to function. This, coupled with the rapid innovation in the GenAI space, creates a complex security landscape that can leave organizations vulnerable.

Even though GenAI is relatively a new trend, our industry survey responses show that most security professionals (**90%**) have already implemented a security policy to secure and govern GenAI adoption. The responses highlight a growing awareness of the need for proactive security measures in this evolving space.



In the survey, we asked: Has your organization implemented a policy to secure and govern Generative AI adoption within the organization?



Here's why security teams need to be extra vigilant with GenAI in SaaS:

Shadow IT and Unsanctioned Use



As a growing trend, GenAI tools are often provided by lesser known companies and many business units tend to try out several tools to better understand the value they can unlock. Free trials and flashy marketing entice employees to connect GenAI tools without IT approval. These "shadow IT" applications create blind spots for security teams, making it difficult to track data exposure and enforce security policies.

Data Everywhere



GenAI is data-hungry. A seemingly innocuous feature might require access to a surprisingly wide range of data across your SaaS environment. This broad access, often by design, increases the potential attack surface and the risk of data breaches. GenAI tools often require access to sensitive data within SaaS applications, such as Zoom call recordings, sales pipeline, instant messaging, or customer data.

Privacy Perils



Many GenAI tools, especially free tools, collect user data for training purposes. Without careful scrutiny of privacy policies and data usage terms, organizations could unknowingly expose their data or even violate regulations like GDPR or CCPA. Furthermore, GenAI models might inadvertently leak sensitive data during outputs, putting confidential information at risk.

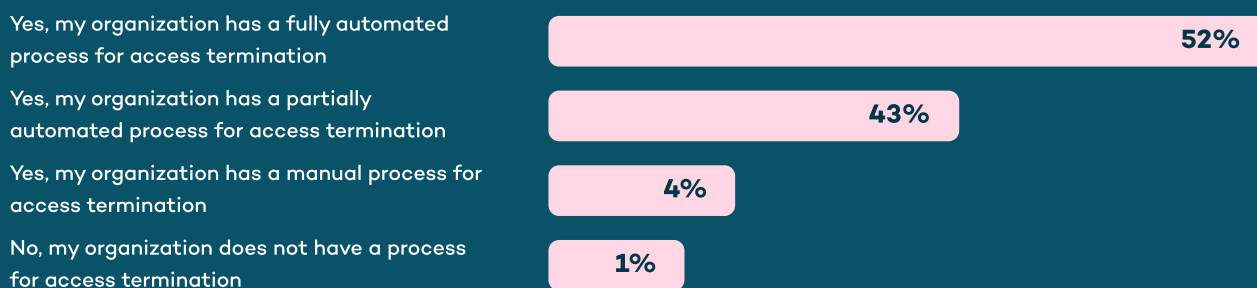
Leaving the Door Open: Ineffective Lifecycle Management and Offboarding in SaaS

The rapid adoption of SaaS applications has transformed how businesses operate. However, this shift comes with a new set of security challenges, particularly around lifecycle management and offboarding. Incomplete or inefficient offboarding practices can leave behind "dormant" external shares, user accounts and unused integrations, creating significant opportunities for attackers to exploit.

When we asked security executives if they have a process to remove unnecessary third-party API integrations as part of their offboarding process, a surprising **95%** indicated that they have a fully automated or partially automated process.



In the survey, we asked: In your organization, do you have a process to ensure third-party API access is terminated when the engagement with the third-party has ended?



But, when we look into the data from our tenants, we see a completely different reality. In platforms such as Microsoft 365, a whopping **65%** of integrations are inactive and aren't used. This means that the API key or OAuth token is valid, but not used by the application on the other end. As we've seen in breach examples, attackers have identified that these integrations are often overlooked and they played a pivotal role in breaches such as Microsoft Midnight Blizzard, Cloudflare, and CircleCI. It's safe to assume that the better you enforce lifecycle management and remove unnecessary integrations, the more secure your environment will be.



In the Cloudflare breach, the security team rotated more than 5,000 production credentials and performed in depth forensic analysis of their systems. During the credential rotation, the team missed one service token and three service accounts that were leaked during the Okta breach. These credentials weren't rotated because they were assumed to be unused. It's unfortunate, but in this case, missing 4 out of 5,000+ credentials lead to the breach - every credential counts!

A frequent cause of inactive integrations is the aftermath of failed Proofs of Concept (PoCs). Organizations may trial various SaaS integrations, granting access during the evaluation process. If the PoC fails, the integration might simply be abandoned without proper offboarding. This leaves an inactive integration with lingering access privileges, creating a potential security risk.

Stale Permissions and Neglected Access

The problem expands beyond just integrations. Valence found that **94%** of external data shares are inactive, where no external user is actively accessing the shared data and are probably not required by the business anymore. This means that users who were previously granted access still retain those permissions, even if they are no longer relevant to the current project or collaboration.

It also raises significant concerns about "stale" access permissions that could be exploited by malicious actors.

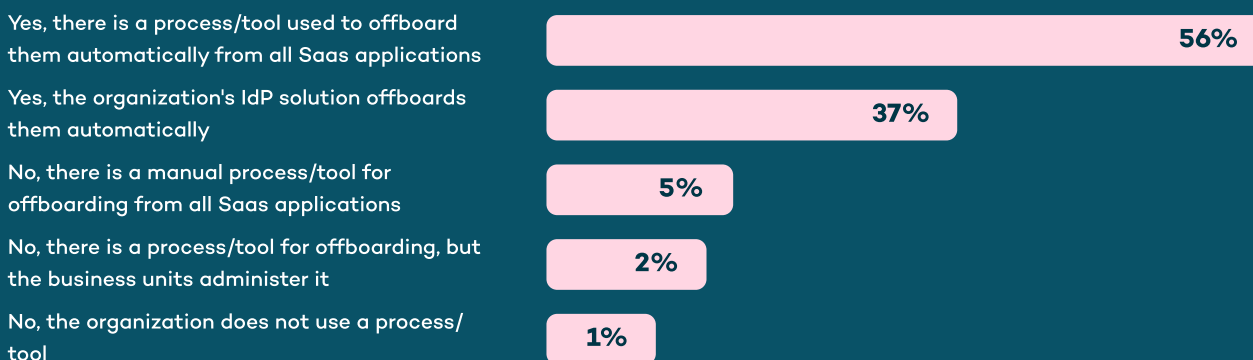
SaaS applications often encourage users to easily share externally or with an open link, or to integrate a new third-party vendor. While it's typically easy to create these shares and integrations which are part of the native user experience, the removal part isn't. A business user needs to have awareness and intent that they want to remove them in order to find the right location to revoke access. Based on the data from Valence, up to **20%** of users will do the right thing and remove unnecessary or unjustified exposure if they were properly notified about the risk and guided how to apply the fix themselves.

In many cases, security teams look for the "quick wins" to tackle first - what risk can I remove without disrupting the business? Removing dormant resources is the easiest place to start since nobody is using them. But, there is still the distributed ownership challenge since most security teams don't feel they "own" the SaaS application and therefore don't feel comfortable removing a resource or share, even if it's inactive. Typically SaaS admins, or even business users, have the business context on what they leverage the SaaS application for - which means that their input and collaboration is required to effectively remediate risks.

Timely removal of access for terminated employees and contractors is the classic use case for failed offboarding. In our survey, **93%** of the respondents indicated that they have an automated process in place to offboard ex-employees and ex-contractors. Once again, our data shows the gaps in perception and reality - in platforms such as Google Workspace, we see on average **6%** of accounts are inactive without recent logins and **4%** of those have admin privileges.



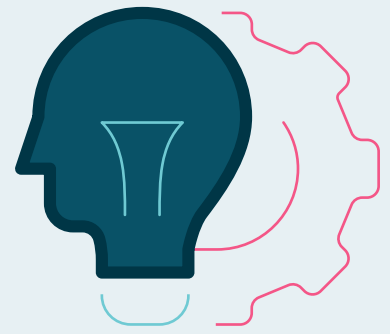
In the survey, we asked: Do the information technology or security teams have automated processes or tools to detect failed offboarding of ex-employees or ex-contractors from the business-critical SaaS?



Failed offboarding of user accounts has been exploited in high-profile breaches. The Microsoft Midnight Blizzard attack involved leveraging a non-production test tenant with weak authentication - a classic example of an unmanaged "dormant" account. Similarly, the Drizly breach stemmed from an unused GitHub account with excessive privileges granted to a departed employee. In both cases, attackers exploited these overlooked, inactive accounts to gain access to sensitive data.

Key Recommendations

Moving Forward



Establish a SaaS Security Program

1

Create Security Policies For Business-Critical SaaS Applications

Conduct a thorough review of each business-critical SaaS application's native security settings. Align configurations with industry best practices outlined by frameworks like NIST, CIS, and ISO. Continuously monitor and document these configurations (manually or automatically) to identify and address any drift.

2

Ensure Monitoring Coverage of SaaS Events for Threat Detection

Implement security tools that provide comprehensive logging and event monitoring capabilities across your SaaS landscape. Analyze SaaS application events, user activities, and admin logs to identify anomalous behavior and potential security incidents.

3

Create and Maintain an Inventory of SaaS Applications

Develop a comprehensive inventory of all SaaS applications used within your organization, including both business-critical and less frequently used applications. Prioritize security efforts based on the criticality of the application and the sensitivity of the data it stores.

4

Establish Strong Collaboration With SaaS Admins and Business Users

Foster open communication between security teams, SaaS administrators, and business users. Create clear security policies and work with the business units to ensure they properly enable the business, while keeping the data secure.

Reduce the Number of Targets

5

Focus On Secure Lifecycle Management

Ensure unnecessary access of ex-employees and ex-contractors or vendors with API and service accounts access is promptly removed upon termination - leverage automated tools to enforce such removal and/or to audit gaps created in the process due to exceptions

6

Enforce Least Privilege Access

Adhere to the Principle of Least Privilege (PoLP) and grant minimal access (internal employees/external contractors/third-party integrations) based on functional needs. Regularly review and audit user access privileges to identify and remove any unnecessary permissions.

7

Implement a Comprehensive GenAI Security Policy

Develop a policy that outlines clear guidelines for selecting, integrating, and governing GenAI tools within SaaS platforms. This policy should address aspects like data access, security controls, and user activity monitoring. Make clear which data can and cannot be used for training, fine-tuning, and in prompts.

Implement Strict Controls

8

Manage Non-Humans Like Humans (If Not More Strictly)

Non-humans have inherent limitations in available security controls, therefore attackers are increasingly targeting them and you should implement strict controls for provisioning, access management, and lifecycle management at least as you do for human users.

9

Manage MFA and SSO Exceptions With Scrutiny

Attackers always look for the weakest link and they just need one account without strong authentication to break in - minimize exceptions of your MFA/SSO policies and regularly review any existing exceptions for these critical security controls.

10

Monitor Privileged Activities Closely

If all security controls fail, you need to ensure real-time monitoring of high-privilege SaaS activities, such as creation of new admin accounts, adding privileges to OAuth accounts, or granting of elevated permissions. Investigate any suspicious activity to identify potential compromise.

About Valence Security

Valence is the first SaaS security company to combine SSPM and advanced remediation with business user collaboration to find and fix SaaS security risks. SaaS applications are becoming decentrally managed and more complex, which is introducing misconfiguration, identity, data, and SaaS-to-SaaS integration risks. The Valence SaaS Security Platform provides visibility and remediation capabilities for business-critical SaaS applications such as Microsoft 365, Google Workspace, Salesforce, GitHub and Slack. With Valence, security teams can empower their business to securely adopt SaaS.

Gain Visibility Into Your SaaS Security Risk Posture

Get a detailed assessment of your SaaS security posture with detailed recommendations for remediating risks for one of your core SaaS platforms like Microsoft 365, Google Workspace, or Salesforce.

[Get Assessment](#)

