# Implementing Zero Trust in Salesforce

# Contents

# Introduction

In the early 2000s, there was a rise in documents being stolen and leaked by employees within organizations, as well as state-sponsored adversaries gaining deep access to networks (a.k.a. Advanced Persistent Threat). In Eoghan Casey's 2006 paper, "Investigating Sophisticated Security Breaches," he described the difficulties of dealing with sophisticated adversaries. Traditional perimeter-based security models are ineffective against insiders with legitimate access and APT adversaries that blend in with normal activities, making them harder to find and eject.

In response to these evolving threats, a zero trust approach to security was introduced- one that did not trust everyone "inside" the network by default. The zero trust framework was developed to protect digital assets against malicious insiders and targeted attacks.

Recently, there has also been an increase in mobile devices operating outside the traditional network perimeter and a growth in cloud-based services.

These trends required a shift in security to embrace deperimeterization, assume compromise, and verify instead of trust.

> This is especially important as we move to a cloud-enabled technology environment where much of the data sits outside of our traditional data centers."

JOHN KINDERVAG
"NO MORE CHEWY CENTERS" FORESTER REPORT

# Threats — And Zero Trust — Move to The Cloud

Organizations are relying more on Software as a Service (SaaS) platforms for mission-critical functions, including financial services, healthcare, e-commerce, and IT operations. As more organizations move mission-critical data into SaaS platforms, zero trust principles must be adapted to these new environments. The **"Zero Trust Architecture Buyer's Guide"**, published by the General Services Administration highlights that applications and workloads in cloud environments and associated data and backups are pillars of zero trust.

> Data includes all structured and unstructured files and fragments that reside or have resided in federal systems, devices, networks, applications, databases, infrastructure, and backups (including on-premises and off-premises) as well as the associated metadata. Organizations must understand where their sensitive data exists. Then they can establish Zero Trust controls to block sensitive data from being accessed and exfiltrated."
>
> ZERO TRUST ARCHITECTURE BUYER'S GUIDE

To date, most problems impacting SaaS data have been caused by internal incidents, such as an employee exfiltrating data before leaving to work for a competitor. However, cybercriminals invariably follow valuable data, and SaaS-conscious attacks are rising. **Salesforce** is currently the largest SaaS provider, making it a suitable example to illustrate how to implement the zero trust framework in such environments, although the same challenges apply to **ServiceNow** and other SaaS platforms.

# Applying Zero Trust to Salesforce

Very few Security and Risk (S&R) professionals have expertise in SaaS data protection and therefore keep it at arm's length, leaving the day-to-day management of these systems to administrators whose primary responsibility is not security. As a result, many organizations have applied a traditional perimeter-based security model to SaaS environments such as Salesforce. Enforcing multi-factor authentication (MFA), IP restrictions, and other mechanisms such as SSPM (SaaS security posture management) and CASB (cloud access security broker) to protect and inspect a Salesforce organization at its ingress/egress points are necessary but not sufficient for SaaS data protection. This "M&M approach" to security — a hard shell and soft center — does not adequately address the greatest risks to data stored in Salesforce.

Zero trust is a data and identity-centric model, and organizations face challenges applying these security models to Salesforce. Last year, Salesforce made headlines for unsecured features in Digital Experiences that allowed external unauthenticated accounts to access more data than was intended. ServiceNow was also in the news last year for a similar issue that permitted public access to data. In addition, Salesforce applications make extensive use of integration accounts, which typically have broad access and permissions within the environment, which can cause havoc when an inadvertent or malicious problem occurs.

Protecting against such vulnerabilities involves looking for permissions and access unknowingly granted to unauthenticated accounts, including the guest user profile, and remediating these data leakage exposures. Zero trust relies on automation to maintain governance, security, and visibility of data stored in Salesforce environments. Therefore, it's essential to employ a tool that routinely runs within Salesforce and automatically alerts administrators to the overexposure of data. It is also advisable to offload any inactive/deprecated data routinely and automatically to a secure archive to reduce the amount of data accessible in production environments.

The zero trust framework can conflict with requirements in development and training environments that require less strict security controls. Therefore, it is necessary to take extra precautions to prevent sensitive data from being exposed, such as anonymizing data before it is used for development, testing, and training. This includes training artificial intelligence (AI) algorithms, which requires curated historical data and sometimes anonymizing certain elements.

# Implementing a Least Privilege Strategy in Salesforce

The zero trust principle of least privilege access is easier said than done in SaaS environments. SaaS solutions such as Salesforce that use role-based security make it easy to over-assign privileges. For instance, Profiles in Salesforce are designed to implement role-based security but often are assigned or cloned for convenience, making it difficult to untangle which accounts have more privileges than they require to perform their job/function.

This is particularly problematic when a default System Administrator Profile is involved. For example, when administrators are assigned a default Profile for their role within Salesforce, they are often assigned permissions that are out of scope for their job function, such as “Author Apex” despite not doing internal development. To prevent such problems, Salesforce recommends **granting access and privileges through permission sets**.

One of the most common problems in Salesforce is that too many users are given the ability to export reports and exfiltrate data in other ways. Users can view records over a longer time period, effectively using a low and slow approach to avoid detection. Such subtle misuse of privileges in Salesforce is harder to catch.

## Practitioner Insight

From my experience, we most commonly see customers in Salesforce with over-privileged users. While we typically focus on the common high-risk permissions such as reporting permissions, export permissions, view all and modify all data, and API permissions, some permissions are less of a focus, but that can cause incidents of unauthorized access and significant data loss. One example I recall actually stemmed from a user being granted the "Manage Profiles and Permission Sets." This permission alone was not the issue, but in this case, the user, during a post-deployment testing session, granted other users in the system elevated access and forgot to change their access back, ultimately leading to one of those users causing a data leak. This example illustrates that even granting administrative access that does not directly relate to data exfiltration can have big impacts. Understanding who is granted what privileges and the effects of that access is critical. To avoid such situations, permissions can be assigned temporarily, limited to a specific session as needed.

SABRINA SIMEROTH
SR. MANAGER OF SECURE SERVICES AT OWN

Changes to access and permissions in a default Profile can cause some users to no longer have access to the data they require and potentially break certain functionalities. Because of this fragility, Salesforce administrators may assign permissions to managers for user management and troubleshooting convenience within their teams. The risk of this approach is that it allows all managers to create a Permission Set, assign it to themselves, and, in turn, grant whatever privileges they want.

Salesforce uses its own coding language called Apex to customize business logic and automated processes. Apex generally runs in a system context, so any permissions, field-level security, and sharing rules are not considered during code execution. This might put applications at risk of inadvertently exposing sensitive data. Internal developers often don't adhere to **Salesforce Secure Coding Guidelines**, e.g., securing Sharing declaration in the statement, manually enforcing CRUD (Create, Read, Update, and Delete) and Field Permissions. Apex code could allow privilege escalation, i.e., someone gaining admin-level or privileged access without being an admin or privileged user.

It is advisable to have tooling that scans code and can detect this type of vulnerability and subsequently explain why it is a risky configuration. Therefore, strictly enforcing access control requires scanning custom Salesforce Apex code for common vulnerabilities that expose sensitive data to unauthorized users.

Data classification forms the foundation for every security framework, identifying sensitive data that requires special treatment from a protection or regulatory standpoint. The ability to restrict what digital assets a user has permission to access implies that the assets have already been identified and labeled based on sensitivity levels.

**Data classification in Salesforce** can be cumbersome and time-consuming without a methodical approach facilitated by automation. As a result of the time and effort, many organizations do not perform proper data classification and, therefore, have a generally weak SaaS data security posture. The harsh reality is that organizations that do not classify data in Salesforce lack a clear understanding of what digital assets need to be restricted, encrypted, archived, etc.

The risks of privilege over-assignment range from risky modifications of Salesforce environments that cause operational disruption and data loss to people accessing sensitive data. Proper use of role-based security starts with a Profile granting the minimum necessary access and permissions and then adding specific privileges as needed. Privilege assignment in Salesforce is not a once-and-done operation but involves active management as the environment evolves.

# Inspecting Internal Effects and Activity Anomalies within Salesforce

The zero trust principle of inspecting and logging all traffic has a network-centric perspective and does not provide comprehensive visibility into SaaS environments. In addition, the basic Salesforce offering is extremely limited when it comes to log monitoring. Salesforce customers must pay extra for Shield Event Monitoring which produces voluminous and varied logs, including information about user activities, API calls, permission changes, and automation. Analyzing Salesforce logs for high-risk activities can be like looking for a needle in a digital haystack and can create alert fatigue.

To detect problems efficiently and effectively, it is important to implement SaaS specific alerting mechanisms and to learn from historical logs to detect deviations from expected activities (see SaaS activity monitoring and anomaly detection). Enhanced Transaction Security is a feature included with Salesforce Event Monitoring that can be configured with specific policy rules that trigger a response when violated, either blocking, alerting, or requiring MFA. In addition, it is advisable to focus on the digital assets that are of the highest value and sensitivity, identified during data classification. Specifically, SaaS activity monitoring can concentrate on the components containing sensitive information that are in use and widely accessible to focus monitoring efforts, referred to as Objects That Should Be Monitored (OTSBM).

It is important to realize that SaaS logs capture specific actions and have visibility gaps. A single action can result in multiple effects within a SaaS environment, including modifications to configuration, data, and metadata. To observe these effects, it is necessary to inspect and log all internal events within the environment, some of which can be under the radar or automated. To gain more comprehensive visibility into potential problems, inspecting the effects on the information within the system is also necessary. In practice, performing this process is nontrivial because there are multiple methods for inspecting the internal effects of activities in Salesforce:

- Routinely scan for newly assigned high risk permissions, including overly broad access to sensitive data, bulk hard delete of data, encryption key management, and authoring of Apex code.
- Review Profiles and Permission Sets to see whether they can access and modify more data than necessary, particularly sensitive data.
- Review sensitive data to see whether they can be accessed or modified by accounts that should not have such privileges.
- Use change tracking on high-risk fields, either due to their sensitive nature or their accessibility to high-risk accounts, including external unauthorized access external users have enabled. This can provide a useful audit trail of activities performed by both internal and external users in the event of an incident or as a periodic check on system integrity.
- Proactively detect changes in SaaS data for significant deletions, corruption, and other modifications.

# Supplement Zero Trust with Response and Recovery

The zero trust framework focuses on the Protect and Detect functions in the overarching NIST Cyber Security Framework (CSF). It is also important to be prepared for handling incidents when (not if) they happen. A comprehensive security strategy also requires that attention be paid to the Respond and Recover functions. Routine backups of SaaS data are the last line of defense against damaging accidents and attacks, provided they are created prior to the problem and are themselves segregated and secured from deletion or corruption. Effective response and recovery requires a combination of people, processes, and technical solutions, which extends beyond zero trust and is for another discussion.

## Conclusion

Implementing zero trust in Salesforce is challenging and requires specialized knowledge, active management and monitoring, supported by fit-for-purpose technical solutions.

Learn how Own can help you achieve and demonstrate Zero Trust readiness for your Salesforce orgs, as well as help you quickly recover should a data incident occur.

Learn more →

Own

OWN YOUR OWN DATA

## About Own

Own is the leading data platform trusted by thousands of organizations to protect and activate SaaS data to transform their businesses. Own empowers customers to ensure the availability, security and compliance of mission-critical data, while unlocking new ways to gain deeper insights faster. By partnering with some of the world's largest SaaS ecosystems such as Salesforce, ServiceNow and Microsoft Dynamics 365, Own enables customers around the world to truly own the data that powers their business.

It's their platform. It's your data. Own it.

Learn more at owndata.com