Secure 3rd party software integration

Veikko Markkanen and Tapio Frantti*
University of Jyväskylä, Jyväskylä, Finland
musa.jallow@huld.io, tapio.k.frantti@jyu.fi
*corresponding author

Abstract— Organisations are seeking ways to centralize their systems and operations. The need for cloud-based Customer Relationship Management (CRM) systems is, without a doubt, increasing due to many benefits like security by design and cost efficiency. However, integrations between CRM systems and third-party software may create vulnerabilities.

This research finds out whether organisations and developers are considering security while creating integrations. The research includes a case study where organisations and developers were asked about their integration security expertise and where they think the responsibility of secure integrations lies. The research aimed to provide security best practices for integration and insight into sharing responsibilities between stakeholders.

Research showed that the size of the organisation and the developer's experience correlate with their security knowledge. However, results also showed that organisations and developers do not focus on integration security as much as needed. This research recognised a need for further research.

Keywords- vulnerabilities; integrations; secure development; security responsibilities

1. Introduction

Digital transformation is one of the key aspects to providing success to modern organisations. Connecting different platforms such as social media, cloud technologies, and data analytics is something organisations are trying to achieve. These days all organisations are facing the need for digitalisation of existing and new business models to stay competitive [1]. Different cloud-based solutions and platforms are often the solutions organisations are seeking. Digitalisation provides optimization, better data management, and a better customer experience. Rot and Sobinska [1] Mydyti et al. [2], and Coltman [3] all highlight that digitalisation is one of the key features for organisations to reach higher business impact, to stay competitive and provide customers experience customers are demanding.

Customer Relationship Management (CRM) has an ever-growing role in the modern business world. CRM provides a modern framework for a better customer experience. It also helps organisations to include customer perspectives better into their business model [3]. Alongside CRM, knowledge management comes up when considering the key success factors of modern organisations [1]. Knowledge management needs digital technologies to be effective and to provide good results [1]. When using CRM organisations can get 360 views of their business. When conducting all the customer and business-related data under one platform, the data is more achievable and, therefore, easier to use [1], [3].

One such CRM platform is Salesforce. Salesforce is the cloud-based CRMs that provides cloud services and application development. It provides different subplatforms, such as the Platform as A Service (PaaS). A benefit of Salesforce is that it is cloud-based and provides demand service, offers in-built facilities, allows you to access it from anywhere, is cost efficient to use and maintain, and is secure [4], [5].

Even though Salesforce can tackle most of the organisation's needs, there is also a need for other systems, applications, and services for organisations to use [6]. Patel and Chouhan [7] bring up the need for integrations from Salesforce to third parties to use collected information Salesforce has. For this purpose, Salesforce provides a comprehensive application programming interface (API) to create integrations between Salesforce and other systems, services, or applications. A problem in this kind of integration relies on the validation of third-party software.

What happens if integration between Salesforce and 3rd party software is made, but 3rd party software is breached? Who is in charge of ensuring this type of situation cannot happen? Soni and Vala [8] state that application providers are responsible for application security. This kind of thinking can be seen as "normal" or a common way of shifting responsibility for security to the 3rd party. On the other hand, Seify [9] brings up that data security level and security policy depends on the security policy of the organisation. Every organisation using CRM should have some sort of CRM risk management [9]. However, sometimes an organisation does not have enough security knowledge and they trust that CRM is secure itself, so there is no need to validate the 3rd parties, and/or they shift security to those in charge of developing their CRM. Therefore, there might be a significant problem if a 3rd party operator has malicious intentions or is not following security standards and precautions. So, the question arises, can we rely only on 3rd party operators in matters of security? Should we look beyond that and shift more responsibility to those who are creating integrations or demanding them?

Understanding how to create as secure as possible integrations between different systems is essential. The need for security arises mainly when the system has a large amount of sensitive data of customers and businesses. In the modern world, where systems and data are becoming a standard for doing and managing organisations at all sectors and levels, we need to

2767-6684/23/$31.00 ©2023 IEEE
DOI 10.1109/DSA59317.2023.00038

find proper precautions, methods, and standards to move data from one system to another. This research will focus on Salesforce, but all the aspects of this research can also be used in other data-centralised digital platforms that are integrated with other systems, applications, and services. We try to answer to questions "*How organisations ensure security of integrations?*" and "*Who is in charge if third-party software is breached and the system is compromised?*" We can continue further and set the questions as: 1. *Is there mismatch between organisations and developers on where they see that responsibility lies on?* 2. *How does Salesforce developers' experience affect their ability to take responsibility for integration security?* 3. *How do an organisation's size and resources in use affect its ability to take responsibility for integration security?*

Even though this research aims to answer some of the questions regarding integration security and related responsibilities, we recognised the need for further research due to the lack of existing research and the shortcomings of this research. Also, researching integration security as a bigger topic would be needed for all the developers worldwide.

2. Literature review

Due to digitalisation, customer behaviour is changing rapidly. A significant player in this change has been social media. When individuals spend more time on social media and the Internet, their expectations for the services and goods they consume change. Organisations must participate in this development and move towards digital and customer-centric solutions. If one organisation is not working in line with customer's standards, the Internet makes it easy to find organisations. These competitors can come from anywhere in the world; therefore, because of digitalisation, organisation's need to start seeing all similar organisations as their potential competitors. [10].

2.1. Customer Relationship Management

Customer Relationship Management (CRM) often comes up when discussing modern organisation digital transformation. Coltman [3] sees CRM as a platform that is an embedded strategic tool. CRM brings all the modern aspects of the customer-, operational- and data management into the use of organisation. In the past, CRM focused more on using technologies and software to enable a good customer experience and relationship with the customers. These days CRM is seen to be more about the experience, which means utilising engagement with the customers to provide the most value. This can mean engaging existing and new customers or creating interactive and fun customer experiences. Creating a forever-lasting partnership with each customer is something organisations are aiming to achieve. [11].

Williams [11] states that having a solid customer strategy is a significant part of developing a CRM framework. Because CRM is more than a technical tool or platform, organisations need a comprehensive strategy on how and to what extent they will implement customer relationship management into their organisation. Customer strategy includes customer portfolio management, segmentation, and segment strategy. Customer portfolio management quantifies the value customers bring to the organisation, the scale of investments to each customer, and steps to achieve set goals. Segmentation means understanding customers, their needs, and expectations for the organization's goods and services. Segment strategy, therefore, means how the organisation executes accordingly aspects that were brought into the light. [11].

CRM is something that organisations need to take care of. CRM provides necessary data and metrics for the decision makers to refine business models and get a higher business impact for their business decisions [3]. In this research, we are looking into one of the leading CRM platforms, Salesforce.

Spremić and Šimunic [12] state that CRM data security should not only be the responsibility of IT sector in an enterprise, but rather, the business infrastructure should be defined in a manner that integrates data security as a default mechanism. He and Zhang [13] emphasize that to increase cyber security in enterprises and within their CRM systems, it is necessary to implement awareness programs for employees and minimize security fatigue. Aldawood and Skinner [14] note also for increasing employee awareness regarding data security social engineering. Aboelfotoh and Hikal [15] state that enterprises have to implement data security systems for detection, analysis, and defence of threats. They also emphasize revival from the cyber attacks. Wang and Wang [16] point out platform vulnerabilities and risks of third party data control. Soltani and Jafari [17] consider cyber security of the integrated systems, such as CRM from the information gathering, analysis, evaluation, risks, recovery, and maintenance point of view. Bakator [18] presents a model that provides an overview on how CRM data security challenges can be addressed. However, all these papers consider improvement of CMR through data security. None of them consider third party system integration to CRM as a whole.

Stewart [19] shows that security best practices during application development to CRM systems have a significant influence on applications created in regulated environments. The paper assumes implicitly that application developers take major responsibility in application development. We note that the applied best practices need all the integration participants views and acceptance leading to the shared security responsibilities.

Backes [20] consider inclusion of apps with third-party libraries on Android. Third-party libraries have been shown to be security and privacy hazards by adding security vulnerabilities to their host apps or by misusing inherited access rights. The paper proposes a library detection technique that is resilient against common code obfuscations and is capable of pinpointing the exact library version used in apps. The authors approach software integration from zero-trust point of view. The problem area different the one we are considering but it is analogous to third-party application integration to CRM platforms.

Kham [21] note that majority of agile software development security challenges reported in literature, occurred due to lack of involvement of security expert. For our research case this means that security challenges are mainly in the development side.

Neha [22] writes that Salesforce, like most cloud service providers, uses the shared responsibility data security model, which means that while Salesforce is responsible for maintaining the security of the cloud, organizations using Salesforce are responsible for maintaining the safety of data in the cloud. Neha [22] also note that APIs are one of the most common channels for attacks on data. Therefore, while Salesforce makes life easy for its users through the wide-range integration of apps, it also leaves users' data vulnerable. Regulating APIs is the best way to ensure that they do not pose a threat. This can be done by keeping regular tabs on the activities of the API through auditing, creating an integration user for every API that accesses your data, setting strict rules for their access permissions, and using the whitelisting feature to determine which apps can integrate with your system. This share security challenges in third party apps integration cases to the Salesforce as an API designer and maintainer and third party apps developers.

2.2. Security development

Because we do not have a lot of literature about integration security, we need to look more into software security. There is much more literature and standards about software development, and because integrations belong under software development, we chose to take this approach.

Software security is becoming increasingly recognised and organisations and developers are starting to act towards securing information and software. Common attack vectors related to the software are Cross-Site Scripting (XSS), Structured Query Language (SQL) injection, and buffer overflow exploitation. In these cases, information is often compromised, and the organisation and its data integrity are violated. Even today, these attacks are still approached mainly by responding and reacting to the attack. In Secure Development Lifecycle (SDL), vulnerabilities are software security flaws that an attacker has been able to manipulate. [23].

Security of integrations cannot be only about reacting and responding to the attack; more than that, it needs to be considered at every stage of integration creation. Let's take a look at SDL, secure software development framework (SSDF), Building Security In Maturity Model (BSIMM), and OWASP Software Assurance Maturity Model (SAMM) to understand how security vulnerabilities could be managed during the creation of integration.

Creating secure integrations should be the intent of all parties. SDL is a set of practices that help to reduce the number of vulnerabilities in software. SDL strongly relates to the generic software development lifecycle (SDLC). SDL principles were initially created by Microsoft (MS SDL). Microsoft [24] describes twelve practices developers should follow. However, we have started seeing new approaches and models to SDL over the years. SDL follows a development framework and extends security as a part of all the different stages in an SDLC. [23].

A secure software development framework (SSDF) is a collection of secure software development documentation from organisations like BSA OWASP and SAFEcode. SSDF version 1.1 was released in early 2022, and NIST has plans to improve and evolve it further. SSDF follows in the same footsteps as SDLC with few addons; therefore, it should be integrated with SDLC implementations. The aim of SSDF is to reduce vulnerabilities in the developed software. SSDF is also seen to help security management and -communication because it provides a common language for describing secure software development practices. [25].

Building Security in Maturity Model (BSIMM) contains 12 practices organized into four domains [23]. These 12 practices are strategy and metrics, compliance and policy, training, attack models, security features and design, standards and requirements, architecture analysis, code review, security testing, penetration testing, software environment and configuration, and vulnerability management. BSIMM [26] report brings up security tendencies related to secure software development. The report includes activities and trends from 128 organisations.

OWASP [27] software assurance maturity model (SAMM) provides a way to analyse and improve SDL. OWASP SAMM is an open framework, and it is an evolutive and risk-driven model. The model contains five business functions which are each split into three practices. The first business function is Governance, and it has the following security practices; strategy and metrics, policy and compliance, education, and guidance. Next up is Design, this business function contains threat assessment, security requirements, and security architecture. The third business function is Implementation which splits into secure build, secure deployment, and defect management. Following that comes Verification. Verification business functions subcategories are architecture assessment, requirements-driven testing, and security testing. Lastly is Operations, which includes incident management, environment management, and operation management. [27].

The strength of OWASP SAMM lies in its comprehensiveness. The model also takes a stand on the different responsibilities of different actors. Governance is more an organisations responsibility to implement and share. The responsibility of Design also lies in organisation. OWASP [27] states that under a design, security requirements bring up how to ensure that third-party or supplier needs to be evaluated. This also includes that in the agreements between organisation and third-party organisational requirements for security needs to be present.

Implementation highlights the responsibility of developers. This part contains practices about documentation, development patterns, development process, and metrics and tracking. Verification is all about assessment and testing [27]. Testing should be done by developers, organisations, and external sources, and it should be done at all levels and stages of the development process. Operations is described as a joint effort, but the responsibility of it lies in organisation. How to manage incidents and which kind of incident response processes there is in place are questions organisations need to have the answer to. Developers need to continuously patch and update applications whenever vulnerability arises. On an operational level the documents and policies concerning security and data protection need to be kept up to date. The responsibility of these organisation-wide documents lies in organisation and they need to follow current laws and standards. [27].

3. Integrations in Salesforce

Force.com provides a wide variety of tools and concepts to create integrations. For example, you can create integrations using code or pre-created AppExchange packages. At the AppExchange, there is a section for integrations. This section has 200 different integration apps. For example, Patel and Chouhan [7] found an existing package containing integration between Salesforce and Twitter from AppExchange, which they used in their research.

Salesforce integrations can be split into architecture, capability, and pattern types. Salesforce has integration architecture which consists of three different types. Point-to-point integration means a one-to-one relationship with Salesforce and another system where those two systems communicate through messages. Hub-and-spoke integration means an integration type where a centralised hub system is in charge of communication between systems. Hub in this kind of integration is in charge of routing traffic between systems. The third integration type is enterprise service bus integration (ESB). ESB is the next generation of hub-and-spoke integration. ESB, like hub-and-spoke, has a centralised connector in charge of routing traffic between systems. The difference is that in ESB, this centralized connector is an integration engine that can be used to create these connections between systems. A significant benefit of ESB is that it offers capabilities to improve integration security. With ESB, for example, one could create authentication and authorization inside the integration. [28].

4. Security of Integrations

This research aims to show how security responsibilities are split between stakeholders regarding Salesforce integrations. Integration's role for organisations is an essential part of modern information technology infrastructures. Organisations are often forced to use outsourced developers due to the high demand for developers or the cost-efficiency it can provide. The importance of this research comes from the lack of literature on integration security. There can be found publications on security aspects and related to integrations, but not publications that cover them both in the same context. This creates a need for further research to give tools for organisations and developers to ensure the security of integrations. Another thing is that because Salesforce is seen as secure itself, Salesforce developers and organisations might be relying on Salesforce for integration security. Salesforce indeed provides a different set of capabilities to ensure information security also in integrations. However, many attack vectors and vulnerabilities still need to be taken care of. Common attack vectors related to the software are compromised or weak credentials, insider threats, missing or weak encryption, misconfigurations, back door vulnerabilities, Cross-Site Scripting (XSS), Structured Query Language (SQL) injection, buffer overflow exploitation, and third and fourth party vendors.

This research tries to find correlations between organisations' size and security knowledge. Salesforce survey conducted for developers aims to find correlations between experience in years and security knowledge. Findings from the developer and organisation surveys are then examined to determine whether there is a mismatch between responsibilities. Salesforce survey for Mulesoft developers mainly highlights good practices integration frameworks can offer.

The main goal of this research is to highlight best practices for the organisations and for the Salesforce developers to cover the security aspect of integrations better. The research contains empirical case study which focuses three analyses of three different surveys with different focus groups. These focus groups are Salesforce developers, organisations, and Mulesoft developers. The survey for Mulesoft developers provides more insight into one of the world’s leading integration platforms and the security implications we can learn from them.

The case study is created for this research because it looks into the phenomenon within its real-life context [29]. In this research, it means that the phenomenon this research investigates is integration security, and the real-life context is how integration security is ensured in Salesforce integrations. This research can be seen as using both embedded and holistic approaches. Organisations are examined with an embedded approach because research explores different organisations using Salesforce. Salesforce developers and Mulesoft developers can be examined with a holistic approach. This can be seen in a way that each developer works with Salesforce and creates integrations in a Salesforce context. The only way differencing developers is their background and how they are creating integrations. An empirical case study works well for this research because it typically uses surveys to describe and explain the phenomenon. Typically, these surveys are conducted quantitatively or numerically. This research uses mainly numerical questions to make responding to the survey as fast as possible. Results are then described in a quantitative research manner. Although it is essential to point out that this study can be called more “context specific”, it speaks more directly to industry needs rather than uses formal methods [29]. Bass et al. [29] bring up that, like in this research, a small sample of the organisations and developers can create biased results. With a small sample, it is also hard to generalise findings, so the descriptive approach is justified.

5. Implementation of the survey

All the surveys were structured in a way that they were fast and easy to answer. All background questions were multiple choice questions. Integration and security questions were primarily using rating scale questions, except in Salesforce survey for organisations there were also some open-ended questions. Responsibility question was made using matrix and best practice question with multiple choice. Multiple choice questions had scale from 1 to 6 where 1 = Strongly disagree, 2 = Disagree, 3 = Somewhat disagree, 4 = Somewhat agree, 5 = Agree, and 6 = Strongly agree.

An idea for survey creation was to get three types of information; background information, integration and security knowledge and practices, and a view of how responsibilities should be distributed. The question about responsibility distribution and best practices is the same in all surveys. The question about responsibilities goes as follows: "When creating integrations, there are different stakeholders involved in the process. Rate each stakeholder's role when making integration secure. (1 = Main responsibility, 2 = Some responsibility, 3 = Little responsibility, 4 = No responsibility)". Question about best practices forces respondents to choose the three most valuable ways to ensure Information Security when creating integrations. These options are presented in Table 1.

Table 1. Options of the best practices in the surveys.

Using solutions found from AppExchange
Creating own set of validations
Creating support ticket to Salesforce
Asking about the security from a third-party software provider
Auditing security of third-party software
Extensive testing
Searching best practices
Encourage organisations using Salesforce to take care of it
Monitoring integrations
Option to describe other than any of the following

Salesforce survey for developers consist of background questions and integration and security questions. Background questions consist of the following parts: type of employer, Job title, Salesforce experience, information technology experience, number of integrations within Salesforce and outside of Salesforce context person has created and what percentage of Salesforce-related integrations have been made using integration platform, such as Mulesoft.

Integration and security questions consist of the following parts: importance of integrations, integration security in Salesforce, knowledge about integration security threats, and how to ensure security of integrations.

Salesforce survey for organisations also consist of the background questions and integration and security questions. Background questions include: size of organisation, Salesforce experience in years, and amount of Salesforce users and in-house developers. Integration and security questions were divided into three categories: Information Security Policy (ISP), information security responsibilities in the organisation and integration security. ISP questions were as follows; the existence of ISP and how integration security is mentioned. Question-related to information security responsible in the organisation was split into two questions; Is there such personnel, and what is that individual's role in ensuring the security of integrations? The last three questions in this section related to integration security were about whether organisations do integrations in-house, using partners, or both, best practices used to ensure integration security, and how the organisation ensures that partners follow security precautions while creating integrations.

Salesforce survey for Mulesoft developers consists of similar questions than to Salesforce survey for developers, with the difference that questions were related to the Mulesoft context rather than the Salesforce integration context. In the end, there was also one extra question: were Mulesoft developers a chance to describe the benefits Mulesoft provides related to information security?

6. Results and discussion

6.1. How organisations ensure security of integrations?

Almost all organisations have some ISP. One smaller and one medium organisation did not have such a policy. Large organisations were the only group where all organisations had information on some ISP. Large organisations also more likely than not have chief information officer and other formal security manager roles (CISO, CIO, Security manager). In small and medium size organisations, a person in charge of information security seems to have multiple roles where security is one of many. Study shows that the size of the organisation clearly affected how clear and formal the security role organisation has. Smaller organisations have more generic security roles where the responsibility of security is whether one of the roles or, in some cases, it is one of the tasks the person has alongside other tasks. In larger organisations, an assigned person is in charge of security, and their primary role is to ensure information security as a whole. Job titles of such persons are often CIO or CISO. The study shows a clear correlation between size and formality of the person in charge of security in the organisation.

All organisations with an ISP state that they are not directly covering integration security in the information security policies. Larger organisations tend to have wider ISP and seem to include aspects of integration and information security better than smaller ones. Only one small organisation covers data security and processing in their ISP.

Organisations, most importantly, need to ensure that they have an information security policy which includes integration security and that ISP is followed by everyone. One way to approach this is to include the CRM security management system (CRM-SMS) in the ISP or have it as a separated document [9]. With CRM-SMS, organisations will have a clear idea of their Salesforce infrastructure security and the security level it has. CRM-SMS should be in line with ISP, and when done properly, it will also include an integrated security management aspect in the Salesforce context.

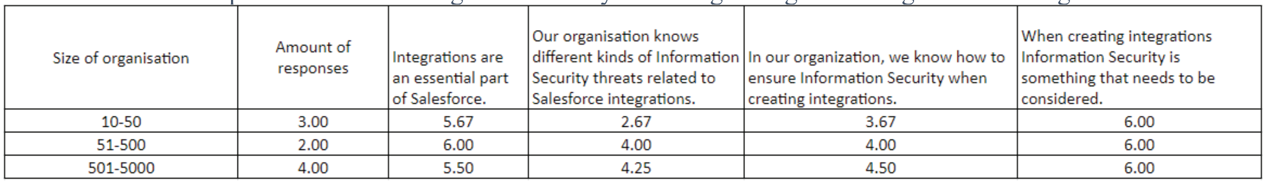
Organisations have different levels of knowledge of what comes about integration security. There is also a significant difference seen between organisations that are large in size than in smaller organisations. It also seems that there is a strong correlation between integration security knowledge and recognising different vulnerabilities integrations create. Also, the number of in-house Salesforce developers affects the organisation's integration knowledge level. Organisations of all sizes have a better understanding of how to implement information security on integration creation than they have of threats that integrations create. These should be well understood in all organisations because all the integration implementations should ensure that these threats are covered. This should be something for the organisations to include in their ISP so that it is easier to point out to developers what kind of threats need to consider.

Organisations approach the creation of integrations differently. Many organisations rely on partners, some have internal developers to implement integrations, and especially larger organisations use both. Pretty much all organisations think that integrations are an essential part of Salesforce. However, there was a slight indication that in the smaller organisations, which were more likely to use partners, the importance of integrations was higher than in larger ones.

When creating integrations, there are also laws, regulations, and standards that organisations need to follow. Organisations are responsible to follow these precautions. Organisations are also the one that is accountable to their customers and clients if the information is leaked or integrations are not created by following laws and regulations. With NDAs and other agreements, organisations can shift responsible towards partners and developers, but in case of a breach, the public opinion will most likely be that organisation is responsible. This implies that organisations need to take ownership of the integrations and ensure that everyone creating integrations to the Salesforce instance is following security best practices. The study conducted for this research shows that organisations see themselves to have the highest responsibility for making integrations secure.

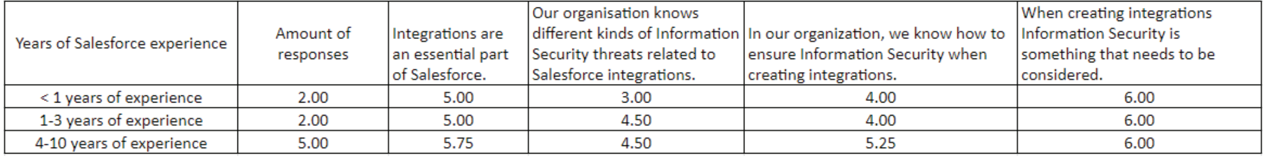
There are two tables, Table 2 and Table 3 created from the responses to the Salesforce survey for organisations. These tables represent average scores of responses. There are table for size of organisation categorisation and for Salesforce experience in years categorisation. Both tables are divided by the grouping of each categorisation.

Table 2. Responses related to integration security knowledge using size of organisation categorisation.



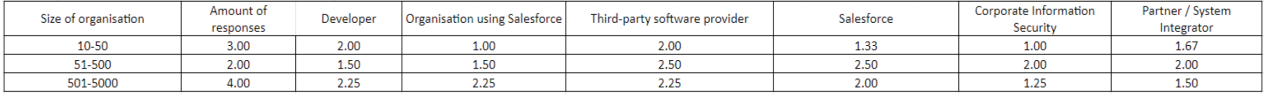
Size of organisation	Amount of responses	Integrations are an essential part of Salesforce.	Our organisation knows different kinds of Information Security threats related to Salesforce integrations.	In our organization, we know how to ensure Information Security when creating integrations.	When creating integrations Information Security is something that needs to be considered.
10-50	3.00	5.67	2.67	3.67	6.00
51-500	2.00	6.00	4.00	4.00	6.00
501-5000	4.00	5.50	4.25	4.50	6.00

Table 3. Responses related to integration security knowledge using years of Salesforce experience categorisation.



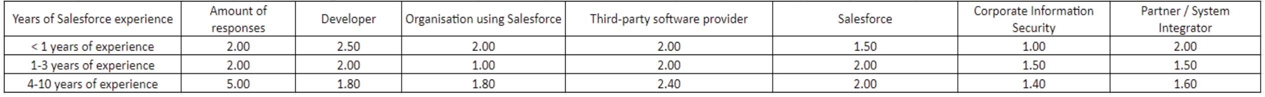
Years of Salesforce experience	Amount of responses	Integrations are an essential part of Salesforce.	Our organisation knows different kinds of Information Security threats related to Salesforce integrations.	In our organization, we know how to ensure Information Security when creating integrations.	When creating integrations Information Security is something that needs to be considered.
< 1 years of experience	2.00	5.00	3.00	4.00	6.00
1-3 years of experience	2.00	5.00	4.50	4.00	6.00
4-10 years of experience	5.00	5.75	4.50	5.25	6.00

Table 4. Responses related to responsibilities using size of organisation categorisation.



Size of organisation	Amount of responses	Developer	Organisation using Salesforce	Third-party software provider	Salesforce	Corporate Information Security	Partner / System Integrator
10-50	3.00	2.00	1.00	2.00	1.33	1.00	1.67
51-500	2.00	1.50	1.50	2.50	2.50	2.00	2.00
501-5000	4.00	2.25	2.25	2.25	2.00	1.25	1.50

Table 5. Responses related to responsibilities using years of Salesforce experience categorisation.



Years of Salesforce experience	Amount of responses	Developer	Organisation using Salesforce	Third-party software provider	Salesforce	Corporate Information Security	Partner / System Integrator
< 1 years of experience	2.00	2.50	2.00	2.00	1.50	1.00	2.00
1-3 years of experience	2.00	2.00	1.00	2.00	2.00	1.50	1.50
4-10 years of experience	5.00	1.80	1.80	2.40	2.00	1.40	1.60

6.2. How developers ensure security of integrations?

There is a clear correlation between the IT experience and the number of integrations made. However, interestingly there was not as much correlation between Salesforce experience and Salesforce integrations people have made. This shows that those who create Salesforce integrations are not often the most experienced Salesforce professionals. However, those who have created a lot of (more than 10) Salesforce integrations have, with few exceptions, at least 4 to 10 years of IT experience. Another thing to highlight is that those who have created more than 10 Salesforce integrations have created at least 11 to 30 integrations.

Like excepted, experience and amount of integrations individual have created correlate with the security knowledge. Those who have 1 to 3 years of experience somewhat disagreed that they are aware of security threats related to Salesforce integrations. They also somewhat disagreed on whether they knew how to create secure integrations.

The second group, those with an average of 4 to 10 years of experience, saw they somewhat disagreed on knowing different security threats integrations have. Although this, they were rated on the high end of the scale, whereas the group containing 1 to 3 years of experience was at the low end, almost disagreeing. 4 to 10 years of experience group thought that even though they did not understand all the different threats, they felt capable of creating secure integrations.

10 to 20 years of experience was interestingly most awakened on integration security matters. By looking into their averages, they thought they somewhat agreed by knowing the different threats integrations face. They also agreed that they know how to create secure integrations. This group contained the highest percentage of respondents that are using integration platforms. In addition, 55% have created at least 76%, and 10% approximately a fourth of the integrations using some integration platforms. This alone highlights a correlation between integration security knowledge and using integration platforms to create integrations. With more than 20 years of experience group somewhat agreed that they understand different security threats related to integrations and had a similar average on creating secure integrations.

Against presumptions, the most amount of experience did not completely correlate to the security knowledge. The group that seems to have the highest level of integration security knowledge is, in fact, the group where participants have 10 to 20 years of experience.

All groups agreed that information security needs to be considered while creating integrations. However, only the group containing participants with 1 to 3 years of experience barely agreed with the statement. Other groups, including Mulesoft developers, were closer to strongly agreeing than somewhat agreeing.

When looking at integration security knowledge with the amount of created integrations point of view, the results show a similar correlation to years of experience. Those who have created less than 5 integrations are somewhat disagreeing that they know different integration security threats and how to create secure integrations. Groups where participants have created 6 to 10 or 11 to 50 integrations somewhat agree on both statements. Those who have created more than 50 integrations and Mulesoft developers somewhat agree on knowing different integration security threats and how to create secure integrations.

Developers are in charge of creating integrations. Because developers have the knowledge and skills to create integrations, it should also be their responsibility to create integrations by following security standards and best practices. Part of being a developer is to make sure that knowledge is not outdated. Developers should follow security, especially Salesforce integration security-related publications and guidelines, to keep up with the security standards. Salesforce provides excellent tools and features which can add to the security of integrations. All developers that create Salesforce integrations should know how to implement these tools and features on the organisation's Salesforce instance. How to manage the integration of users and profiles? How to monitor integrations? What is the level of access each integration needs, and how it controls that integration has access only the data it needs to be able to? These are questions developers need to have answers to. It is essential to understand that developers need to take main responsibility for the Salesforce integrations even though there are no agreements or discussions about security from the organisation's side. Organisations and Salesforce partners need to ensure that there is proper support for the developers to create security within their integrations.

Those with more than 20 years of experience seem to rely less on auditing the security of third-party software than other groups. However, they rely more on monitoring integrations and creating their own set of validations.

Group 1 to 3 years of experience most selected choice was auditing security of third-party software, and extensive testing, encourage organisation using Salesforce to take care of it, and searching best practices were tied to second place. 4 to 10 years of experience group had auditing security of third-party software and extensive testing tied on a first place, and monitoring integrations on a third place. 10 to 20 years of experience group had auditing security of third-party software and searching for best practices in the first place, and monitoring integrations in a third place. More than 20 years of experience group had monitoring integrations first, extensive testing second, and creating its own set of validations and searching best practices on third place. With Mulesoft developers auditing security of third-party software and extensive testing got first place, and monitoring integrations and following MuleSoft's designing and building patterns were in third place.

Results show that developers are pretty much in the same place on what comes to ensuring the security of integrations. We saw few differences in the responses, but overall, the responses were in line between different groups and categorisations. Best practices for the developers include security knowledge about common threats on Salesforce integrations, design patterns to ensure secure development, and ensuring that every Salesforce integration is monitored and tested. The biggest challenge for the developers is that they usually work under a deadline, and there is not enough time for security and extensive testing.

6.3. Responsibilities

Table 4 and Table 5 show how the responses were split with each categorisation and groups. When taking a look on how organisation's see responsibilities there is clear indication that organisation's think that they themselves have the highest responsibility. By taking a look on averages how organisations rated each stakeholder we see that Corporate Information Security (CIS) was ranked having most responsible. Organisation using Salesforce and partner, or system integrator were tied to the second place. These all were ranked to have main responsibility on creating secure integrations. Fourth were Salesforce having overall ranking as main responsible at a slight margin. Developer was fifth and at the last place were third-party software provider. Developer and third-party software provider was seen to have some responsibility. It is important to point out that when considering on an overall average all stakeholders had greater score than 2.5 on a scale 1 to 4. Therefore, it is safe to say that all the stakeholders highlighted were seen to have at least some responsibility what comes to creating secure integrations.

There were some exciting results when looking at how developers see different stakeholders' responsibilities on integration security. With less experience, the individual is more likely to see that most of the different stakeholders are mainly responsible. For example, a group containing 1 to 3 years of experience thought that five out of six different stakeholders have main responsibility. In other groups, the number of stakeholders having main responsibility was four, and with Mulesoft developers, it was two.

All groups agreed that developers have the main responsibility for what comes with creating secure integrations. Fewer integrations individual has made it more likely they think that the main responsibility lies on organisations. With years of experience, those who have 1 to 3, 10 to 20, and more than 20 years of experience think that organisation has the main responsibility. Other groups, including Mulesoft developers, think that the organisation has some responsibility.

Third-party software providers' responsibility was seen as the main responsibility of only those who have created 10 or fewer integrations or 10 or fewer years of experience. Others saw that third-party software provider has some responsibility. Interestingly while other groups thought that Salesforce had some responsibility, only in the group where individuals have 1 to 3 years of experience Salesforce was seen to have the main responsibility. In this group, 86% of respondents thought that Salesforce had the main responsibility. This indicates that those with less experience think that the platform provider, in this case, Salesforce, is responsible for ensuring the security of integrations. Those who have created 10 or fewer integrations agreed that the main responsibility lies in Salesforce, even though the result was not as straightforward as within 1 to 3 years of experience group. The rest of the groups in both categorisations thought that Salesforce had some responsibility.

Corporate information security (CIS) was seen to have the main responsibility. Only Mulesoft developers thought that this was not the case, and they saw CIS as having some responsibility. Lastly, when analysing results on how responsible partner or system integration was seen to be, there was a clear indication that it is seen to have main responsible. Interestingly, only one group thought that partner or system integrator does not have the main responsibility. This group contained those who have 1 to 3 years of experience. They thought that partners or system integrators had some responsibility. These are interesting results because all but one respondent in this group worked for a Salesforce partner. The one in this group working for an organisation using Salesforce see that the partner or system integrator has the main responsibility.

When looking at the average scores of each group in experience categorisation, all groups except 1 to 3 years of experience think that developers are the most responsible stakeholder out of everyone else. A group of respondents with 1 to 3 years of experience think that Salesforce is the most responsible stakeholder. There is a clear correlation between experience and the amount of integrations individual has created and how they see the responsibility of third-party software provider and Salesforce.

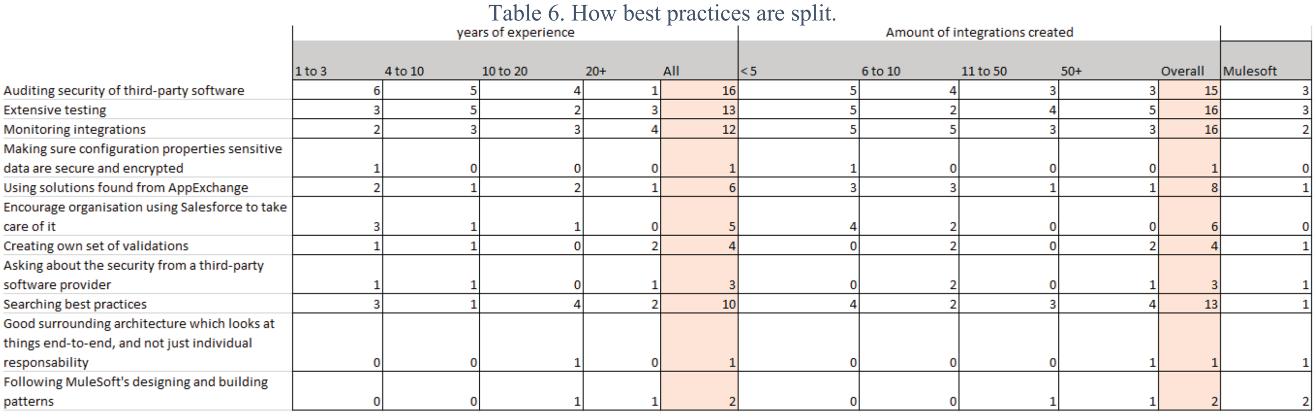
This study shows that organisations and developers see themself to be the most responsible stakeholder in making integrations secure. This is a good result because it implies that both sides of the coin are taking ownership of the security. Ilmarinen and Koskela [30] argued that security is everyone's responsibility. It does not matter whether you are an organisation's security personnel, developer, or user you need to make sure that you are not the weakest link.

One thing this research did not cover is how 3rd party software providers see their responsibility. Both organisations and developers thought that 3rd party software providers had some responsibility. It would be interesting to compare third-party software providers' thoughts on where they see the responsibility lies.

6.4. Best practices to ensure integration security

Table 6 shows how the best practice answers were split. There were 11 different answers chosen between all different groups and categorisations. Auditing third-party software security, extensive testing, monitoring integrations, and searching best practices made a clear top four on the list with both categorisations. With the different categorisations, there were almost no differences between the two. When looking at different groups, there were some interesting differences. Most interesting is that 43% of those belonging to the group where individuals have 1 to 3 years of experience, and 44% of those who have created less

Table 6. How best practices are split.



	years of experience					Amount of integrations created					
	1 to 3	4 to 10	10 to 20	20+	All	< 5	6 to 10	11 to 50	50+	Overall	Mulesoft
Auditing security of third-party software	6	5	4	1	16	5	4	3	3	15	3
Extensive testing	3	5	2	3	13	5	2	4	5	16	3
Monitoring integrations	2	3	3	4	12	5	5	3	3	16	2
Making sure configuration properties sensitive data are secure and encrypted	1	0	0	0	1	1	0	0	0	1	0
Using solutions found from AppExchange	2	1	2	1	6	3	3	1	1	8	1
Encourage organisation using Salesforce to take care of it	3	1	1	0	5	4	2	0	0	6	0
Creating own set of validations	1	1	0	2	4	0	2	0	2	4	1
Asking about the security from a third-party software provider	1	1	0	1	3	0	2	0	1	3	1
Searching best practices	3	1	4	2	10	4	2	3	4	13	1
Good surrounding architecture which looks at things end-to-end, and not just individual responsability	0	0	1	0	1	0	0	0	1	1	1
Following MuleSoft's designing and building patterns	0	0	1	1	2	0	0	1	1	2	2

than five integrations, thought that encouraging organisations using Salesforce to take care of the security was one of the three choices. Those with more than 20 years of experience seem to rely less on auditing the security of third-party software than other groups. However, they rely more on monitoring integrations and creating their own set of validations than other groups.

Group 1 to 3 years of experience most selected choice was auditing security of third-party software, and extensive testing, encourage organisation using Salesforce to take care of it, and searching best practices were tied to second place. 4 to 10 years of experience group had auditing security of third-party software and extensive testing tied on a first place, and monitoring integrations on a third place. 10 to 20 years of experience group had auditing security of third-party software and searching for best practices in the first place, and monitoring integrations in a third place. More than 20 years of experience group had monitoring integrations first, extensive testing second, and creating its own set of validations and searching best practices on third place. With Mulesoft developers auditing security of 3rd party software and extensive testing got first place, and monitoring integrations and following Mulesoft's designing and building patterns were in third place.

Results show that developers are pretty much in the same place on what comes to ensuring the security of integrations. We saw few differences in the responses, but overall, the responses were in line between different groups.

7. Conclusions

Organisations are forced to take digital leaps to stay relevant. This makes many organisations search best-suited customer relationship management system to add to their information technology infrastructure. This research went through features that create Salesforce's security. Cloud computing infrastructure with all the typical security features and the Salesforce platform's security controls provide a layer of security for organisations. However, integrations can create vulnerabilities. At Salesforce, integrations are commonly either processing and providing information for the 3rd party software or taking data to form 3rd party sources and storing it into Salesforce. If the 3rd party software is exploited the malicious actor can violate the confidentiality and integrity.

The research included a study where, Salesforce developers and Mulesoft developers were asked about integration security and responsibilities related to the integration security. The studies show that all parties are taking ownership of ensuring security. By looking at results conducted with the studies, we can see that small organisations' Salesforce integrations made by inexperienced developers are the most vulnerable. Another key finding was that organisations do not have integrations included in their ISP. Those organisations that do not have ISP at all should create one.

This research brings up guidance and best practices on ensuring the security of integrations. Organisations of all sizes and developers from inexperienced to experienced should ensure that these considerations are understood. We also recommend that all developers and development teams should implement SDL model into their development processes to follow industry best practices developing secure software.

The major drawback of this research was the sample size. It means that results cannot be generalised. There were also design flaws in the survey. Questions about responsibilities should have been created so that differences between stakeholders would have come out more clearly. Without a doubt there is need for further research.

Acknowledgment

University of Jyväskylä is acknowledged for financial support of the research.

References

[1] Rot, A. and Sobinska, M. (2020). "Challenges for Knowledge Management in Digital Business Models," 2020 10th International Conference on Advanced Computer Information Technologies (ACIT), 2020, pp. 555-558, doi: 10.1109/ACIT49673.2020.9208867.

[2] Mydyti, H.; Ajdari, J. and Zenuni, X. (2020). "Cloud-based Services Approach as Accelerator in Empowering Digital Transformation," 2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO), 2020, pp. 1390-1396, doi: 10.23919/MIPRO48935.2020.9245192.

[3] Coltman, T. R. (2006). "Where Are the Benefits in CRM Technology Investment?" Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06), 2006, pp. 111c-111c, doi: 10.1109/HICSS.2006.535.

[4] Patel J. and Chouhan, A. (2016). "An approach to introduce basics of Salesforce.com: A cloud service provider," 2016 International Conference on Communication and Electronics Systems (ICCES), 2016, pp. 1-8, doi: 10.1109/CESYS.2016.7889991.

[5] Manchar, A. & Chouhan, A. (2017). "Salesforce CRM: A new way of managing customer relationship in cloud environment," 2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT), 2017, pp. 1-4, doi: 10.1109/ICECCT.2017.8117887.

[6] Seth, M. (2018). "Mulesoft – Salesforce Integration Using Batch Processing," 2018 5th International Conference on Computational Science/ Intelligence and Applied Informatics (CSII), 2018, pp. 7-14, doi: 10.1109/CSII.2018.00009.

[7] Patel, J. & Chouhan, A. (2017). "An integration of salesforce.com with Twitter: A case of AppExchange," 2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT), 2017, pp. 1-6, doi: 10.1109/ICECCT.2017.8117882.

[8] Soni, K. & Vala, B. (2017). "Roadmap to salesforce security governance & salesforce access management," 2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT), 2017, pp. 1-4, doi: 10.1109/ICECCT.2017.8117831.

[9] Seify, M. (2006). "New Method for Risk Management in CRM Security Management," Third International Conference on Information Technology: New Generations (ITNG'06), 2006, pp. 440-445, doi: 10.1109/ITNG.2006.99.

[10] Lakaniemi, Ilkka (2014). Digitalisaatio keskisuurissa yrityksissä. Liikenne- ja Viestintäministeriön julkaisuja 14/2014. Liikenne- ja Viestintäministeriö. ISBN: 978-952-243-399-2.

[11] Williams, D. (2014). Connected CRM: Implementing a big-data-driven, customer-centric business strategy. John Wiley & Sons, Inc.

[12] Spremić, M., & Šimunic, A. (2018). Cyber security challenges in digital economy. In Proceedings of the World Congress on Engineering,1, pp. 341-346). ISSN: 2078-0958.

[13] He, W., & Zhang, Z. (Justin). (2019). Enterprise cybersecurity training and awareness programs: Recommendations for success. Journal of Organizational Computing and Electronic Commerce, 1–9. https://doi.org/10.1080/10919392.2019.1611528.

[14] Aldawood, H., & Skinner, G. (2019). Challenges of Implementing Training and Awareness Programs Targeting Cyber Security Social Engineering. 2019 Cybersecurity and Cyberforensics Conference (CCC). https://doi.org/10.1109/ccc.2019.00004.

[15] Aboelfotoh Aboelfotoh, S. F., & Hikal, N. A. (2019). A Review of Cyber-security Measuring and Assessment Methods for Modern Enterprises. JOIV: International Journal on Informatics Visualization, 3(2), 157-176. https://doi.org/10.30630/joiv.3.2.239.

[16] Wang, L., & Wang, X. V. (2017). Challenges in Cybersecurity. Cloud-Based Cyber-Physical Systems in Manufacturing, 63–79. https://doi.org/10.1007/978-3-319-67693-7_3.

[17] Soltani Z., Navimipour N. J. (2016), Customer relationship management mechanisms: A systematic review of the state of the art literature and recommendations for future research. Computers in Human Behavior, Elsevier, Volume 61, 667-688. https://doi.org/10.1016/j.chb.2016.03.008.

[18] Bakator, M., Đorđević, D., Ćoćkalo, D. Z., Ćeha, M. (2021). CRM and customer data: Challenges of conducting business in digital economy. In Journal of Engineering Management and Competitiveness, 11(2):85-95. DOI:10.5937/jemc2102085B.

[19] Stewart, H. (2022). Security versus Compliance: An Empirical Study of the Impact of Industry Standards Compliance on Application Security. In International Journal of Software Engineering and Knowledge Engineering, Vol. 32, No. 03, pp. 363-393.

[20] Backes, M., Bugiel, S., Derr, E., (2016). Reliable Third-Party Library Detection in Android and its Security Applications in Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 356–367.

[21] Khaim, R., Naz, S., Abbas, F., Iqbal, N., Hamayun, M. (2016). A Review of Security Integration Technique in Agile Software in International Journal of Software Engineering & Applications (IJSEA), Vol.7, No.3.

[22] Neha. (2020, November 06). Everything you need to know about data privacy in salesforce. Retrieved December 16, 2022, from https://docmation.com/data-privacy-in-salesforce/.

[23] Ransome, J. F., Misra, A., Schoenfield, B. & Schmidt, H. A. (2014). Core software security: Security at the source. CRC Press, Taylor & Francis Group.

[24] Microsoft (24.08.2022). What are the Microsoft SDL practices? https://www.microsoft.com/en-us/securityengineering/sdl/practices.

[25] NIST (24.08.2022). Secure Software Development Framework SSDF. https://csrc.nist.gov/Projects/ssdf.

[26] BSIMM (24.08.2022). BSIMM12 2021 Insights & trends report. https://www.bsimm.com/content/dam/bsimm/reports/bsimm12.pdf

[27] OWASP (24.08.2022). OWASP SAMM. https://owaspsamm.org/.

[28] Renwick, Priscila (2022). Ultimate Introduction to Salesforce Integration. SalesforceBen. Online article: https://www.salesforceben.com/salesforce-integration/ - Sited 21.07.2022.

[29] Bass, J. M.; Beecham, S. & Noll, J. (2018). Experience of Industry Case Studies: A Comparison of Multi-Case and Embedded Case Study Methods. 2018 IEEE/ACM 6th International Workshop on Conducting Empirical Studies in Industry (CESI), pp. 13-20. ISBN: 978-1-4503-5736-4.

[30] Ilmarinen, V. & Koskela, K. (2015). Digitalisaatio: Yritysjohdon käsikirja (1. ed.). Talentum.