Crimson Publishers
Wings to the Research

Review Article

Exploring and Managing the Impact of Cyber Security on Customer Experience in Cloud Hosted Software Services

Dominique Barker[1]* and Nicholas Patterson[2]

[1]DeakinCo, Australia

[2]Deakin University, Geelong, Australia

ISSN: 2640-9739

***[1]Corresponding author:** Dominique Barker, DeakinCo, Melbourne, Australia

Submission: March 21, 2019

Published: April 17, 2019

Volume 1 - Issue 5

How to cite this article: Dominique B, Nicholas P. Exploring and Managing the Impact of Cyber Security on Customer Experience in Cloud Hosted Software Services. COJ Elec Communicat. 1(5). COJEC.000521.2019.

Abstract

The Customer Experience (CX) ideology has continued to dominate the pages of marketing strategies across varying organizations and industries. Creating and maintaining a strong customer following has become synonymous with business success. Vendors attempt to drive customer loyalty and profit retention by offering an omni-channel, media-rich experience. At every turn, the digital world attempts to lure customers with gamified widgets, rewards programs and personalized software services. One less considered interaction affecting the CX journey, online security, has been somewhat ignored, assumed, or even, taken for granted in cases. The unfortunate side to this equation is that this can have catastrophic effects on the customer-and their experience. The purpose of this research is to understand if it's possible to deliver the same level of CX inside a high-security environment, as inside a low or medium security environment, using cloud-hosted software as a service (SaaS) as a basis for this study. By using text-mining techniques, data collection and testing a defined use case, we ascertained that security has an impact on CX in cloud-hosted software services. As a result of this research we have identified techniques for creating inclusive, security-conscious SaaS environments with the overall goal of maximizing audience participation.

Keywords: Information systems; Information security; Cyber security; Access control; Cloud services; Corporate data; IT industry

Introduction

Software as a service (SaaS) has become a popular way for organizations to create links in their digital value chain [1]. Its efficiency to get up and running, low-cost and low-barrier to entry make it a practical solution for businesses. However, previous studies indicate that a tenuous link between a positive customer experience (CX) and SaaS remains [1] and its management is not widely understood [2]. CX or client satisfaction is highly dependent on how much a customer or client trusts your brand [3]. Issues, bugs or sub-par interactions within online platforms may damage the trust relationship for the customer. The element of trust is becoming more and more powerful with regards to internet technologies and the customer experience. Customers are increasingly becoming omnichannel shoppers and using many avenues such as physical stores, websites, social platforms and mobile apps to conduct their purchases [4]. If a customer no longer trusts your brand, then this can have a direct impact on your business.

Additionally, measuring CX in this space is difficult, because it is impressionistic and based on all of the interactions a customer may have with an organization. As part of this research, we will isolate security as a key variable in the digital CX and consider results from data collected across three groups of customers, operating inside varying levels of secure environments. The overall aim of this research is to look at security in a SaaS context, considering its effects on CX. The problems or research rationale that exists in this space are as follows:

A. Security is such a major part of the customer experience (CX) that if it is not considered by the vendor up front, it can actually block the customer from completing their user journey. This would quite obviously contribute to a poor CX.

B. When architecting SaaS solutions for potential customers, it is difficult to understand the internal environments or security restrictions in which they will complete their user journey, particularly inside business-to-business (B2B) environments.

The research outcomes show that users in high-security environments are completely blocked at certain points in the user journey. This heavily impedes CX, meaning that one interaction can in fact, be integral to the total experience

Related Work

Modern society is heavily dependent on software services, which should be reliable, maintainable and secure [5]. In fact, the economic opportunity and growth of a nation's wealth and opportunities are dependent on the development and use of digital technologies, especially as the world is becoming interconnected through the internet [6]. As it stands, existing research specifically related to CX within high-security SaaS environments is limited. This is largely due to the fact that as a concept, CX is fairly new and is predominantly considered a marketing function [2].

Online security on the other hand, is always a technology function [7]. Although there is an interdependence on security being up to scratch to facilitate the CX, this is yet to be recognized as an important interaction in the context of SaaS. Goode et al. [8] argues this is because in a B2B setting, security is subjective and is closely related to the client's experiences and biases. Here we will look at some of the literature that exists for each to identify potential areas for scholarship.

Understanding customer experience (CX)

CX can be defined as a customer's impression of the quality of a company and its affiliated products and services, based on the totality of interactions they have with an organization-this could be digitally, face-to-face or by some other medium [1]. This is a common definition that exists for CX. As a result, this could potentially mean that CX could be overly generalized, while not addressing specific issues that exist within different interactions. "Experience" as a term can be also be subjective in its interpretation. This is because ultimately, it relates to how a customer "feels" about your organization. However, it is important to note that some interactions have more effect on the CX than others: for example, if an online security discrepancy completely blocks the user in their given journey-how will that user feel about the organization then? Undoubtedly, CX has the ability to strongly influence the reputation of any company [9].

At an organization, according to Batra [1], CX typically has three key stakeholder groups:

A. Marketing and product strategy staff, sales staff, account managers, middle management and customer service staff.

B. In-house or external digital providers such as SaaS providers, (technology teams) and

C. Industry analysts, experts, thought leaders and researchers.

If we consider the stratification of the above stakeholder groups, the outcome is that the implications for CX from a technical perspective are not unanimously understood across an organization. Furthermore, as a concept, management of CX across many groups is not widely understood either [2]. What we do know however, is that if issues are not addressed at an interaction level it can lead to a poor CX. Examples of these within a security context might be, digitally-inhibited CX due to external network parameters, unfixed system defects which are not known to the vendor but impede CX, or differences in system requirements. The digital "part" of the customer journey is often one of the last interactions. And, importantly, this is usually what the customer remembers about their experience with your organization.

Some practitioners argue that you can "get away" with a few sub-par interactions, as long as the majority of experiences that the customer has are good. As referred to by Batra [1] it is the customer's perception of the overall experience that matters. However, this really depends on the importance of the interaction itself. Additionally, it is often difficult or even idealistic to expect organizations to adopt this level of macro-thinking. When we add a variable as serious as security to the mix, it has been said that we cannot rely on end users to make good security decisions [10]. Instead, the responsibility falls upon the vendor to cultivate best practice security features to facilitate the process.

Security and CX

Let's now look at some of the disparity that exists between security practices and how they are executed in the context of CX. Firstly, we must consider the software service that was used as a basis for this research. Our custom-built micro-credentialing software service, (which allows users to be credentialed for their existing capability, and be awarded a digital badge), was a completely new product in the education space. With that in mind, the marketplace, target demographic and the needs of the users started out as a black box. Its specifications are similar to an e-learning platform, and as such, some of the key security characteristics can be identified as data integrity and privacy [11].

Iteration two of our software service was used as the basis for this research. We know, as mentioned by Vakeel et al. [12], that often the original security policies represent only the vendor's perspective. This was correct in this case also. Subsequently, we have seen a misalignment between the actual, and required policies. This is particularly true at a business to business (B2B) level. The reason for this, is that even though our target audience is both business to customer (B2C) and B2B, vendors often apply the same policy to both as clearly acknowledged by Vakeel et al. [12]. This is both ignorance and negligence on the vendor's part-and our previous solution was no exception to this common security trap. The research design goes some way to segmenting these user groups to offer greater levels of personalization. We know that B2C audiences care more about intimacy and privacy of their personal details, but, B2B audiences are more concerned with security at a broader level, such as hosting providers and access [12].

The adoption and use of cloud computing are expected to become more pervasive as time goes on, leading it to becoming a fundamental cornerstone of the Internet [13]. Cloud computing now actually has evolved to include management of processes, storage of big data which has come about through technologies such as search engines and their optimization through machine learning [14]. When we consider hosting providers and access,

although the use of cloud-hosted software services is appealing for vendors because of its ability to provide flexible, just-in-time services in a low-cost manner, [15], its security vulnerabilities tend to be the biggest downside. Right now, the only real barriers in place to prevent attacks are legislation, contract, and good practice. Cloud providers do what they can to restrict data access to as few employees as possible [16]. However, as we know in the process of data collection, storage and use, this can lead to issues in the security realm with regards to the leakage of personal or customer information as well as elements that deem data being difficult to discern [17].

If your client is security-conscious, they will have certain restrictions in place to prevent access to some digital platforms. At a less severe level, just some restrictions may apply to certain features on your SaaS. The result of this is a mismatch between the stated and desired security policy [16] as well as a compromised CX, as it is unlikely that a user will be able to complete their journey. However, what happens if your client or customer doesn't value security? [8]. From the vendor's perspective, up-front, time and money could be spent for an audience who derives only marginal benefits, or worse, no benefit at all from expensive security enhancements. There is also little doubt that security-conscious organizations believe that sensitive data uploaded outside of the organization's network is inherently riskier, [7] and this mind-set will not easily be changed.

What we can see from this literature review is that customer experience as a stand-alone concept is not enough-and that specific interactions (such as security-affected software experiences) should be considered important enough to make or break a positive CX. As vendors, we must meet our legal and ethical obligations to protect the privacy of our users, and in a B2B context, we must uphold and adhere to existing internal security practices to extend on the organization's efforts to protect sensitive information. This is what drives a positive CX from the inside out. The methodology assists us with testing numerous security features and measuring a baseline of support tickets across varied environments to support this case, eventually assisting us with building security profiles for our customers, a concept borrowed from Cotroneo et al. [18].

Materials and Methods

The methodology used for this research was primarily quantitative. The first step was to identify a baseline of security issues logged by cohorts or users operating in different environments from between May-December 2017. Each of these cohorts would represent a low, medium or high security environment (as defined below in Figure 1 and would allow us to provide CX-related security issues data to evidence the need for this research. Each cohort consists of around n=30 users. One group exists within a high-security environment, and the other within a medium security environment. Organizational cohorts of users do not currently exist within a low security context, as this has always been attributed to B2C users. For this group, we considered individual users within a sample size of n=30 (similar to the cohort). These users formed the low security environment.

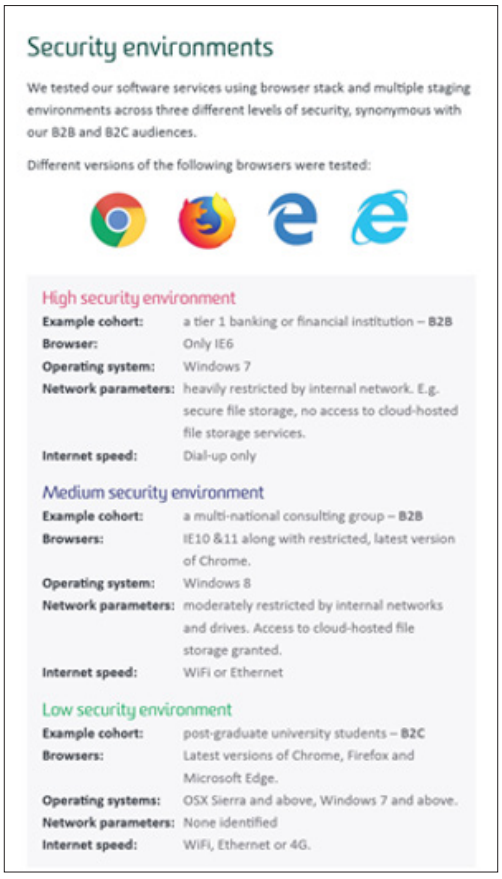

Figure 1. Browser and associated variables.

Establishing the security baseline

To obtain this data, we had to apply text mining techniques to all of the helpdesk queries that we had received over the period of May-December 2017. As we had already been working on these tickets, we had non-specific knowledge of the causal relationship between CX-related security issues and the type of cohort affected.

Effective text mining techniques such as text categorization within email topic modelling helped us to identify hidden consistencies or relationships through pattern recognition, facilitating extraction of more meaningful information [19]. As such, we had to narrow down the helpdesk queries by [NAME OF ORGANISATION] and [TYPE OF HELPDESK ENQUIRY]. Secondary to this, we classified queries that were related to the following variables that we wanted to control as part of this research.

Variables

A. Browsers and operating systems

B. Internal network parameters

C. Internet speed

Trialing secure and non-secure use cases

Once we had established the baseline of security-related CX queries, results shown in Figure 1 we were able to isolate key variables for further interrogation.

The use case that was being followed is below:

A. Login to the platform as candidate (user).

B. Select and save your criteria.

C. Upload and evidence files of your professional experience (.mp4, .mp3, .pdf, .docx, .ppt, .txt. xlxs, .png).

D. Upload and attach a testimony file (docx.)

E. Declare that it is your own work and submit your files for assessment.

For the high and medium security environments, we created an additional staging site to allow us to add or remove certain features to test their usability within different levels of security constraints (Table 1).

Table 1: Testing environment sections.

Environment 1	High
Browser	Test with browser stack Internet Explorer 6
Operating system	Test with browser stack Windows 7
Internal Network parameters	Identified. Use local file uploader only
Internet speed	Code changes and Plupload libraries required

Environment 2	Medium
Browser operating system	Test with browser stack latest versions of Chrome, Firefox and Edge as well as Internet Explorer 10 and 11
Operating system	Test with browser stack Windows 8
Internal Network parameters	Identified. Use local file uploader only
Internet speed	Normal (4G, Wi-Fi, Ethernet) No changes required

Environment 3	Low
Browser	Test with browser stack latest versions of Chrome, Firefox and Edge
Operating system	OSX Sierra, Windows 7 and above
Internal Network parameters	None identified - use Upload Care widget for file uploads
Internet speed	Normal (4G, Wi-Fi, no changes required)

Browser and operating systems

One of the key features of SaaS in the context of cloud computing is that a user should be able to access the software from anywhere, at any time, as long as they have an internet connection [20]. Using Browser stack [21] for testing was a way to verify if this was actually true. We tested the different operating systems and browsers using cloud testing techniques. We can see from the different levels of security that candidates operating within different environments have different levels of browser requirements, ranging from very old, (Internet Explorer 6) to current (latest versions of Chrome and Firefox).

Browser stack is an example of testing as a service (TaaS) provider [22]. It works by running a live version of the browser you want to test, in your actual browser. This can mean that it runs more slowly but it is an accurate way to ascertain how your software service is behaving. It also allows the user to test with different combinations of browsers and operating systems e.g. Windows 7 and IE11, Windows 8 and IE11. It is most useful for testing of simple, linear use cases as it is limited to manual testing and does not include automation options, or pass/fail statuses [22].

Internal and network parameters: Internal network parameters that can negatively influence the CX can generally only be known when we are notified directly by clients. However, one of the key features that impacts security is file upload ability. According to Uddin & Jabr [23] file uploads are one of the most highly-utilized functionalities within SaaS applications, but unfortunately, if correct sanitization and security methods are not employed, [23], significant security issues are presented to the user. For example, we had previously used a separate widget called Upload care for uploading files, which supports multiple (simultaneous) file uploads, social networks and cloud storage integration [24]. On the downside, such flexibility is not welcome in high-security environments. Upload care is not concerned with file sanitation and is also dynamically impacted by different cloud SaaS providers' security policies. As part of this research we removed the Upload care widget and replaced it with a standard, local file upload feature only. Because of our very limited use case noted above, this was the main feature that was likely to affect users within our SaaS. However, internet speed issues noted below do overlap with this variable.

Internet speed: In the high-security environment defined above, we can see that internet speed is used inside these organizations to inhibit internet use. By default, this creates a poor user experience, which is of course one of its intentions, to discourage staff from accessing websites that could compromise security. As the use case required inside our SaaS requires uploads of multiple files (including a combination by the user of potentially video, audio and large document types) we needed to find a solution for this. The solution we decided to test as part of this research involved writing some custom code and then utilizing the libraries from a product called Plupload [25]. This allows batch uploading of multiple files, but the difference is, it will process this upload in smaller chunks-i.e. a batch at a time, which allows the uploads to be completed on slower internet connections.

Controlling the variables

Once there was a decision made on how we would adjust each of the variables, the testing environments were defined and set up as in Table 1.

Browser compatibility issues: Users within high and medium security environments were affected by browser compatibility. At the time data was collected, our SaaS, created in March 2017, only supported the latest versions of Chrome, Firefox and Microsoft Edge browsers. Small adaptations were made to upload care, which is mentioned below in the internal network parameters section. End-to-end testing on Browser stack using a combination of IE6 and Windows 7 indicates software compatibility has now been achieved. Greater browser compatibility was also possible through the optimization of JavaScript code which has the added benefit of improving the user interface, specifically for older browsers such as IE10.

Internal network parameter issues: Users within high and medium security environments typifying very large financial institutions or multinational consulting groups, experienced increased "blockages" from their internal networks. This prevented them from uploading important files. Without making use of the file upload feature, they would be unable to complete their submissions. It would simply say "file upload failed." In this case, no other error reporting information was exposed to the user. It is also important to note here that there were some interesting findings on file sanitization, and the results remain somewhat unclear. This is because we cannot be exactly sure what the user is seeing on their side. This was evident in our medium security environment.

We do know, that some users within the medium security environment were blocked by their internal network parameters when it came to uploading files in some cases. This was because issues were detected locally, and the user was notified with an "upload failed" message within our SaaS. However, if they uploaded a sanitized file, or a file from an approved location (such as a specific file or drive) there were no issues. This meant, that a handful of users would have issues with some files but not others depending where they were stored. At a CX level, the interface did not expose specific messaging to the user as to why the file had failed. And in some cases, we would not know why vendor-side. This affected troubleshooting due to the sporadic, non-specific nature of the upload rejection, and the fact that it was client side meant the reason for the fail was not obvious. Replacing the cloud file uploader (Upload care) with a local file uploader, meant that there were less issues with the upload being blocked, but our findings indicate we are still unable to replicate the testing of classified files from an unapproved local location inside of our testing environments. This was particularly pertinent in the medium-security environment.

Internet speed: Users on dial-up internet connections, in some cases (depending on the file size) were not able to upload files at all, completely blocking the customer experience. This was because the file was too large, particularly when it came to uploading video or audio files. To get around this, the batch uploader solved this issue - this was tested with files up to 20MB, which is the maximum file size supported by the system. Rather, than attempting to upload the file all at once, the batches allowed a bit at a time. It is important to note that this is still not an optimal CX and cannot be as good when compared with a user on a Wi-Fi, Ethernet or cable connection. If you have only ever had dial up speed however, perhaps your tolerance for a compromised CX is higher. To put it into perspective, 512kb/s download speed is common for dial up, while I just ran a home speed test on my cable internet, which indicates around 28mb/s. Each mb represents 1024kb. Needless to say, the difference is significant. Note that internet speeds inside organizations not running dial-up, will be higher again, when compared with home internet speeds.

Discussion

When we began this research, we had existing data regarding the candidate journey to assist with "proving" whether or not the CX is compromised within varying levels of security environments. Batra [1] states that CX can be based only on the totality of a given user's experience. This means we should not focus on individual touchpoints. However, our research has found that some

touchpoints are so integral to the CX, which ignoring them as a specific function or interaction can be the difference between a good or bad CX. Vakeel et al. [12] found that in e-commerce, there was a stark difference a B2B and a B2C's security and privacy needs. We needed to focus on both as a primary (B2B) and secondary (B2C) audience as part of this research. By defining the security profiles, we were actually able to understand all requirements in the same place. They were just profiled differently. Although their needs are substantially different as shown in Figure 2, our results illustrate how both needs can be met in the same environment.

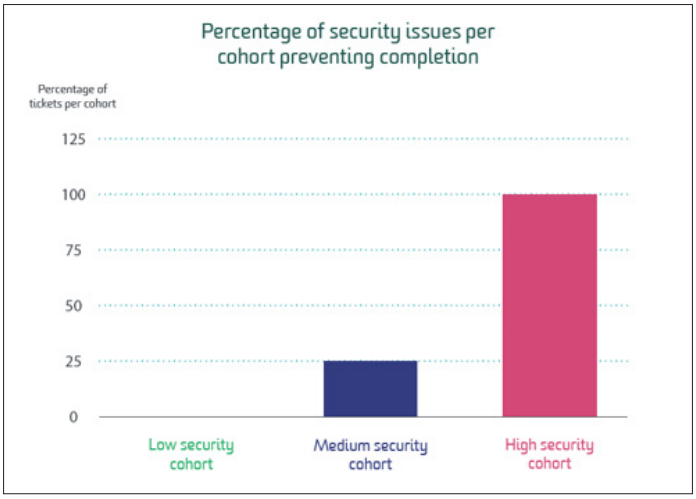

Figure 2. Percentage of security issues per cohort preventing completion..

We also found that Browser stack was the best testing service to identically mimic the user's environment. Although it has been noted that Browser stack's lack of automated testing is a limitation, [22] we found that for CX testing, this gave us the most exact results and did not see it as a limitation in this context. It must be noted however, that like any software-related defect or user error, we ran into issues when we were unsure of internal network parameters blocking certain user actions on the client-side. If we cannot know this from the client upfront, this will be continually problematic. There are situations where this does happen, and at that point, we request a phone call with their technology team. Therefore, training of sales and frontline staff is essential to allow them to collect this information and facilitate changes before B2B cohorts begin using the SaaS.

To address any functionality gaps or restrictions that may exist after this point, we will collect data on an ongoing basis. The software service employed as part of this research is underpinned by micro-level data capturing and reporting of a user's given activities as they move through the system. Data was collected using techniques adapted from experience application programming interface. xAPI [26] is a service that allows us to create statements of experience and then capture these securely for storage in a learning record store (LRS). If we cannot be all-knowing at a pre-sales stage, we can at least draw confidence from real-time user activity statements to address contentious digital interactions, allowing us to troubleshoot more quickly. In saying this, this research certainly bought security considerations to the fore and forced us to look at additional security practices that could be implemented. In addition to this, the revised upload functionality is a direct upload from the user's machine into an AWS securely-hosted S3 bucket. The benefit of this is that is considered industry best-practice. This is because there is a direct connection between the user's data and S3. There is no intermediary required. Data is passed using the standard HTTPS protocol and stored in S3 using 256bit encryption [27].

The hashed URL's of the file uploads were also modified to ensure they were protected by authentication - this means that the files could only be accessed if a user was logged in. This also relates back to our original problem-that it is difficult to know all issues that will be encountered, but if we start to follow best practice, this will give us a head start with regards to security compatibility, as well as building better trust relationships with the client.

Conclusion

This research had certain limitations that the reader should consider. Firstly, the sample size of the cohorts tested with each environment only consisted of around n=30 users. In saying that, the issues above manifested within each of the variables in a way that allows us to conclusively state that cohorts operating in similar environments would experience the same issues (although there would be small fluctuations) this indicates that the sample size itself becomes almost irrelevant. Additionally, this research was conducted under specific conditions, using a linear use case. Future research should consider greater diversity when testing specific software features synonymous with SaaS solutions. We were limited to testing file uploads, Internet speed and browser compatibility. From a CX perspective, there is also no data on how the users in each group were feeling before they had issues. Capturing a user's feelings about the software throughout the entire journey in relation to digital experiences, is an area ripe for research.

The results show that CX can be heavily impeded by security discrepancies across varying environments, as identified in the research problems. This indicates that security considerations form a critical part of the discovery phase of any software project.

Although understanding all security requirements upfront is nigh impossible, we must work closely with external clients to understand their network parameters and seek to make ongoing changes to cultivate best practice security. Marginal software features were removed in this case to make our software usable by the lowest common denominator. When it came down to it, the changes were so minimal that the effects on the overall CX were trivial at best. This meant, that with minor changes, we could deliver a consistent user journey across all environments. The CX now remains somewhat agnostic to the level of security imposed by the described internal networks, internet speed and browsers and operating systems. Going forward, third-party suppliers of SaaS widgets, should improve their products through better understanding of operable environments, like those highlighted in this study.

References

1. Batra MM (2017) Customer experience-an emerging frontier in customer service excellence. Competition Forum- American society for competitiveness 15(1): 198-207.
2. Homburg C, Jozić D, Kuehnl C (2017) Customer experience management: Toward implementing an evolving marketing concept. Journal of the Academy of Marketing Science 45(3): 377-401.
3. Chou SW, Chiang CH (2013) Understanding the formation of software-as-a-service (SaaS) satisfaction from the perspective of service quality. Decision Support Systems 56: 148-155.
4. Parise S, Guinan PJ, Kafka R (2016) Solving the crisis of immediacy: How digital technology can transform the customer experience. Business Horizons 59(4): 411-420.
5. van Deursen A, Mesbah A, Nederlof A (2015) Crawl-based analysis of web applications: Prospects and challenges. Science of Computer Programming 97(1): 173-180.
6. Teoh CS, Mahmood AK (2017) National cyber security strategies for digital economy. International Conference on Research and Innovation in Information Systems (ICRIIS), USA, pp. 1-6.
7. Sabahi F (2011) Cloud computing security threats and responses. Communication Software and Networks (ICCSN), IEEE 3rd International Conference, USA, pp. 255-299.
8. Goode S, Lin C, Tsai JC, Jiang JJ (2015) Rethinking the role of security in client satisfaction with Software-as-a-Service (SaaS) providers. Decision Support Systems 70: 73-85.
9. Foroudi P, Jin Z, Gupta S, Melewar TC, Foroudi MM (2016) Influence of innovation capability and customer experience on reputation and loyalty. Journal of Business Research 69(11): 4882-4889.
10. Akhawe D, Felt AP (2013) Alice in warning land: A large-scale field study of browser security warning effectiveness. Proceedings of the 22nd USENIX conference on Security, Washington DC, USA.
11. Luminita DC (2011) Information security in E-learning platforms. Procedia-Social and Behavioral Sciences 15: 2689-2693.
12. Vakeel KA, Das S, Udo GJ, Bagchi K (2017) Do security and privacy policies in B2B and B2C e-commerce differ? A comparative study using content analysis. Behaviour & Information Technology 36(4): 390-403.
13. Botta A, Donato Wde, Persico V, Pescapé A (2016) Integration of cloud computing and internet of things: A survey. Future Generation Computer Systems 56: 684-700.
14. Singh J, Pasquier T, Bacon J, Ko H, Eyers D (2016) Twenty security considerations for cloud-supported internet of things. IEEE Internet of Things Journal 3(3): 269-284.
15. Singh S, Jeong YS, Park JH (2016) A survey on cloud computing security: Issues, threats, and solutions. Journal of Network and Computer Applications 75: 200-222.
16. Ryan MD (2013) Cloud computing security: The scientific challenge, and a survey of solutions. Journal of Systems and Software 86(9): 2263-2268.
17. Zhang D (2018) Big data security and privacy protection. Atlantis Press 77: 275-278.
18. Cotroneo D, Paudice A, Pecchia A (2016) Automated root cause identification of security alerts: Evaluation in a SaaS cloud. Future Generation Computer Systems 56: 375-387.
19. Kim Y, Ju Y, Hong S, Jeon SR (2017) Practical text mining for trend analysis: Ontology to visualization in aerospace technology. KSII Transactions on Internet and Information Systems 11(8): 4133-4145.
20. Fan H, Hussain FK, Younas M, Hussain OK (2015) An integrated personalization framework for SaaS-based cloud services. Future Generation Computer Systems 53: 157-173.
21. (2018) Browserstack. List of browsers and operating systems.
22. Oliveira RRd, Martins RM, Simao AdS (2017) Impact of the vendor lock-in problem on testing as a service (TaaS). IEEE International Conference on Cloud Engineering (IC2E).
23. Uddin N, Jabr M (2016) File upload security and validation in context of software as a service cloud model. 6th International Conference on IT Convergence and Security (ICITCS) pp. 1-5.
24. (2017) Upload care. Upload care widget documentation.
25. (2018) GitHub.
26. (2018) xAPI.
27. AMAZON (2018) Protecting Data in Amazon S3.