Concerns About Security on Cloud

Cloud systems are useful for companies that do not want to have their operations on site for many reasons such as cost or maintenance. When the system is not owned by the company, it is true that the company does not have to work on the configurations or maintenance, however, that also means a lack of control over the system for any specifications or security. Soofi et al. (2014) claims that there is a major number of cloud-based projects that are not acceptable because of security concerns. Gonzalez et al. (2012) studies security in three aspects; architecture, compliance and privacy. Architecture consists of network security, interfaces and virtualization. Compliance is split up into two as service and provider. Finally, privacy is split up into two as data security and legal issues. Identity management technologies solve a majority of these concerns. The term identity management describes the process of managing individuals, entities or group authentication, authorization, roles, privileges, etc. across the enterprise information systems suite. An identity management system is most of the time centralized because tracking the same user across all of the application accesses, to have consistency and standardization across the whole system, and reduction or elimination of duplication costs in the system. All of the large providers offer, as a packaged service, some kind of identity management tier which can be used to mediate access to cloud resources, but cloud-based applications need to support some identity management system at the application layer which may or may not be able to leverage the vendor’s offering. User identities are created using a process often known as activation. This is a one-time event where a user is created within the identity management system and assigned the appropriate privileges. For example, onboarding a new employee or creating a materials supplier account. The term authenticate means to establish and verify the identity of some user. Activated users authenticated through credentials usually comprising a publicly known part and one or more secrets, known only to the user. The base assumption is that the activated user who can successfully pass a credentials check is authenticated. The authenticated user then assumes some preassigned identity within the applications for subsequent access and actions. Security is second only to correctness in enterprise information systems because of the sensitivity of the data being managed and processed. As authentication is the first line of defense of the access network and the application software itself, enterprises increasingly employ ever more sophisticated credential schemes to make them harder to compromise. In general, the more secrets that are required, the more secure the scheme will be. Multi-factor authentication implies two or more secrets are needed; something you know and something you have, includes a so-called airgap. The chicken and egg problem for authentication systems is how the would-be authenticating user has possession of their credentials before they need to use them (e.g., a shared secret or two-factor device). This is typically solved by using some out-of-band distribution mechanism, i.e., over some other interface which is not the API itself. Credentials distribution can be a complex problem and is the concern of the wider identify management system itself. JSON web tokens (JWTs) are an example of a pre-authentication and token scheme. In pre-authentication schemes, the API caller performs the authentication steps in advance and then uses some issued token to make subsequent API requests. The issued token, usually temporary in nature, acts as a cheaper-to-implement proxy for having to authenticate each request separately. In addition to being less resource intensive, JWT pre-authentication, being stateless, can also be easier to distribute, allowing for better performance and scalability characteristics of the API system. The JWT specification allows for different crypto technologies to be used in the signing process. The authenticating authority can include a field in the token specifying which hashing and encryption algorithm was used to create the token signature (e.g., HMAC-SHA256 or RSA-SHA256). If symmetric key signing is used, then both the issuer and the verifier need to possess the same shared secret. If asymmetric key signing is used, then the issuer can hold the private key and the verifiers can use the public key. It is assumed the shared secret key is known only to the client and server and would be “hard” to guess (e.g., 320 bits). The signature is based on a known-secure hashing algorithm with an extremely low probability of collisions (e.g., SHA256). The hash is encrypted using a shared key symmetric-key algorithm (e.g., HMAC-SHA256). It is impossible for an attacker to generate the MAC of message and a secret key without knowing the secret key. The message cannot be altered by an attacker and also pass the message integrity and signature check. These techniques are implemented and are presented as a product in examples such as Ondato, Microsoft Identity Manager, Microsoft Azure Active Directory, Oracle Identity Management, Okta Identity Management, Zoho Vault, OneLogin, etc.

References:

Doshi, R., & Kute, V. (2020). A review paper on security concerns in cloud computing and

proposed security models. 2020 International Conference on Emerging Trends in Information Technology and Engineering (Ic-ETITE). <https://doi.org/10.1109/icetite47903.2020.37>

Gonzalez, N., Miers, C., Redígolo, F., Simplício, M., Carvalho, T., Näslund, M., & Pourzandi, M.

(2012). A quantitative analysis of current security concerns and solutions for cloud computing. Journal of Cloud Computing: Advances, Systems and Applications, 1(1), 11. <https://doi.org/10.1186/2192-113x-1-11>

Ramgovind, S., Eloff, M. M., &Smith, E. (2010). The management of security in cloud

computing. 2010 Information Security for South Africa. <https://doi.org/10.1109/issa.2010.5588290>

Samarati, P., & De Capitani di Vimercati, S. (2016). Cloud security. Encyclopedia of Cloud

Computing, 205–219. <https://doi.org/10.1002/9781118821930.ch17>

Soofi, A. A., Khan, M. I., & Amin, F. e. (2014). A review on data security in cloud computing.

International Journal of Computer Applications, 94(5), 12–20.