Traceability for Adaptive Information Security in the Cloud

A. Nhlabatsi[1], T. Tun[2], N. Khan[1], Y. Yu[2], A. Bandara[2], K. Khan[1], B. Nuseibeh[1,3]
[1]Department of Computer Science and Engineering, Qatar University, Doha, Qatar
[2]Department of Computing and Communications, The Open University, Milton Keynes, United Kingdom
[3]Lero, University of Limerick, Limerick, Ireland

***Abstract* – One of the key challenges in cloud computing is the security of the consumer data stored and processed by cloud machines. When the usage context of a cloud application changes, or when the context is unknown, there is a risk that security policies are violated. To minimize this risk, cloud applications need to be engineered to adapt their security policies to maintain satisfaction of security requirements despite changes in their usage context. We call such adaptation capability Adaptive Information Security. The paper argues that one of the prerequisites to adaptive information security is the use of traceability as a means to understanding the relationship between security requirements and security policies. Using an example, we motivate the need for improving traceability in the development of cloud applications.**

Keywords - Cloud Applications; Context; Adaptive Information Security; Traceability

I. ADAPTIVE INFORMATION SECURITY IN THE CLOUD

Information security is about protecting valuable information assets from intentional harm [1]. With the popularity of mobile and ubiquitous uses of computing infrastructures, such as the cloud [2], both technical and social contexts in which software applications operate are increasingly dynamic. By context we mean the properties of the environment in which a cloud application operates that have an effect in its behavior. For example we may want a cloud application to change its security behavior depending on where it is used (i.e location), who is using it (i.e. subject), or when it is being used (i.e. time). As a result of dynamic context, the assets, their values, and attack scenarios can change easily from one situation to another, increasing the challenge of finding out what the information assets are, who their owners are, where in the system vulnerabilities lie, and the extent to which the security requirements are satisfied.

A security mechanism that works in one usage context may not work in another. Cloud applications often have a diverse set of consumers with different security requirements that need to be satisfied across different contexts. For cloud applications to comply with the various security requirements despite changes in contexts their security mechanisms need to be flexible.

As a motivating example consider a cloud provider, CloudX, that offers an image storage and processing service through an application, ShutterClick, as illustrated in Figure 1. MediScan and NewsMedia are cloud consumers with subscriptions to ShutterClick. The cloud consumers have both common and specialized information security requirements. For example, MediScan has its specific information security requirements for administrators, doctors, and patients; whilst NewsMedia also has its specific set of information security requirements for journalists, photographers, and editors. Some of the roles are exemplified by: Adam as the CTO at CloudX, Bob as an employee of MediScan responsible for auditing a Service Level Agreement between MediScan and CloudX, and Sara as a developer at CloudX.

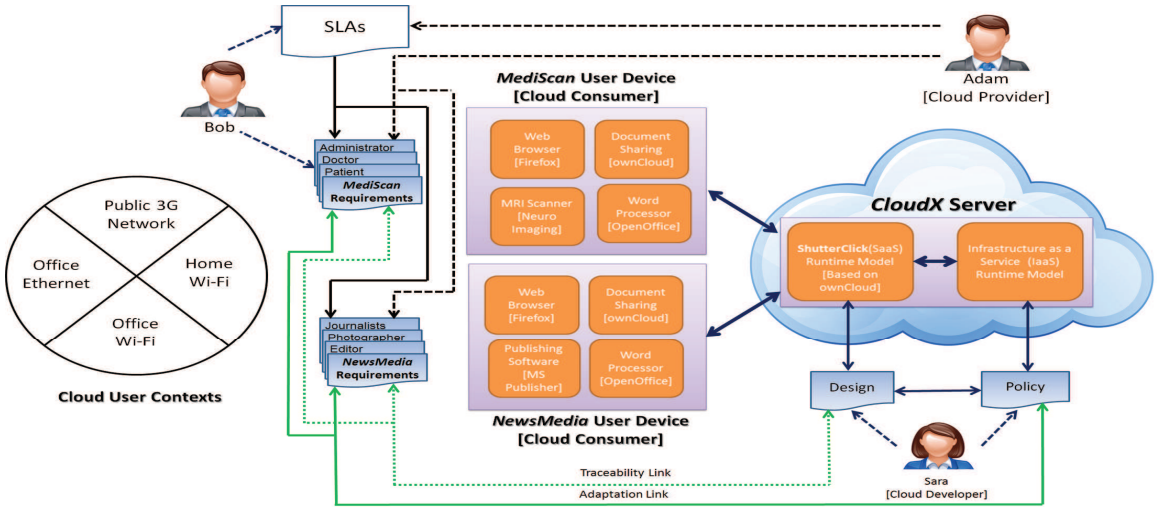

Figure 1. A Cloud Application Usage Scenario Diagram

Furthermore, the different employee roles (e.g., doctors, journalists, etc.) of the cloud consumers may use ShutterClick in different usage contexts. For example a doctor may want to retrieve a medical scan using his office Ethernet, home Wi-Fi, or the public 3G networks. In each case, ShutterClick needs to adapt its access control mechanism depending on the doctor's location when attempting to access the information. When accessing through the hospital Ethernet network no authentication is required but if the doctor is accessing from outside the hospital through a 3G network, authentication is required to preserve the confidentiality of medical records.

II. Traceability for Adaptive Information Security

Current approaches to developing adaptive security applications are based on enumerating the different contexts in which an application operates and engineering behaviors that would be applied exclusively in each context [3]. These approaches only work when the application always encounters known contexts whose attributes were engineered into the application at design-time. They do not handle how the application should behave at run-time when it encounters context properties which were unknown, unclear, or incomplete at design-time. The diversity of cloud consumers and their contexts implies the difficulty for a cloud application to operate securely in a context that was not known at the design-time. To address this problem, cloud applications need the ability to reason about unknown or incomplete contextual properties and assess the impact of changes to such properties on the satisfaction of security requirements. Such an impact analysis can be achieved only if the relationship between requirements and the enforced policies is well understood and maintained.

Therefore, when engineering cloud applications to support adaptive information security it is crucial for a cloud provider to understand the relationship between requirements and policies. In software engineering, such relationships are called *traceability links* [4], which can help the cloud provider in many ways. For example: monitoring the violation of information security requirements at run-time; identifying the policies responsible for the satisfaction of those requirements; assessing completeness and satisfaction of requirements; and certifying the assurance of security requirements.

Traceability links between requirements and policies need to be captured at design-time. In cloud applications, the traceability links should be captured in both forward and reverse directions. In forward traceability, requirements are traced to the components that implement the behavior satisfying the requirements. Upon violation of security requirements at run-time, the reverse traceability should make it possible to diagnose the components that contributed to the violation.

Continuing with our example in Figure 1. Sara uses requirements from MediScan and NewsMedia to derive the design of the access control functionality of ShutterClick and the policies that govern the run-time adaptation of functionality required to handle changes in the usage context. When a confidentiality requirement is violated it is often difficult for Sara to ascertain which parts of the existing ShutterClick application contributed to the violation. To address this problem, at design-time, Sara needs to ensure that each feature (such as authentication) required by MediScan and NewsMedia is traced to the design of ShutterClick. She needs a systematic technique of maintaining the traceability between the access control policies that control the adaptive behavior of ShutterClick and the requirements from which these policies are derived. Sara also needs the traceability to be invariant at runtime– i.e. it should not be accidentally altered as ShutterClick adapts its behavior to satisfy the security requirements of its end users in different contexts. The proposed techniques need to incorporate context information as part of the traceability links in order to understand the effect of changes to contextual properties on the traceability links.

III. Conclusion and Further Work

This paper argues that security in cloud applications need to be adaptive in order to cater for the different variations resulting from requirements of diverse cloud consumers and emergent context properties where cloud applications are used. We have proposed that traceability is a key prerequisite for adaptive information security for understanding the relationship between security requirements and the policies that enforce those requirements. For adaptive information security, traceability has to incorporate contextual assumptions that an application makes into its adaptation mechanisms. This paper is the genesis of an approach we are developing for traceability in adaptive security. We plan to evaluate the proposed approach through case studies and prototypes developed using open source cloud applications such as ownCloud (http://owncloud.org).

Acknowledgment

This work was made possible by the support of a grant (NPRP 05-079-1-018) from the Qatar National Research Fund (QNRF).

References

[1] C. Haley, R. Laney, J. Moffett, and B. Nuseibeh, "Security Requirements Engineering: A Framework for Representation and Analysis," *IEEE Trans Softw Eng*, vol. 34, no. 1, pp. 133–153, Jan. 2008.

[2] C. M. Revoredo da Silva, J. L. Costa da Silva, R. B. Rodrigues, G. M. Medeiros Campos, L. Marques do Nascimento, and V. Cardoso Garcia, "Security Threats in Cloud Computing Models: Domains and Proposals," in *2013 IEEE Sixth International Conference on Cloud Computing (CLOUD)*, 2013, pp. 383–389.

[3] M. Salifu, Y. Yu, A. K. Bandara, and B. Nuseibeh, "Analysing monitoring and switching problems for adaptive systems," *J. Syst. Softw.*, vol. 85, no. 12, pp. 2829–2839, Dec. 2012.

[4] O. C. Z. Gotel and A. C. W. Finkelstein, "An analysis of the requirements traceability problem," in, *Proceedings of the First International Conference on Requirements Engineering, 1994*, 1994, pp. 94–101.