Challenges in Securing the Interface Between the Cloud and Pervasive Systems

Brent Lagesse
Cyberspace Science and Information Intelligence Research
Computational Sciences and Engineering
Oak Ridge National Laboratory
lagessebj@ornl.gov

***Abstract*—Cloud computing presents an opportunity for pervasive systems to leverage computational and storage resources to accomplish tasks that would not normally be possible on such resource-constrained devices. Cloud computing can enable hardware designers to build lighter systems that last longer and are more mobile. Despite the advantages cloud computing offers to the designers of pervasive systems, there are some limitations of leveraging cloud computing that must be addressed. We take the position that cloud-based pervasive system must be secured holistically and discuss ways this might be accomplished.**

In this paper, we discuss a pervasive system utilizing cloud computing resources and issues that must be addressed in such a system. In this system, the user's mobile device cannot always have network access to leverage resources from the cloud, so it must make intelligent decisions about what data should be stored locally and what processes should be run locally. As a result of these decisions, the user becomes vulnerable to attacks while interfacing with the pervasive system.

I. Introduction

Significant research has been focused on both cloud and pervasive computing in the last few years years; however, this research has not always examined the combination of the two ideas. From the perspective of the pervasive computing community, clouds offer an opportunity to perform intense computation and large-scale data storage without introducing a significant load on devices that are designed to be minimally noticeable. The use of clouds can enable hardware designers to create more compact devices that are less obtrusive to users since there is less need for powerful processing and large storage hardware. Further, they can greatly extend the battery life of mobile devices by offloading much of the logic used by applications[1], [2]. Research has begun to emerge that connects pervasive and cloud systems [3], [4], but these works have not yet strongly addressed security issues in these systems. In the remainder of this paper, we talk about the advantages of using cloud-based pervasive systems, and then discuss the security concerns that must be addressed when designing such systems. While our comments are mainly targeted at systems based on traditional cloud computing, the security aspect also applies to non-traditional systems such as cloudlets [4] and ad-hoc clouds.

Consider the following scenario. Alice utilizes a cloud service to execute many of her applications. It also serves as storage for all of her work. This enables Alice to work from anywhere without having to manage the location of her projects or to worry if the system she is using has the ability to execute the computationally intense applications that she runs. The downside of such a system is that it requires Alice to have near-constant connectivity to the network providing the cloud resources. As a result, Alice's batteries can be more quickly depleted than necessary if the network connection is not managed correctly. Furthermore, there are many times when Alice does not have network connectivity on the device she is using. In some cases, this is because she does not have network coverage in the area she is visiting or because she is not allowed to use her wireless network cards. In this scenario, Alice is traveling from her home to a conference. She is able to work on her presentation on the train to the airport, but once she boards the plane, she is required to turn off her wireless network card for the duration of the flight; however, she still wants to continue work while on the plane. In this case, her pervasive system should be able to assure that she has the applications and data necessary for her to continue working until she is able to reconnect to the cloud. Furthermore, this should happen secure so that all data and applications are securely transferred to her laptop, and so that the laptop, which is an untrusted device, cannot damage the data and applications through malware that might pre-exist on the laptop.

In this case, the cloud is not just another system, but rather it is an extension of the pervasive computing system. As a result, these types of systems inherit both the vulnerabilities of the pervasive system and the cloud system. Further, they also inherit the constraints of the cloud and pervasive system. Neither traditional approach to cloud security or pervasive security alone is sufficient. Cloud-based approaches to security can typically rely on static and powerful machines to each other to execute their security mechanisms; however, with the addition of pervasive computing elements, the security mechanisms must be adapted to handle resource-constrained and latency-adverse devices. Likewise, pervasive security solutions vary widely based on the type of pervasive system. Many of these solutions utilize trust-based mechanisms. Additionally many of these systems have the opportunity to leverage context information as part of the security mechanism; however, with the addition of the cloud component, which, by its nature,

attempts to abstract away the contextual details of the underlying system, this pervasive approach becomes more difficult to apply to the full system.

II. Advantages of Cloud Computing to Pervasive Systems

Cloud computing offers many advantages to pervasive systems. The most attractive of which is the ability to store large amounts of data and to perform intense computation. Leveraging the resources of the cloud can lead to smaller and cheaper clients that, when paired with cloud services, can accomplish tasks on par with more powerful systems. The cloud also offers an opportunity to offload the issues associated with data management. If a pervasive system uses a reliable cloud service, it can avoid many of the issues associated with mobile data management. Additionally, the cloud also adds an opportunity for certain systems to either remove or add isolation as necessary to pervasive systems.

A. Thin Clients

Thin clients have always been used in designing pervasive systems, but the downside is that these devices are resource-constrained and can very seldom accomplish complex tasks. Thin clients do provide extreme mobility of pervasive system and can often be added to a system for a small cost. Cloud services can enable these thin clients to operate with what appears to be more power.

Cloud services could also act as an interface between the user and the rest of the Internet. Data that the user requests could be acquired by the ISP's cloud and presented in a manner that requires minimal effort by the user's thin client since most of the computation will be performed on the cloud.

B. Data Management

Much research has gone into data management [5], prefetching [6], [7], caching [8], consistency [9], and opportunistic transfer [10], [11] of data in pervasive systems. While much of this work can still be leveraged to improve the interface between pervasive and cloud systems, it becomes less of an essential portion of the system. With the cloud acting as a reliable source of storage, the pervasive system can avoid the burden of actively managing data to provide availability and rely on the cloud services to perform such work.

C. Isolation of Systems

In a technology-rich environment that consists of pervasive technology, but is not truly pervasive yet, users may use multiple pervasive systems. These systems could take advantage of the cloud in two distinctly different ways when this is the case. The first way that the cloud can be used is to remove the isolation between systems. In the case that the pervasive systems that a person uses are from multiple vendors, it is possible that they do not have support through a direct interface for sharing of information and services. The cloud can be used to remedy this situation. Pervasive services that are moved to the cloud can be used by any of the systems, and information such as user context can be shared through cloud storage. There are many issues that would need to be worked out in a system such as this, but the cloud interface could further enhance otherwise isolated system.

The second way that the cloud can be used is to add isolation to systems. Since multiple cloud services exist that could be used with a pervasive system, pervasive systems can utilize several different cloud services to store data and provide other services. The advantage of isolation is that it mitigates the user's exposure to vulnerabilities from improperly secured or malicious cloud providers. If the user has either several or a single pervasive system it is likely that some portions of the pervasive systems do not really need to know anything occurring elsewhere. In the case of a single pervasive system, the system can avoid exposing personal data to risk by splitting its services between cloud providers. In doing this, the system makes it more difficult for all of the information to be retrieved to link personal data or understand what the user is doing that may be necessary to keep private. Likewise, the user can split multiple systems over a variety of cloud providers to prevent those systems from linking together data that should not otherwise be linked together. For example, if a user has a pervasive system for assistive health care, the user would likely not want that information available to other systems or people who do not have a need to know that information.

III. Security Risks

In this section we discuss the location of vulnerabilities in the cloud-pervasive system. These vulnerabilities exist on the cloud itself (in terms of processing in storage), during the migration to and from the cloud, and in the processing and storage in the pervasive system. Figure 1 demonstrates the location of these vulnerabilities in the interaction of a cloud-based pervasive system.

Security of cloud-based pervasive systems has been considered by [12]. The authors focus on augmenting mobile devices with elastic resources from the cloud. The authors propose an approach that involves using trustworthy containers, authentication, secure session management, logging/auditing, authorization, and access control. As the authors mention, this constitutes a first step toward securing their system. We agree with their approach, but in this section we augment with several alternative methods of accomplishing a secure cloud-based pervasive system that should be considered depending on the threat model and the constraints and purposes of the system itself.

A. On the Cloud

Most commonly considered threats in cloud computing focus on security of information while on the cloud. As a result, we will not discuss these threats in detail, but instead will just provide a brief overview. The main concerns in this area include the privacy of information stored on the cloud, the execution of correct code on the cloud, and availability of information stored on the cloud.

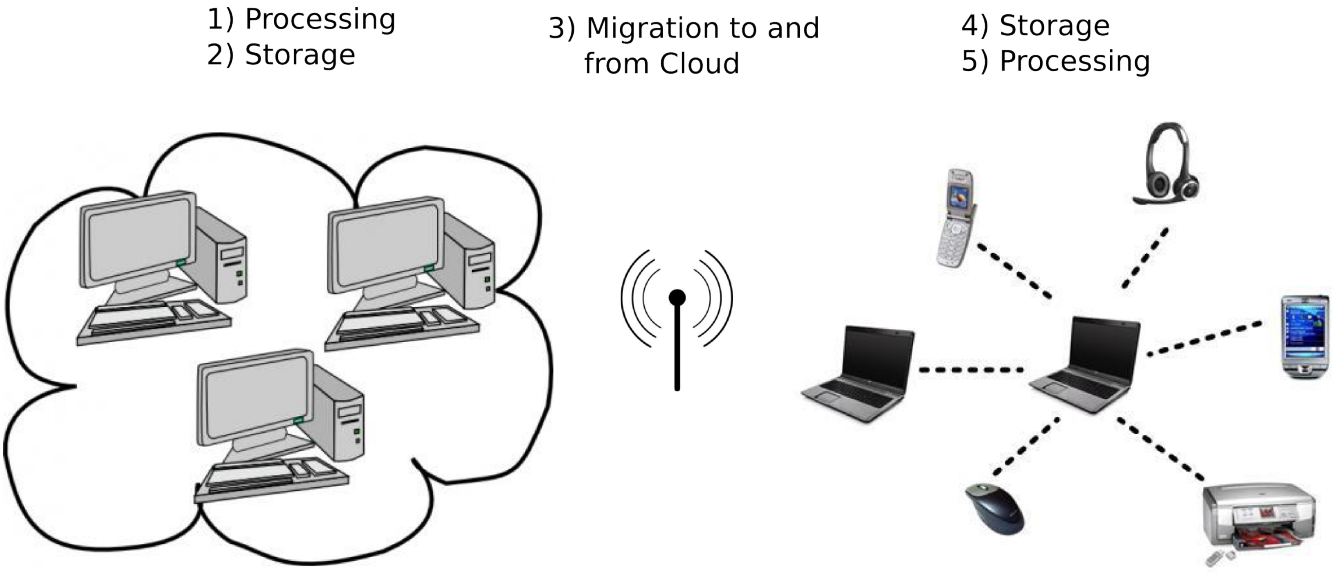

Fig. 1: Locations of Vulnerabilities Between the Cloud and Pervasive Systems

Privacy is a concern since users are storing information that could be used against them on a third party's server. Encryption is one possible solution to this problem, but for cloud applications to utilize the data that is stored, they must have an unencrypted version available. As a result, the cloud provider typically controls the key to encrypted data that is stored on the cloud. This approach relies on the cloud provider being trustworthy and reliable. If there is a security breach on the cloud, such as a malicious insider, then this assumption could lead to significant damage. Another solution that should be considered is the adoption of homomorphic encryption techniques. Homomorphic encryption would enable the cloud to perform operations on the data without ever having to see what is contained in the data; however, homomorphic encryption operations are computationally intense, so it may not be feasible to do this for all information. It may be acceptable to strategically encrypt only specific pieces of information that would drastically reduce the overall utility to the attacker if they were to obtain the information stored on the cloud.

Furthermore, an attacker (especially if a cloud was malicious) could mine easily mine the data collected from a variety of pervasive systems. This is especially dangerous if these systems are utilizing context data to make decisions. This scenario is very likely in a cloud situation where the cloud provides more specialized services (such as activity recognition from context information) rather than a more generalized cloud system. In Figure 2 we demonstrate one possible privacy violation where the cloud receives information to process from a variety of sources and pieces together that information to learn something it should not know. For example, a person's entertainment pervasive system may be kept in isolation from their healthcare pervasive system, but if a cloud (or colluding clouds) were able to link the information through usage patterns or some other technique, confidential information could be leaked to the attacker.

Another issue that may arise with information and services provided by the cloud is the execution of code on the cloud. The functionality and security of operational software can be compromised at any time. Global software supply chains provide opportunities to insert malicious content during development, as does insider access during maintenance and operation. As a result, software must be continually revalidated to maintain assurance of its validity. The problem is compounded by streaming executables that can quickly distribute malicious payloads in cloud computing networks. Current methods are insufficient to deal with the required scope and frequency of validation. The best testing processes can do no more than sample massive populations of possible executions, and most executions remaining untested when software enters operational use. Malicious content triggered by obscure input conditions, for example, specific times or coordinates, will likely escape detection in testing. Syntactic scanning of code depends on pre-defined syntactic signatures. Scanning cannot find problems for which no signatures exist, and is easily thwarted by simple obfuscation techniques. As a result, fast automated validation that can be applied in dynamic cloud computing environments is needed.

Availability of services and data is another problem that can arise from the remoteness of the cloud. A common solution of replication is possible (both by the clients and the clouds), but the result can lead to more issues of data management to ensure consistency. The same is true for services that are provided by the cloud. Services can be updated and improved over time, which could lead to inconsistent results or failures if a redundant, but older service is used when a newer one is

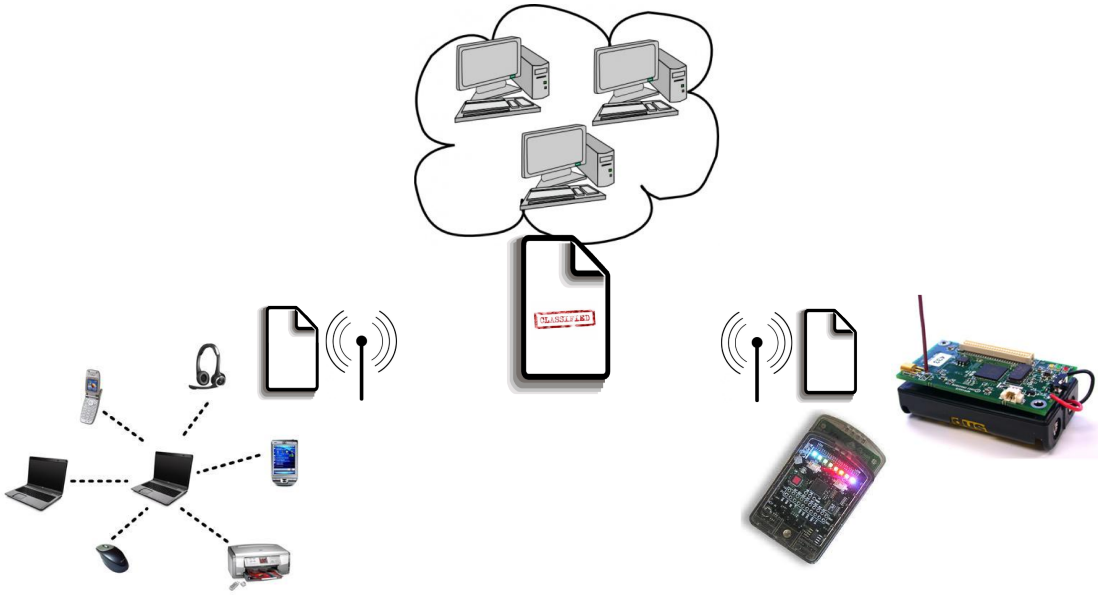

Fig. 2: A Cloud Combining Information from Multiple Pervasive Systems

expected by the pervasive application.

B. Migration to Pervasive Systems

During migration of data to pervasive systems, data is susceptible to attacks such as one might expect when transmitting data: Denial of service of data transmitted over the channel, Data manipulation (insertion, deletion, modification), and Eavesdropping.

In the case of a denial of service attack, there is often little that can be done to prevent it. One mitigating solution is to used multi-homed devices. It does not prevent an attacker from preventing the user from accessing necessary data, but it can greatly increase the cost. The attacker either has to launch the DoS attack closer in the network to the cloud (which presumably has greater bandwidth and other resources) or the attacker must launch an attack on all the communication channel that the pervasive system can use to access the cloud.

Data manipulation is another problem that can arise when transferring data to and from the cloud. The simple approach is to encrypt and/or sign all data that is being transmitted back and forth; however encryption, even symmetric key encryption, does add an additional burden to the CPU and battery. Symmetric key encryption also results in two points of vulnerability where the key could be stolen by an attacker (or attacker's malware). Further, encryption and other mechanisms that may be added to secure transmissions must not add a significant overhead to the latency. [13], [14] discuss the effect of latency on user experience, specifically users of thin clients.

The problems related to eavesdropping are very similar to that of data manipulation and can generally be approached through encryption-based solutions; however, even if the data is encrypted, some information can still be determined. It has been shown in wireless medical devices [15] that much of the information that is encrypted can be deduced from information about the size and frequency of the packets. It may be possible to apply similar techniques by eavesdropping certain pervasive systems as they communicate with the cloud.

C. Storage and Execution on Pervasive Systems

Cloud systems can provide further benefit to mobile users through pre-distributing content to the users during off-peak times. Users have a tendency to charge their mobile devices such as smart phones and laptops overnight and with the trend to a smart power grid, we expect them to continue to do so given the lower cost of energy during off-peak hours. This fact can be leveraged to predictively pre-distribute content to mobile users rather than streaming it from the cloud as it normally would be distributed. By pre-distributing content while mobile devices are not using energy from the battery, we believe we can greatly extend the life of mobile devices. For example, ISPs can provide a service to acquire data that is commonly accessed by the user, such as morning news websites or music, and push them to the user's mobile device shortly before their alarm goes off. Further, the user's mobile device could contact the ISP's cloud services any time it is plugged into a power source and opportunistically utilize cloud services and acquire data to extend battery life. As a result, information that would normally be the responsibility of the cloud would then be available on the mobile system where it could be tampered with by malware.

Malware has long affected desktops and laptops, but since 2004, it has begun to spread onto smartphones [16]. This spread of malware to smartphones and other devices running the same operating systems as these vulnerable devices is a threat to the security of cloud systems used for pervasive

computing. It is well-known that malware can record data and steal passwords, so obviously any encryption keys that may be used in the interaction between cloud and mobile systems are at risk if malware exists anywhere in the pervasive system. Approaches such as the use of one-time keys can be used to reduce the risk, but most of these approaches go against the vision of pervasive computing [17].

In Alice's case from the earlier scenario, her data and applications were fetched for her so that she could continue working while disconnected from the cloud. If Alice is using a computer with malware on it (a recent study by AV Comparatives [18] shows that no anti-virus software is able to catch all malware), that malware could easily modify her data from the cloud either before she sees it, or more likely, before she commits it back to the cloud. As mentioned earlier, malware on Alice's pervasive system could also steal her keys that she uses to access encrypted data on her cloud.

IV. Conclusion

In this paper, we have presented challenges associated with securing the interaction of pervasive computing systems and cloud systems. There is much to be gained by pairing the two, but in doing so, the vulnerabilities of losing control of the applications and data of both is increased. Many existing solutions can be leveraged to make attacking these systems more difficult, but much work remains to create a holistic solution.

Acknowledgements

This work was sponsored by a contractor of the United States Government under contract DE-AC05-00OR22725 with the United States Department of Energy. The United States Government retains, and the publisher, by accepting this submission for publication, acknowledges that the United States Government retains, a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this submission, or allow others to do so, for United States Government purposes.

References

[1] A. P. Miettinen and J. K. Nurminen, "Energy efficiency of mobile clients in cloud computing," in *Proceedings of the 2nd USENIX conference on Hot topics in cloud computing*, ser. HotCloud'10. Berkeley, CA, USA: USENIX Association, 2010, pp. 4–4. [Online]. Available: http://portal.acm.org/citation.cfm?id=1863103.1863107

[2] E. Cuervo, A. Balasubramanian, D.-k. Cho, A. Wolman, S. Saroiu, R. Chandra, and P. Bahl, "Maui: making smartphones last longer with code offload," in *Proceedings of the 8th international conference on Mobile systems, applications, and services*, ser. MobiSys '10. New York, NY, USA: ACM, 2010, pp. 49–62. [Online]. Available: http://doi.acm.org/10.1145/1814433.1814441

[3] B.-G. Chun and P. Maniatis, "Augmented smartphone applications through clone cloud execution," in *Proceedings of the 12th conference on Hot topics in operating systems*. Berkeley, CA, USA: USENIX Association, 2009.

[4] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for vm-based cloudlets in mobile computing," *IEEE Pervasive Computing*, vol. 8, pp. 14–23, October 2009. [Online]. Available: http://portal.acm.org/citation.cfm?id=1638591.1638731

[5] A. Anand, A. Gember, A. Akella, and V. Sekar, "Tracking semantic relationships for effective data management in home networks," in *Proceedings of the 2010 ACM SIGCOMM workshop on Home networks*, ser. HomeNets '10. New York, NY, USA: ACM, 2010, pp. 49–54. [Online]. Available: http://doi.acm.org/10.1145/1851307.1851320

[6] Y. Luo, K. T. Lam, and C.-L. Wang, "Path-analytic distributed object prefetching," *Parallel Architectures, Algorithms, and Networks, International Symposium on*, vol. 0, pp. 98–103, 2009.

[7] N. J. Tuah, M. Kumar, and S. Venkatesh, "Resource-aware speculative prefetching in wireless networks," *Wirel. Netw.*, vol. 9, pp. 61–72, January 2003. [Online]. Available: http://dx.doi.org/10.1023/A:1020829124143

[8] H. Shen, M. Kumar, S. K. Das, and Z. Wang, "Energy-efficient data caching and prefetching for mobile devices based on utility," *Mob. Netw. Appl.*, vol. 10, pp. 475–486, August 2005. [Online]. Available: http://dx.doi.org/10.1145/1160162.1160171

[9] Y. Huang, J. Cao, Z. Wang, B. Jin, and Y. Feng, "Achieving flexible cache consistency for pervasive internet access," *Pervasive Computing and Communications, IEEE International Conference on*, vol. 0, pp. 239–250, 2007.

[10] I. Carreras and D. Linner, "Self-evolving applications over opportunistic communication systems," in *Eigth Annual IEEE International Conference on Pervasive Computing and Communications, PerCom 2010, March 29 - April 2, 2010, Mannheim, Germany, Workshop Proceedings*, 2010, pp. 153–158.

[11] A. Heinemann, J. Kangasharju, and M. Muehlhaeuser, "Opportunistic data dissemination using real-world user mobility traces," in *Proceedings of the 22nd International Conference on Advanced Information Networking and Applications - Workshops*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 1715–1720. [Online]. Available: http://portal.acm.org/citation.cfm?id=1395080.1395410

[12] X. Zhang, J. Schiffman, S. Gibbs, A. Kunjithapatham, and S. Jeong, "Securing elastic applications on mobile devices for cloud computing," in *Proceedings of the 2009 ACM workshop on Cloud computing security*, ser. CCSW '09. New York, NY, USA: ACM, 2009, pp. 127–134. [Online]. Available: http://doi.acm.org/10.1145/1655008.1655026

[13] H. A. Lagar-Cavilla, N. Tolia, E. De Lara, M. Satyanarayanan, and D. O'Hallaron, "Interactive resource-intensive applications made easy," in *Proceedings of the 8th ACM/IFIP/USENIX international conference on Middleware*, ser. MIDDLEWARE2007. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 143–163. [Online]. Available: http://portal.acm.org/citation.cfm?id=1785080.1785091

[14] N. Tolia, D. G. Andersen, and M. Satyanarayanan, "Quantifying interactive user experience on thin clients," *Computer*, vol. 39, pp. 46–52, March 2006. [Online]. Available: http://portal.acm.org/citation.cfm?id=1128587.1128640

[15] M. Salajegheh, A. Molina, and K. Fu, "Privacy of home telemedicine: Encryption is not enough," *Journal of Medical Devices*, vol. 3, no. 2, April 2009, design of Medical Devices Conference Abstracts. [Online]. Available: http://www.cs.umass.edu/~kevinfu/papers/salajegheh-DMD09-abstract.pdf

[16] A.-D. Schmidt, H.-G. Schmidt, L. Batyuk, J. Clausen, S. Camtepe, S. Albayrak, and C. Yildizli, "Smartphone malware evolution revisited: Android next target?" in *Malicious and Unwanted Software (MALWARE), 2009 4th International Conference on*, 2009, pp. 1 –7.

[17] M. Weiser, "The computer for the 21st century," *Scientific American*, vol. 265, no. 3, pp. 66–75, January 1991. [Online]. Available: http://www.ubiq.com/hypertext/weiser/SciAmDraft3.html

[18] A. Comparatives, "On-demand detection of malicious software," *Antivirus Comparatives*, vol. 15, February 2010.