CSC574 – Fall 2011 – Project Proposal

Ashwin Shashidharan and Jitesh Shah
{ashashi3,jhshah}@ncsu.edu

September 7, 2011

1 Offloading encryption/decryption of data/image to the cloud

1.1 Abstract

For the cloud to be feasible, it is important for the clients to be able to verify that the image they are booting into isn't tampered with and that their data is safe in the cloud. If the cloud provider is not to be trusted, this means that the virtual image needs to be encrypted and the key stored with the client rather than the cloud provider. Also, the data in the cloud needs to be encrypted. We intend to address this problem using SSL to authenticate the server and then establishing a secure channel to exchange keys for decrypting the virtual image and a different key to secure sensitive data. The final result is offloading encryption/decryption to the cloud rather than doing it on the client-side. As far as the implementation goes, our initial plan is to use Eucalyptus or the VCL sandbox to mimic a cloud environment to work on.
Note: We are not clear about the exact scope of the project yet. We can't judge its difficulty level yet.

1.2 Experience

Both team members have been involved in the study of the VCL cloud computing environment at NCSU. In addition, Ashwin has experience setting up certificates for web servers and PHP/PERL application development in the cloud. Jitesh has worked on low-level Ring-0 VMMs.

1.3 Expected Work Breakdown

Jitesh and Ashwin will together be involved in developing the above mentioned authority on a node in the cloud. Ashwin will be responsible for setting up and verifying the certificates and the key-exchange over SSL. Jitesh will be responsible for encryption/decryption of the kernel image as well as the data.

2 Analysis of the Android browser for typical browser attacks

2.1 Abstract

Smart-phones have come a long way. Attacking the Android browser is effective to extract a lot of sensitive details - contacts, username/passwords, etc. This project aims at studying how vulnerable the Android browser is by trying to mount as many known attacks as possible - tapjacking, CSRF, XSS, etc. We'll also try to implement defence mechanisms for vulnerabilities we find in the Android browser. Our take-away from this project is a comprehensive knowledge of attack vectors on a browser and ways to defend against them. Eventually, our result will be a list of known vulnerabilities that the Android browser should fix (and possibly the actual fixes too)

2.2 Experience

Both the team members know how to pull APKs from a android phone and how to decompile them. Both of them can set-up an Android development environment but neither has done any serious work on Android/Java.

2.3 Expected Work Breakdown

We intend to divide-up the attack types among ourselves.

3 Designing a VMM-based Honeypot for a cloud environment

3.1 Abstract

Our aim through this project is to collect information about attacks on honeypots using a VMM-based monitor implementation by eliminating limitations of monitoring honeypots from outside the honeypot. To this effect, we propose inclusion of software hooks in the VMM to directly intercept and trigger recording of suspicious events. With the help of the VMM, attacks on the honeypot kernels may be monitored and binary rewriting may be carried out in an ascertained manner. The controller in the cloud environment can enable or disable these hooks to allow for specific monitoring details. Through this implementation we hope to identify and study attacks in a cloud environment. Furthermore, cloud operators may use the results to protect their setup from such attacks and study the actions of an attacker following a compromise.

3.2 Experience

Both team members have been involved with VM management in the VCL cloud computing environment at NCSU. Jitesh has worked on low-level ring-0 VMMs, specifically, sharing memory pages among co-operating VMs. Ashwin has experience with configuring and managing Virtual Machines.

3.3 Expected Work Breakdown

Initially, we plan to identify open source virtualization platforms and study their architecture. Later on, Jitesh will be involved in interfacing new event handlers in the VMM with the OS, while Ashwin will be involved in writing software for the hooks that will interact with the application layer.