Francois D'Ugard

[Draw your reader in with an engaging abstract. It is typically a short summary of the document. When you’re ready to add your content, just click here and start typing.]

MISSION CRITICAL CLOUD COMPUTING

Feasibility Study and Project Plan

# Abstract

Today’s public and private clouds are areas of great interest for both research and production purposes as they give users on-demand access to virtually unlimited pools of computational resources at a cost proportional to their actual consumption. In turn, clouds providers are able to maximize the utilization of their hardware investment by accommodating the virtual machines (VMs) of several customers on the same physical infrastructure. However, this approach lowers the difficulty of collocating a malicious VM with its target VM in order to extract information or disrupt its functioning in some way.

In this document we evaluate the feasibility of several alternatives for the implementation of the proposed system and provide a high-level description of our plan to make it a reality. We propose an approach to defend mission-critical VMs by realizing a moving-target defense (MTD) strategy, where VMs are frequently and randomly migrated across physical hosts within a cloud. The purpose of this system is to reduce the possibility of a malicious VM locating the mission-critical VM and achieving voluntary co-residency with it, or reduce the attacker’s opportunity to cause harm by limiting the co-residency time if it is ever achieved. Two key difficulties in implementing this approach are achieving a dynamic and unpredictable migration policy, and keeping the network connectivity between the migrating VM and its peers after the migration completes.

# Introduction

Virtualization is an increasingly popular approach to manage rising information technology costs and complexity in every sector of the economy. Cloud computing allows organizations of any size to provision infrastructure resources as needed and flexibly scale technology resources to meet changing demands. IaaS providers like Amazon, Microsoft, or Rackspace pool hardware resources such as compute, memory, and storage and allocate them based on the provisioning requests of their users and available resources. In order to efficiently and cost effectively allocate these resources providers must necessarily aggregate users of different requirements and workloads onto the same physical infrastructure.

The rising popularity and diffusion of hardware virtualization technology among organizations of every size has led researchers and technology professionals to seriously consider the security of such computer systems. By allocating the virtual machines of different user onto the same infrastructure various security vulnerabilities are created. However many of the unique capabilities of cloud computing can also be used to increase the reliability of the cloud as well as ensure its security and resilience to an attack.

An important quality of virtualization is the ability for users to quickly provision resources as needed. This capability lends itself well to the concept of migration wherein a virtual machine is suspended or shutdown and its resources are reallocated to another physical machine and then restarted or rebooted. This capability can be used to ensure the security of a cloud computing environment.

## Problem Definition

The rising popularity of cloud computing concepts has introduced significant security vulnerabilities into the information technology infrastructure of many organizations. Currently virtualized information technology systems are designed and built to operate on relatively static configurations. Typically cloud services providers fulfill provisioning requests on an as needed basis. In order for cloud computing to remain cost effective infrastructure providers must provision pools of resources, such as CPU, memory, and storage among many different users.

This configuration means that a malicious user can reside alongside a virtual machine that computes mission critical or sensitive data. The sharing of physical infrastructure among virtual machines belonging to different users opens up the possibility of a side-channel attack. A side channel attack occurs when a malicious users is able to locate a target virtual machine and spawn another virtual machine alongside this target. The malicious user is then able to extract information by monitoring the hardware resources shared by both virtual machines.

## 1.2. Background

## 1.3. Definitions, Acronyms, and Abbreviations

API - Application Programming Interface: specifies how some software components should interact with each other [16].

Cloud computing: this phrase commonly refers to network-based services, which appear to be provided by real server hardware, and are in fact served up by virtual hardware that is simulated by software running on one or more physical machines [17].

Co-residency: a VM is co-resident with any VM when they are running on the same physical machine, and describes a great security risk when hosting VMs with sensitive data.

Cryptosystem: shorthand for "cryptographic system," which is any computer system that involves cryptography [18].

IP Address - Internet Protocol address: a numerical label assigned to each device participating in a computer network that uses the Internet Protocol for communication [19].

Hardware virtualization: this term refers to the creation of a virtual machine that acts like a real computer running an operating system [26].

LAN - Local Area Network: a computer network that interconnects computers in a limited area such as a home or a school using network media [21].

Live virtual machine migration: the process of moving a running virtual machine from a physical host to another physical host [6].

MTD - Moving-Target Defense: an approach that has been proposed to better protect important network systems and critical computing infrastructure by dynamically changing properties of their configuration in some way [1].

Non-live virtual machine migration: the process of moving a powered off virtual machine from a physical host to another physical host. After the transfer completes, the migrated virtual machine is restarted [6].

P2P network - Peer-to-peer network: a type of decentralized and distributed network architecture in which individual nodes in the network (called "peers") act as both suppliers and consumers of resources [22].

Private cloud: it is a cloud infrastructure operated solely for a single organization. It can be managed internally or by a third-party, and hosted internally or externally [17].

Public cloud: a cloud is said to be public when the services are rendered over a network that is open for public use, most commonly the Internet [17].

Side-channel attack: any attack based on information gained from the physical implementation of a cryptosystem, rather than brute force or theoretical weaknesses in the algorithms [23].

UML - Unified Modeling language: a standardized, general-purpose modeling language in the field of software engineering. It includes a set of graphic notation techniques to create visual models of object-oriented software-intensive systems [25].

Virtual cluster: a group of VMs configured for a common purpose with associated storage resource, operating system, software environment, communication protocol, and network configuration [12].

VM - Virtual Machine: a software-based emulation of a physical computer [27].

VMM - Virtual Machine Monitor or hypervisor: a piece of computer software, firmware, or hardware that creates and runs virtual machines [20].

Virtual network: a computer network that consists of virtual network links as opposed to physical (wired or wireless) links between connected devices. It is implemented using methods of network virtualization [28].

VPN - Virtual Private Network: A VPN extends a private network across a public network, such as the Internet. It enables a computer to send and receive data across shared or public networks as if it were directly connected to the private network, while benefiting from the functionality, security and management policies of the private network [29].

## Overview of Document

This document is organized into various different sections.

Chapter 1 is an introductory section. Section 1.1 defines and describes the problem we will solve. 1.2 provides some background information related to the problem including previous research. Section 1.3 we define and explain various terms and abbreviations found in this document.

Chapter 2 is a study into the feasibility of the proposed system. Section 2.1 describes the current system in place and identifies its current limitations and constraints. Section 2.2 presents the purpose of the system in detail. Section 2.3 is an overview of the user requirements that must be met upon the successful completion of the project. Based on those requirements section 2.4 compares other possible solutions and based on this analysis section 2.5 presents our recommended solution.

Chapter 3 is the project plan. The project plan describes the organization of the project in section 3.1 Also included in section 3.1.1 is the project personnel organization and hard ware and software resources in section 3.1.2. Lastly section 3.2 clearly states that tasks, milestones and deliverables expected upon completion of the project.

Chapter 4 is the appendix. It includes a project schedule in section 4.1. The feasibility matrix in section 4.2 a cost matrix section 4.3 and a diary of meetings in section 4.4

Finally chapter 5 is a bibliographic reference of any works cited in this document.

# Feasibility Study

The following section analyzes the current system and compares the feasibility of our proposed system considering the current system as well as other alternative solutions. Section 2.1 is a description of the current system, afterward section 2.2 provides an overview of the proposed system. Section 2.3 outlines the user’s requirements of a new system. Section 2.4 outlines alternative solutions to the problem. Section 2.5 finalizes our recommendations for a new system based on the information gathered in the previous sections.

## 2.1. Description of Current System

Currently virtualized information technology systems are designed and built to operate on static configurations. Typically cloud services providers fulfill provisioning requests as requested by their user base. In order for cloud computing to remain cost effective infrastructure providers must provision pools of resources, such as CPU, memory, and storage among many different users. This policy, while cost effective and efficient, opens up the cloud to the possibility of a side channel attack. By mapping the internal structure of the cloud environment a malicious attacker is able to identify a target virtual machine a spawn a malicious vm co-resident to the target. This allows the malicious vm to monitor the shared physical infrastructure and extract sensitive or mission critical data.

Existing Virtual Machine Monitors implement both live and no live migration techniques. However, these capabilities are usually intended to facilitate load balancing by moving workloads from one physical host to another to improve performance, or to serve as a maintenance tool, by relocating a hosts virtual machines in order to perform physical repairs. No current virtual machine monitors implement a live migration policy for the purpose of making the cloud more resilient or to improve the security of the cloud.

## 2.2. Purpose of New System

This project will deliver mission assurance to mission-critical applications in cloud computing systems. We will do so by leveraging the unique capabilities that develop a virtual machine based approach to run applications with good security and reliability in typical cloud computing systems. This project will build upon the previous project's results:

1) A VM management system that dynamically migrate VMs across hosts on an OpenStack-based cloud platform;

2) A P2P overlay network that interconnect the OpenStack VMs based on the IP-over-P2P (IPOP) framework. The project will focus on developing an extension to IPOP that will allow for the communications among the VMs to be routed by an overlay network in an OpenStack-based cloud system.

This project will implement an application-level peer to peer virtual overlay network that interconnects each VM in the mission critical cloud. Additionally this network will be

## 2.3. High-level Definition of User Requirements

This section describes the requirements that the customer wants to include in the final software product and the constraints on its operation. The client cannot demand features outside the following list and the development team cannot claim completion of the project if it does not satisfy any of the items. The development team reserves the right to consider and negotiate additional features requested by the client.

* The system shall allow a user to dynamically migrate a VM across many hosts in an Openstack based cloud platform.
* The system shall allow the user to list all VM’s with the service enabled
* The system shall allow a user to enable the migration service for all vm’s
* The system shall allow a user to enable the migration service for a subset of vm’s
* The system shall allow a user to disable the migration service for all vm’s
* The system shall allow a user to disable the migration service for a subset of vm’s
* The system shall allow a user to create a P2P overlay network
* The system shall allow a user to destroy an overlay network
* The system shall allow a user to connect a vm to an existing P2P overlay network
* The system shall allow a user to disconnect a vm from a P2P overlay network
* The system shall expose both the migration and network services via a command line interface.
* The system shall allow a user to specify a configuration file that species all parameters for the overlay network.
* The system shall allow the user to specify a migration time range.
* The system shall schedule and execute migrations at random intervals based on the parameters specified by the user.
* The system shall allow a user to start, stop and restart the migration service at any time.
* The system shall support the execution of applications that use TCP and UDP protocols.
* The system shall prioritize the reconnection process with peers based on the traffic history.
* The system shall not constrain network infrastructure in order to function properly.

## 2.4. Alternative Solutions

This section provides an overview of existing alternative solutions for the components of our project. The goal of this project is to protect Mission critical virtual machines processing confidential or sensitive data. Approaches that limit the attacker’s ability to successful place a malicious virtual machine co-resident with their target virtual machine or monitor network traffic between co-resident virtual machines will be considered. A brief description of the project problems and background information are included bellow.

1. Migration Management
   1. Manual Migration
   2. Proactive random and automatic migration
2. Communication Network
   1. Centralized host communication server
   2. VPN P2P Overlay Network (IPOP)

### Description of Alternatives

This section provides a high level description of the proposed alternative solutions

Migration Management

Both Policies considered achieve moving target defense requirements.

A1. Manual Migration

Although manual migration is essential no implementation and therefore easy to implement.

A2. Proactive random automatic migration

Communication Network

### Selection Criteria

We consider three criteria for the selection of project solutions they are

Operational Feasibility

Technical Feasibility

Economic Feasibility

### 2.4.3. Analysis of Alternatives

## 2.5. Recommendations

Based on the quantitative study outlined in the previous sections it is the recommendation of this document to proceed with design and implementation of a proactive random and automatic migration management tool as well as a VPN overlay communication network. These two approaches will best guarantee the security and resilience of a mission critical cloud computing system. The proactive random and automatic migration policy will reduce the likelihood that a malicious cloud user will by able to accurately map the structure of the cloud and successful spawn a virtual machine co-resident to their target. Additionally the use of a VPN overlay communication network will allow any migrating virtual machines to preserve communication and collaboration among other friendly virtual machines while limiting the snooping ability of co-resident malicious virtual machines.

# Project Plan

This chapter presents the project organization including the personnel assigned to the project, and the mentor and client. We also specify the hardware and software resources required to complete the project. Lastly we identify the tasks, milestones and deliverables of our project and include a timeline of their completion dates.

## 3.1. Project Organization

All tasks are identified with a unique string in the format of component number, milestone number, and task number i.e. C#M#T#.

* Setup Development Environment
  + Milestone 1: Complete setup of development environment and network configuration and setup NFS file system to support live migration capabilities of Openstack.
    - C1M1T1: Install Hardware Compenets
    - C1M1T2: Install Ubuntu 12.04 on all development machines
    - C1M1T3: Configure Development Network
    - C1M1T4: Deploy Openstack cloud to all development machines
    - C1M1T5: Setup Network File sharing service for live migration
* Migration Manager
  + Milestone 1: Design Migration Manager Component
    - C2M1T1: Design Migration Manager algorithm
  + Milestone 2: Implement Migration Manager Component
    - C2M2T1: Implement Migration Manager Algorithm
    - C2M2T2: Implement Command line interface for migration management component
  + Milestone 3: Integrate Migration Manager Component into Openstack system
    - C2M3T1: Systems integration and testing.
* Communications Network: Design and implementation of a distributed P2P VPN communications network that obfuscates communications with collalborating vitrutal machines and prioritizes communication reestablishment by communication history.
  + Milestone 1: Design distributed collaborative communication network using IPOP P2P VPN.
    - C3M1T1: Design modifications to IPOP controller
    - C3M1T2: Design link restoration algorithm based on communications history
    - C3M1T2: Design communications obfuscation algorithm to hide mission critical communications between collaborating virtual machines.
  + Milestone 2: Implement Communications Network Component
    - C3M2T1: Implement Communications Network link reestablishment Algorithm
    - C3M2T2: Implement Communications Network obfuscation Algorithm
    - C3M2T3: Implement Command line interface for Communications Network component
  + Milestone 3: Integrate Communications Network Component into Openstack system
    - C3M3T1: Systems integration and testing.

Project Milestones

1. Complete setup of development environment and network configuration and setup NFS file system to support live migration capabilities of Openstack.
2. System accepts input parameters (VMs, migration frequency boundaries,...) for the migrations.
3. Successful autonomous and random migrations within given parameters.
4. Working P2P system through which VMs communicate and retrieve network information from peers.
5. Working attacker system.
6. Successful MTD system.

## 3.1.1. Project Personnel Organization

Francois D’Ugard: Lead Developer, Tester and Time Keeper for the VM Migration Management and Virtual Private Network of P2P overlay work domains.

Dr. Ming Zhao: Mentor and Client

## 3.1.2. Hardware and Software Resources

The following section lists the hardware and software resources necessary for the successful completion of the project.

* Hardware
  + 3 personal computers for development, testing, and demos. Minimum requirements: Dual-core CPU @ 1.2 GHz, 2GB RAM, 128GB HDD.
  + 3+ servers where the cloud infrastructure will be set up. Minimum requirements: Dual-core CPU @ 1.4 GHz, 8GB RAM, 500 GB HDD, 2 x 1Gbps PCI LAN cards.
* Software
  + OpenStack cloud computing platform.
  + Python 2.7
  + Ubuntu Server 12.04 or 14.04 LTS.
  + KVM Virtual Machine Manager.
  + Google Drive, Google Talk, and Thunderbird or any other mail client for collaboration.
  + Microsoft Word, Microsoft Excel, Microsoft PowerPoint, and Adobe Reader
  + StarUML - UML modeling tool.
  + Netbeans – IDE
  + Unittest - a unit test framework for Python.
  + Pencil for user interface mockups.

## 3.2. Identification of Tasks, Milestones, and Deliverables

## 

# 4. Appendix

## 4.1. Appendix A - Project Schedule

## 4.2. Appendix B - Feasibility Matrix

## 4.3. Appendix C - Cost Matrix

## 4.4. Appendix D - Diary of Meetings

# 5. References

http://www.dhs.gov/csd-mtd