Computers and Electrical Engineering 000 (2017) 1–13

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

A generic framework for application streaming service☆

Asad W. Malik[a,b,*], Raihan ur Rasool[a,b,c], Zahid Anwar[a,b,d], Shahid Nawaz[a,b]

[a] School of Electrical Engineering and Computer Science (SEECS), Pakistan
[b] National University of Sciences and Technology (NUST), Pakistan
[c] Victoria University, Melbourne, Australia
[d] Fontbonne University, USA

ARTICLE INFO

Article history:
Received 19 April 2017
Revised 26 September 2017
Accepted 26 September 2017
Available online xxx

Keywords:
Streaming
Virtualization
Virtual machines
Virtual appliance
Cloud Computing

ABSTRACT

The Cloud provides a shared computing environment to execute complex and compute intensive virtual appliances (VA). With network function virtualization and software defined networks, we are expected to see the physical network devices and their management consoles as VAs streamed on demand. Here, our focus is on the download mechanism that is used to distribute software that requires complex installation and rigid dependencies on the operating system. With limited bandwidth, it is difficult to download several pieces of software, especially with slow or sporadic internet connectivity. In this paper, we present a novel framework for application streaming named as Ceaseless Virtual Appliance Streaming (CVAS) system that offers virtual machine streaming through small executable blocks. The user can use a VA while it is being downloaded. The experimental section demonstrates the effectiveness of the proposed framework in terms of buffering time to execute virtual appliances. A feature analysis in comparison to competing approaches demonstrates that CVAS has improved buffering time, responsiveness and state pre-fetching accuracy.

1. Introduction

Virtual machines (VMs) are executed by virtual machine monitors or hypervisors from onboard VM images. With the combination of host-level virtualization and storage area network facilities, Cloud Computing offers an extensive variety of new applications for the VM technology. The virtualized public and private clouds facilitate organizations to transfer their operation from their dedicated hardware to the shared infrastructure having virtual servers. This allows proficient hardware utilization, simple management, and faster disaster and failure recovery. The users can access this preconfigured software via the internet or they can download to work locally. In the latter case, the user has to download the complete VM before user can start to use it. At the same time, downloading a VM instance may result in network congestion. The foremost impediment in the popularity of virtual machines is the unsupported streaming facility. Therefore, it is necessary to devise a mechanism for the management and transmission of the virtual machines in a similar fashion as the video file. In this way, with the transmission of a smaller chunk of virtual machine, the user may be able to begin the execution of the virtual machine as it is being downloaded. Further, according to the principle of factor scarcity, eighty percent of the users utilize only twenty percent functionality of the software. Therefore, it is a waste of the expensive network resources if virtual

☆ Reviews processed and recommended for publication to the Editor-in-Chief by Associate Editor Dr. Z. Arnavut.

* Corresponding author.
E-mail addresses: asad.malik@seecs.edu.pk (A.W. Malik), raihan.rasool@live.vu.edu.au (R.u. Rasool), zanwar@fontbonne.edu (Z. Anwar), 12msitsnawaz@seecs.edu.pk (S. Nawaz).

https://doi.org/10.1016/j.compeleceng.2017.09.029

machines are to be transmitted. It is also important that the streaming server predicts the subsequent function or chunk of a virtual appliance (VA) that can be required by the user at a later stage.

Virtualization means the creation of a virtual version of operating systems, hardware platforms, storage devices, and network resources. A virtual machine, as the vertebral column of virtualization, is a software emulation to facilitate the installation and operation of diverse types of operating systems as a guest over the host operating system. It relies upon a middleware virtualization layer called the hypervisor to allocate hardware resources to the guest operating system. The host system is transparent to the guest operating system (OS) and it functions in a similar fashion as it is directly installed on the hardware. A virtual machine aids the execution of those applications on computing devices which are inherently not compatible with the host operating system. Numerous virtual machines can be installed on a single host operating system provided there is enough storage capacity available. The prime advantage of the virtual machine is its security and platform independence. It can be executed on diverse types of computing devices regardless of the disparity in hardware architecture.

The VA is an image file of the virtual machine consisting of the pre-configured operating system and a single application [1]. It is not a complete virtual machine but an image file which is intended to run on a virtual machine having type 1 or type 2 hypervisor platforms. It usually hosts a single application and functions like an application but is managed as a virtual machine. The idea behind the concept of a virtual appliance is to create and deliver a ready-made package of an application and operating system for its easy installation and operation over non-compatible devices. Only the essential components of the operating system along with a single application are bundled in the form of an image file. As the necessary operating system components are embedded in the virtual appliance, it can operate on diverse types of architectures. For instance, a game developed for the Windows operating system can be played on Linux with the same ease. Virtual appliances can be categorized as closed and open. Closed virtual appliances are configured and distributed as indestructible units, whereas users have an option to modify open virtual appliances according to their requirements through a web page user interface.

In this paper, we adopted a mechanism of video streaming to facilitate end users having very slow Internet connections. In such a scenario, downloading a complete VM in order to use only a portion of it can take a significant amount of time. Therefore, we adopted a streaming technique for application streaming by converting it into small sized packets, where every packet can execute separately. In the performance evaluation section, we benchmark the application streaming concept and compared it with the traditional download mechanism.

The paper is divided into seven sections. Section II covers the state-of-the-art, section III presents motivation behind this work; section IV and V cover the proposed design and architecture. At the end, experimental evaluation and conclusion are presented.

2. Literature review

With technological advancement and easy availability of Cloud infrastructure, multimedia streaming is a widely used application over the intranet. To facilitate a large number of users, multimedia streaming applications require huge infrastructure to provide seamless services. The traditional client-server model is insufficient for multimedia services due to its bottleneck and a limited number of concurrent user's support. Therefore, researchers have adopted various mechanisms to handle this issue efficiently.

Hyun et al. [2] presented an automatic testing environment based on peer-to-peer multimedia streaming application. The proposed framework is based on various simulations and is available at GitHub. In Cloud Computing, energy is one of the major concerns that require different power optimization techniques. Ma et al. [3] proposed a power saving mechanism for multimedia streaming inside the cloud. The proposed technique is based on Dynamic Voltage and Frequency Scaling (DVFS) to dynamically adjust the power based on the state of a device. Through varying the frame rate, the overall power consumption is reduced by 1.5%.

The exponential growth of the mobile industry is also the reason for increased demand for multimedia objects. Due to significant demand and to reduce the delay, Hypertext Transfer Protocol (HTTP) adaptive streaming concept has been widely used to deliver video contents. In [4], Kim et al. presented a concept-aware prefetching scheme that is based on the HTTP adaptive model. The proposed scheme is designed for interactive multimedia streaming. The main objective of this model is to reduce the delay and improve the quality of experience. Usually streaming services such as on-demand television channels are provided through resource reservation, but still do not efficiently utilize the communication channels. One of the major challenges for multimedia streaming is latency and Round Trip Time (RTT). For good interactive applications, the latency should be less than 100 ms. Although many streaming protocols meet the 100 ms requirement, the users still face jitter effects due to buffer underflow. Therefore, techniques that can reduce latencies or jitter are still an open area for research [5].

In order to provide smooth multimedia services, a dedicated bandwidth is required that is not possible to meet over the Internet due to constantly fluctuating bandwidth and delay. Protocols such as the Transmission Flow Rate Control (TFRC) have been proposed for multimedia streaming. TFRC reacts less radically to congestion to achieve a better rate compared to transmission control protocol (TCP). TFRC is not well suited for multimedia streaming. The streaming works on the transmission rate provided by TFRC throughout its communication which is good but not optimal from an application's perspective. Schmidt et al. [5] presented the congestion control framework, utility-driver transmission flow rate control (UTFRC) that is designed for multimedia streaming. UTFRC is more media friendly and is based on TFRC.