Cloud Gaming and Containers

Essay for DD2482: Automated Software Testing and DevOps

Stefano Scolari
sscolari@kth.se

April 2023

I certify that generative AI, incl. ChatGPT, has not been used to write this essay. Using generative AI without permission is considered academic misconduct.

1 Introduction

Cloud gaming has experienced a surge in popularity over the last five years, with the market capitalization for *Cloud Gaming* valued at approximately $244M in 2020. According to the Global Cloud Gaming Report, its market capitalization was reported to be around $1.3B in 2022, and its growth is expected to reach $13B by 2028 [1]. Cloud gaming was developed based on the concept of *gaming on demand*, which enables users to play games anytime and anywhere, without worrying about computational power and download time. As games are becoming increasingly complex, the direct demand for Graphical Processing Units (GPUs) power has also increased. At a high level, the underlying mechanism for cloud gaming is relatively simple. Users connect to a gaming server that runs the game instance, and the input is captured in real-time from the end-user and sent to the server. This is then used as input for the game instance, and the next frame is computed, encoded, and sent back to the end-user's thin client. In essence, cloud gaming is an interactive video streaming service.

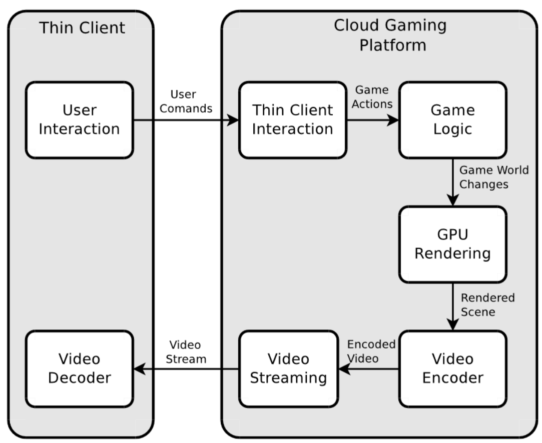

Figure 1: Cloud Gaming architecture [20]

Given that the game is rendered on a remote *Cloud Server*, *Cloud Gaming* services require the use of expensive and high-end hardware. GPUs need to be shared and orchestrated among multiple simultaneous users. Some GPUs can handle up to 30 simultaneous users [3]. Virtualization techniques are used to isolate and separate users' access and utilization of the underlying hardware.
This paper aims at comparing what type of server *Virtualization* technique is best suited for *Cloud Gaming*, as this field presents unique challenges and requirements. In particular, traditional *Virtual Machines* will be compared with *Containers*.

2 Virtualization in Cloud Gaming

Benchmarking has been performed on both Virtual Machines (VMs) and Containers, the results show that the use of VMs to virtualize game instances can result in performance overhead [24, 12]. Increasing awareness around the benefits of Containers over Virtual Machines could also help *less innovation-prone* and habitual sectors like the *Game Development* one, to more extensively use and implement this kind of technology [9]. As explained by Whitehead and Jantunen [2], Containers hold great potential to enhance, simplify and optimize the Game Development process. Furthermore, they offer a more scalable solution for handling *game servers* and *backend microservices* [5].
Traditionally, *Cloud Gaming* instances were booted through a *Hypervisor* used to generate hardware-level VMs. This made it possible to obtain isolation, while also having resource sharing between users.
The most important hardware component for *Cloud Gaming* is the GPU, and when accessed through a VM, performance levels struggle to equal native ones. `State-of-the-art` solutions for Cloud Gaming architectures, propose the use of container-based *lightweight virtualization techniques*. An example is `TG-SHARE` [8]. This novel technique results more cost-effective compared to techniques built around traditional Virtualization.

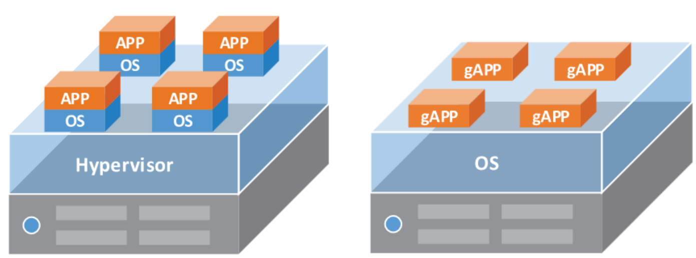

Figure 2: Traditional VMs (left) and Containers (right) compared. Note that *gApp* refers to a Gaming Application.

TG-SHARE also supports the *off-the-shelf* consumer GPUs, with no need to use the expensive proprietary `vGPU` technology [6]. Through further exploitation of the hardware capabilities per computing node, `TG-SHARE` is able to increase the density of concurrent users of about 5x on average.

Regarding the server architecture differences, figure 3 shows a simple side-by-side comparison between a *VM-based* and a *Container-based* solution for a Cloud Gaming server. In this example, `X Server` is used, and it refers to the `X Windowing System` used for graphics workstations. It is based on a Client/Server model, where a networked computer or workstation (in our case the Cloud Gaming server) runs an `X Server`, and *client programs* running on connected workstations request services from the server.

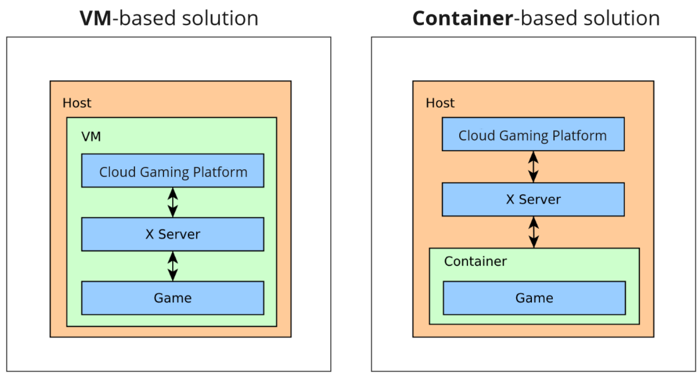

Figure 3: VMs and Containers: comparison for *Cloud Gaming* servers (reworked from [12])

3 Containers for Cloud Gaming

Containers have a smaller footprint when compared to VMs. This means that they require fewer resources to run, making them more efficient in terms of CPU and memory usage. This improved efficiency allows handling more game instances simultaneously on the same physical *Cloud Gaming server*.
This is of extreme importance for Cloud Gaming as it makes it possible for more players to be served with the same amount of resources, resulting in *increased scalability* and *lower costs*.
The greater flexibility provided by Containers also makes them a preferred solution over VMs. Containers can be easily moved between servers and platforms, which allows for more efficient resource allocation and better load balancing [18]. This is especially important for *Cloud Gaming*, where game instances need to be provisioned and de-provisioned on demand, and where traffic can fluctuate rapidly [7]. Containers can be deployed quickly and easily, allowing for fast scaling up and scaling down of resources as needed.
The existence of systems such as `Kubernetes`, results in an improved and increased ease of management, deployment, and scaling of game instances [26].
In all of the following comparisons, `Docker` was used as Container, while the `QEMU KVM` as Virtual Machine.

3.1 CPU performance

Even though GPUs are usually more well suited for video encoding, as shown in the work by Safin et. al [19], the use of some hybrid CPU-GPU models has been researched with promising results [10].

Via `ffmpeg` and `x264`, video was encoded to `H.264/MPEG-4 AVC` video coding format. The results [12] reveal that Virtual Machine introduce an overhead of up to 16%.
Cloud Gaming requires very low delays, most sources recommend latencies below 40ms [22, 23]. Because of such tight requirements, encoding overheads can result in noticeable reduced Quality of Experience (QoE).

3.2 Memory and power usage

Virtual Machines usually are assigned a static and constant amount of the host's RAM, leading to sub-optimal memory allocation and usage. On the other hand, Containers are not affected by this problem.
Moreover, power consumption was found to be lower when using Containers over Virtual Machines [15], which is an important aspect for a Cloud Gaming service.

3.3 GPU performance

Coming now to the GPU comparison, VMs were found to have up to an 8% loss compared to native performance. Containers, on the other hand, achieve no significant loss in performance. Furthermore, as Containers are instances of processes running in the OS, sharing the GPU resources is automatic. Quite different if we compare it with VMs, which require specific GPU passthrough technology such as Nvidia's vGPU [6] to achieve similar results.
When comparing GPU performance based on *frames per second* (fps), the following are the results from this study [12]:

- `Native`:
  - FULL HD video (fps): 297
  - 4K video (fps): 85
- `Container`:
  - FULL HD video (fps): 297 (±0%)
  - 4K video (fps): 85 (±0%)
- `VM`:
  - FULL HD video (fps): 224 (−22.4%)
  - 4K video (fps): 56 (−33.6%)

3.4 Game performance

After analyzing factors influencing scalability and cost on the infrastructure's side, we are now evaluating the different Quality of Services (QoS) and QoE.
Also, in this case, VMs presented significant overheads and worse performance when benchmarked by testing various games. Depending on the GPU used, overhead could reach values of up to 27%.

3.5 Startup time performance

Although overseen, startup time is of vital importance to assure a high *QoE*. Long waiting times would disrupt the concept of *gaming on demand*. Containers can offer almost instantaneous startup times, compared to almost half a minute of waiting [13] for traditional VMs. Moreover, recent research has discovered new ad-hoc dependency scheduling methods to further improve containers' startup time [11].

4 Virtualization and Security

Despite the positives, Containers present some security limitations in comparison to traditional VMs.
Containers are known to offer minimal separation between the hosting OS and other containers on the same machine. This, as a consequence, leads to security boundaries being less robust compared to Virtual Machines [25].
This means that performance and scalability come with a trade-off when compared to security. Despite the former's importance, security must be a central concern.
Considering the case of Docker, the work from Lee et. al [14] demonstrate that the network connection among Docker containers can lead to weakened independence container characteristics. Because of this, `DDoS` (Distributed Denial of Service) and *Man-In-The-Middle* (MITM) attacks are potentially possible. Considering that both accounts and profiles used to access various stores and specific game-related content are accessed from within the *Cloud Gaming* instances, means that in case of security failures or MITM attacks, a data breach is to be considered as a potential risk.
Hybrid solutions have been proposed, where containerization is performed from within Virtual Machines to solve security issues and concerns [17]. Although an interesting solution, it's difficult to see this as a feasible option for *Cloud Gaming*, as performance limitations would still be an issue due to VMs' hardware-level virtualization.
Also, the more limited isolation in Containers means that if a security vulnerability exists in the Kernel, it could affect all the containers running on that host, leading to decreased *QoS* and *QoE*, together with all the security risks associated.

5 Containers in Game Development

An in-depth analysis of the advantages of Containers when dealing with Virtualization requiring GPU-intensive computations is of great importance for *AAA* Game Development. Several articles have shown how, despite there being great advantages, the progress and speed at which this new technology is being integrated and used could be greatly improved [9].

A good example of efficient integration of Containers in a AAA Game Development environment is given to us directly from a Riot blog post [21] by Stewart. Here, he explains how the creators of League Of Legends, transitioned to using containers for *Continuous Deployment*. Using Docker container files meant they could achieve their *core principle*: engineering ownership over their build environments. Developers found it faster to iterate and deploy improvements to their audience, which meant League of Legends players were always getting the best possible experience and the latest code.
As explained in [16, 5], containers together with Kubernetes can make it possible to better scale the game servers.
Improved and more easily manageable back-end architectures for game servers will help create new and better online gaming experiences, capable of handling a greater number of players. An example of this can be found in [4].

6 Conclusions

As thoroughly presented, *Containers* have important advantages over traditional virtualization methods for *Cloud Gaming* solutions.
The lightweight, portable, and scalable nature of *Containers* makes them an ideal choice for *game developers* and Cloud Gaming providers.
The clear benefits of *Containers* extend beyond the realm of gaming, with the potential to revolutionize *game development* itself. Providing a consistent environment for developers, allows them to create and deploy games with ease.

As the popularity of *Cloud Gaming* continues to rise, *Containers* are sure to play an increasingly critical role in the industry, becoming the *go-to* solution for *game developers* and *Cloud Gaming* providers.

References

[1] Cloud Gaming Market: Global industry trends, share, size, growth, opportunity and forecast 2023-2028. https://www.imarcgroup.com/cloud-gaming-market. Accessed: 2023-04-06.

[2] Containers for game development. https://pages.awscloud.com/Containers-for-Game-Development_2020_0322-GAM_OD.html?&trk=el_a131L0000084iEtQAI&trkCampaign=March_2020_0322-GAM&sc_channel=el&sc_campaign=pac_Q1-2020_exlinks_blog_OTT_03DGAB&sc_outcome=Product_Adoption_Campaigns&sc_geo=NAMER&sc_country=mult. Accessed: 2023-04-07.

[3] How many users can share a gpu? https://www.citrix.com/blogs/2013/02/22/how-many-users-can-share-a-gpu/. Accessed: 2023-04-06.

[4] Large-scale multiplayer gaming on kubernetes (cloud next '19). https://www.youtube.com/watch?v=Imf_urBB7SI. Accessed: 2023-04-25.

[5] Optimize game servers hosting with containers. https://aws.amazon.com/it/blogs/gametech/optimize-game-servers-hosting-with-containers/. Accessed: 2023-04-07.

[6] Quantifying the impact of virtual GPUs: See how nvidia benchmarks ux in vmware virtualized environments. https://images.nvidia.com/aem-dam/Solutions/design-visualization/solutions/resources/documents1/quantifying-impact-of-vgpu-whitepaper.pdf. Accessed: 2023-04-07.

[7] Wei Cai, Ryan Shea, Chun Ying Huang, Kuan Ta Chen, Jiangchuan Liu, Victor C.M. Leung, and Cheng Hsin Hsu. A survey on cloud gaming: Future of computer games. *IEEE Access*, 4:7605–7620, 2016.

[8] Hao Chen, Xu Zhang, Yiling Xu, Ju Ren, Jingtao Fan, Zhan Ma, and Wenjun Zhang. T-Gaming: A Cost-Efficient Cloud Gaming System at Scale. *IEEE Transactions on Parallel and Distributed Systems*, 30(12):2849–2865, 2019.

[9] Game Design. Kubernetes for Game Development. 2021.

[10] Ling Yu Duan, Wei Sun, Xinfeng Zhang, Shiqi Wang, Jie Chen, Jianxiong Yin, Simon See, Tiejun Huang, Alex C. Kot, and Wen Gao. Fast MPEG-CDVS encoder with GPU-CPU hybrid computing. *IEEE Transactions on Image Processing*, 27(5):2201–2216, 2018.

[11] Silvery Fu, Radhika Mittal, Lei Zhang, and Sylvia Ratnasamy. Fast and efficient container startup at the edge via dependency scheduling. *HotEdge 2020 - 3rd USENIX Workshop on Hot Topics in Edge Computing*, 2020.

[12] Teemu Kamarainen, Yuanqi Shan, Matti Siekkinen, and Antti Yla-Jaaski. Virtual machines vs. containers in cloud gaming systems. *Annual Workshop on Network and Systems Support for Games*, 2016-January, 2016.

[13] Kuljeet Kaur, Tanya Dhand, Neeraj Kumar, and Sherali Zeadally. Container-as-a-service at the edge: Trade-off between energy efficiency and service availability at fog nano data centers. *IEEE Wireless Communications*, 24(3):48–56, 2017.

[14] Haneul Lee, Soonhong Kwon, and Jong-Hyouk Lee. Experimental analysis of security attacks for docker container communications. *Electronics*, 12(4), 2023.

[15] Martin Lindström. Containers Virtual machines comparison. 2022.

[16] "Mark Mandel". Scaling dedicated game servers with kubernetes. `https://www.gamedeveloper.com/programming/scaling-dedicated-game-servers-with-kubernetes-part-1-containerising-and-deploying`. Accessed: 2023-04-08.

[17] Ilias Mavridis and Helen Karatza. Combining containers and virtual machines to enhance isolation and extend functionality on cloud computing. *Future Generation Computer Systems*, 94:674–696, 2019.

[18] Manoj Kumar Patra, Sanjay Misra, Bibhudatta Sahoo, and Ashok Kumar Turuk. Gwo-based simulated annealing approach for load balancing in cloud for hosting container as a service. *Applied Sciences*, 12(21), 2022.

[19] Ramil Safin, Emilia Garipova, Roman Lavrenov, Hongbing Li, Mikhail Svinin, and Evgeni Magid. Hardware and software video encoding comparison. *2020 59th Annual Conference of the Society of Instrument and Control Engineers of Japan, SICE 2020*, (September):924–929, 2020.

[20] Ryan Shea, Jiangchuan Liu, Edith Ngai, and Yong Cui. Cloud gaming: Architecture and performance. *IEEE Network*, 27(4):16–21, 2013.

[21] "Maxfield F. Stewart". Thinking inside the container. `https://technology.riotgames.com/news/thinking-inside-container`. Accessed: 2023-04-08.

[22] Mirko Suznjevic, Ivan Slivar, and Lea Skorin-Kapov. Analysis and QoE evaluation of cloud gaming service adaptation under different network conditions: The case of NVIDIA GeForce NOW. *2016 8th International Conference on Quality of Multimedia Experience, QoMEX 2016*, pages 2–7, 2016.

[23] Abdul Wahab, Nafi Ahmad, Maria G. Martini, and John Schormans. Subjective Quality Assessment for Cloud Gaming. *J*, 4(3):404–419, 2021.

[24] Miguel G. Xavier, Marcelo V. Neves, Fabio D. Rossi, Tiago C. Ferreto, Timoteo Lange, and Cesar A.F. De Rose. Performance evaluation of container-based virtualization for high performance computing environments. *Proceedings of the 2013 21st Euromicro International Conference on Parallel, Distributed, and Network-Based Processing, PDP 2013*, pages 233–240, 2013.

[25] Anuj Kumar Yadav, M. L. Garg, and Ritika. *Docker containers versus virtual machine-based virtualization*, volume 814. Springer Singapore, 2019.

[26] Qi Zhang, Ling Liu, Calton Pu, Qiwei Dou, Liren Wu, and Wei Zhou. A Comparative Study of Containers and Virtual Machines in Big Data Environment. *IEEE International Conference on Cloud Computing, CLOUD*, 2018-July:178–185, 2018.