Failures in Modern Cloud Systems

Frankie Chukwudolue, Ryan Shah, Prince Uduka, and Luke Johnston

Kennesaw State University

IT3423: Op Sys Concepts & Admin

Professor Jamie Jamison

May 8, 2023

Introduction

Failures in modern cloud systems can have significant impacts on organizations, including downtime, data loss, and financial losses. To prevent failures in modern cloud systems, cloud providers must implement robust controls and safeguards for software, hardware, environmental and scheduling risks. Additionally, cloud providers must have disaster recovery plans in place to mitigate the impact of failures in modern cloud systems.

Environmental Failures

Data is one of the most valuable assets that any company can hold. One of the best ways to store these assets is within a modern cloud system. As a result, modern cloud systems have become the go-to platform for organizations to store and process their data due to their scalability, cost-effectiveness, and flexibility. However, like any other technology, modern cloud systems are not immune to environmental failures.

Environmental failures in modern cloud systems are unexpected events that disrupt the availability, performance, or security of cloud-based services, resulting in downtime, loss of data, and financial loss for organizations. Examples of environmental failures in modern cloud systems include power outages, weather/natural disasters, and cooling system failures. Power outages are one of the most common environmental failures in modern cloud systems. They occur when there is a disruption in the power supply to the data center, causing the cloud infrastructure to shut down. The impact of power outages can be severe, resulting in loss of data, downtime, and financial loss for organizations. To prevent power outages, cloud providers usually have backup generators and uninterruptible power supply (UPS) systems in place. Natural disasters such as fire, hurricanes, earthquakes, and tornadoes can cause significant damage to cloud infrastructure, resulting in downtime and data loss. To mitigate the impact of natural disasters, cloud providers usually have disaster recovery plans in place that involve backing up data in multiple locations and ensuring that critical systems are replicated across multiple data centers. Temperatures are crucial in maintaining the health of cloud infrastructure. High temperatures can cause equipment to overheat and fail, resulting in downtime and data loss. Similarly, low temperatures can cause condensation, which can damage equipment and result in data loss. To maintain optimal temperatures, cloud providers usually have specialized cooling systems in place to maintain the temperature in the data center.

Some real-world examples of environmental failures in modern cloud systems include GCP, OVHCloud, AWS, and Azure. In 2018, a power outage in the eastern United States disrupted the availability of services for Google Cloud Platform (GCP) customers. The outage was caused by a failure at a substation in Virginia, which caused widespread power outages in the region. GCP customers in the affected areas experienced outages for a variety of services, including Google Compute Engine, Google Cloud Storage, and Google Kubernetes Engine. A data center run by OVHCloud was destroyed in early 2021 by a fire. All four data centers had been too close, and it took over six hours for firefighters at the scene to put out the blaze. This severely affected the cloud services run by OVHCloud and spelt disaster for companies whose entire assets were hosted on those servers. In June 2016, storms in Sydney battered the electrical infrastructure and caused an extensive power outage. This led to the failure of a number of Elastic Compute Cloud instances and Elastic Block Store volumes which hosted critical workloads for a number of large companies. This meant that some heavily trafficked websites and the online presence of some of the biggest brands was decimated for over ten hours on a weekend, severely affecting business. In 2019, a hardware failure at a data center in Hong Kong disrupted the availability of services for Microsoft Azure customers in Asia. The failure was caused by a hardware failure in a cooling system, which caused the temperature in the data center to rise to dangerous levels. Azure customers in the affected areas experienced outages for a variety of services, including Microsoft Azure Virtual Machines, Microsoft Azure Storage, and Microsoft Azure SQL Database.

Software Failures

Cloud computing is a paradigm that enables the delivery of various services over the Internet, such as data processing, storage, resource management, and more. Cloud systems are composed of multiple independent and interacting subsystems, each providing specialized functionalities. However, cloud systems are also prone to various types of failures, such as fail-stop, fail-partial, Byzantine, and fail-slow (Zhaosheng et al., 2023). These failures can affect the reliability, availability, performance, and security of cloud services, and cause service outages or degradation. One of the main causes of software failures in cloud systems is the complexity of the software and the underlying infrastructure. Cloud systems typically consist of multiple layers of software and hardware such as operating systems, virtualization platforms, databases, network components, and applications. These layers interact with each other in complex ways and any failure in one layer can propagate to others leading to system-wide failures. For example, a software bug in an application running on a virtual machine may cause the virtual machine to crash which in turn may affect the availability of other applications running on the same machine or other machines in the same cluster.

Another cause of software failures in cloud systems is cross-system interaction failures (CSI failures). These failures are caused by discrepancies between interacting subsystems that cannot be understood by analyzing one single system in isolation (Tang et al., 2023). CSI failures are manifested through interactions across the system boundaries. For example, a failure in a data processing subsystem may propagate to a storage subsystem through a data transfer operation. Similarly, a failure in a resource management subsystem may affect a data processing subsystem through a resource allocation operation. These failures are not caused by bugs or faults within one single subsystem but by discrepancies between interacting subsystems. CSI failures are hard to detect and diagnose because they involve multiple subsystems with different designs, implementations, and semantics. Moreover, CSI failures are affected by the non-deterministic behavior of cloud systems, which introduces noise and variations in the timing and ordering of events (Zhaosheng et al., 2023). Therefore, CSI failures require a comprehensive understanding of the cross-system interactions and their failure patterns.

To mitigate or prevent software failures in cloud systems cloud providers can take several steps. First, they can implement redundancy/failover mechanisms to ensure that critical applications and data are available even in the event of software failures. Second, they can improve their visibility and control over the underlying infrastructure by using monitoring/management tools that provide real-time performance metrics, alerts, and dashboards. Lastly, they can prioritize testing and quality assurance by conducting regular penetration testing, vulnerability scanning, and code review.

Hardware Failures

Hardware failures consist of the internal components of a computer failing to complete a function. Whether this is not powering another component correctly, powering on itself, or not being able to communicate with another device. When it comes to cloud computing hardware failures typically occur when under stress from the environment, defective components, and lack of maintenance (Bakhshi et al., 2014).

One of the main causes of the environment causing hardware failures is temperature (Bakhshi et al., 2014). If cooled improperly, components can reach temperatures of 100 degrees Celsius, causing components to cease function temporarily or permanently. Often these components are kept in cool rooms to keep components from failing. Though if the cooling systems are not properly maintained or checked, they could fail over time causing systems to overheat and fail.

Power surges are also another environmental cause of hardware failure. When a surge of power goes through computer components, it can cause the circuitry to fail temporarily or permanently and on multiple components. Using power surge protectors can help reduce the likelihood of a power surge but cannot prevent it. Power surges often occur during storms but can occur at any time due to a failure in infrastructural power.

Defective parts can occur on any component in the computer. Though recently with cloud computing there seems to be a bigger issue with processors. As processing chips have gotten smaller the ability to determine defects has become more difficult (Bakhshi et al., 2014). Often these failures are difficult to pinpoint what caused them. Though due to the sheer number of transistors on a single chip there are bound to be defective chips. Though at this point, as we advance, all we can do is research ways that allow us to determine if one of these processing chips is defective or not.

Lack of maintenance on these cloud systems can lead to hardware failures. If the systems or environments are not properly maintained these cloud devices could overheat and fail. Also, if there are not redundancies in place, if a storm passes through and causes a power surge, a lot of devices could be damaged permanently. Lastly, if the systems are not checked for failures in the hardware or software frequently, then these systems are prone to fail without notice due to defective parts.

Scheduling Failures

Scheduling problems are problems when attempting to match elements from different sets and are expressed as “triple (E,S,O) where E is the set of examples, S is the set of feasible solutions and O is the object of the problem" (*Scheduling problems for Cloud computing,5).* Task scheduling plays a vital role in the efficiency of cloud systems as they can reduce turnaround time and vastly improve resource utilization. Because of the structure of the cloud environment, it is difficult to prevent task failure and maintain a high-level quality of service. Large scale data centers with huge sets of services for global consumers cannot afford to fail and multi-objective scheduling of data center services led to consistent failures. If a scheduling algorithm is not optimized, it leads to inefficient use of energy and resources as the cloud system cannot fully keep up with the level of workload scheduled.

There are two main types of scheduling problems depending on the object. It is either an optimization problem or a decision problem. Optimization problems are more difficult to solve as they require all feasible solutions to be analyzed to find the optimal solution in the set while a decision problem only requires a positive or negative answer to if object O is achieved. Depending on if it is an optimization problem or decision problem different algorithms are created that follow a specific method to solve the problem. In the case of a decision problem or NP-complete the enumeration method is employed where each possible solution is compared one by one until a solution is found. If it is an optimization problem, then the heuristic method or approximation method is used where we either relax constraints imposed on the original solution or iteratively improve a suboptimal approximate solution.

There have been multiple methods used to try to mitigate potential failures in scheduling. One solution is the improved use of distributed scheduler for user system performance and a centralized scheduler enhanced system performance. A proposed innovation to current cloud systems is fault-tolerant failure aware task scheduling system which guarantees execution of a task in real time.

Works Cited

Bakhshi, R., Kunche, S., & Pecht, M. (2014). Intermittent failures in hardware and software. *Journal of Electronic Packaging*, *136*(1). https://doi.org/10.1115/1.4026639

Bing Huang, Xiaojun Li, Ming Li, Bernstein, J., & Smidts, C. (n.d.). Study of the impact of hardware fault on software reliability. *16th IEEE International Symposium on Software Reliability Engineering (ISSRE'05)*. https://doi.org/10.1109/issre.2005.39

Birke, R., Giurgiu, I., Chen, L. Y., Wiesmann, D., & Engbersen, T. (2014). Failure analysis of virtual and physical machines: Patterns, causes and characteristics. *2014 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*. https://doi.org/10.1109/dsn.2014.18

Cotroneo, D., Simone, L. D., Liguori, P., & Natella, R. (2021, July 12). Enhancing the analysis of software failures in Cloud Computing Systems with deep learning. Journal of Systems and Software. https://www.sciencedirect.com/science/article/abs/pii/S0164121221001400

Proactive failure-aware task scheduling framework for cloud computing. (n.d.). Retrieved May 8, 2023, from https://ieeexplore.ieee.org/abstract/document/9500123

Rock, T. (2022, October 11). 7 Strategies to Avoid Data Loss from Natural Disaster. InvenioIT. Retrieved April 26 from https://invenioit.com/continuity/data-loss-from-natural-disaster/

Scheduling problems for Cloud computing - dergipark. (n.d.). Retrieved May 8, 2023, from https://dergipark.org.tr/tr/download/article-file/714041

Szymański, K. (2021, June 21). Disaster Recovery in Cloud Computing. Netguru. Retrieved April 26, 2023, from https://www.netguru.com/blog/disaster-recovery-in-cloud-computing

Tang, L., Zhang, Y., Bhandari, C., Ji, S., Karanika, A., Xu, T., & Gupta, I. (n.d.). Fail through the cracks: Cross-system interaction failures in modern cloud systems. https://dprg.cs.uiuc.edu/data/files/2023/eurosys23-fall-final-CSI.pdf

Zhaosheng, Lu, R., Xu, E., Zhang, Y., Zhu, F., Wang, M., Zhu, Z., & Xue, G. (2023, February 9). Detecting fail-slow failures in large-scale cloud storage systems. USENIX. https://www.usenix.org/publications/loginonline/detecting-fail-slow-failures-large-scale-cloud-storage-systems