SYNCHROINFO JOURNAL

www.media-publisher.eu/synchroinfo-journal

# PROACTIVE TESTING AS A METHOD OF ENSURING THE GAME SERVERS EFFICIENCY

**Andrey Ladonov [1], V.A. Dokuchaev [1]**

[1] Network Information Technologies and Services, MTUCI, Moscow, Russia
vufhg4@mail.ru; v.a.dokuchaev@mtuci.ru

**ABSTRACT**

In a highly competitive environment where players have instant access to a multitude of alternatives, server failures, manifested as increased network latency, connection drops, or complete service unavailability, directly lead to negative reviews, player attrition, and significant financial losses. Therefore, developing a comprehensive approach to ensuring the stability, fault tolerance, and efficiency of game servers is a critical task requiring the systematic application of proactive testing methods and the deployment of continuous monitoring. This article is devoted to a detailed analysis of these methods and their integration into a unified reliability system. The research focuses on methods for ensuring the stability, fault tolerance, and efficiency of game servers. The goal of the study is to identify stability, fault tolerance, and efficiency metrics and methods for achieving optimal performance based on proactive testing.

DOI: 10.36724/2664-066X-2026-12-1-50-56

*Received*: 07.12.2025
*Accepted*: 10.02.2026

**Citation:** Andrey Ladonov, V.A. Dokuchaev, "Proactive testing as a method of ensuring the game servers efficiency," *Synchroinfo Journal* **2026**, vol. 12, no. 1, pp. 50-56.

Licensee IRIS, Vienna, Austria.



**KEYWORDS:** *testing, video games, computer games, performance, stability, monitoring, event log, fault tolerance*.

## Introduction

In the video game industry, particularly in the multiplayer online gaming segment, ensuring the stable and uninterrupted operation of server infrastructure [1-4] is transforming from a technical challenge into a matter of commercial success and maintaining user loyalty. In a highly competitive environment where players have instant access to a multitude of alternatives, server failures, manifested as increased network latency, connection drops, or complete service unavailability, directly lead to negative reviews, player attrition, and significant financial losses. Therefore, developing a comprehensive approach to ensuring the stability, fault tolerance, and efficiency of game servers is a critical task requiring the systematic application of proactive testing methods and the deployment of continuous monitoring. This article is devoted to a detailed analysis of these methods and their integration into a unified reliability system.

## Ianalysis of game server performance testing tools

Proactive testing, which is conducted during the development and build stages and before the deployment of major updates, is the foundation for creating a resilient system . Its primary goal is to identify and fix performance and functionality defects under controlled conditions, before these issues can impact the end-user experience [5]. Central to this process is load testing, which aims to verify the server system's behavior under the predicted workload. This type of testing is regulated by the principles set forth in GOST R 56938-2016 and involves simulating the activity of thousands of virtual users performing typical game actions: authentication, movement through the game world, use of abilities, and interaction with objects [6]. The testing strategy includes both a gradual increase in load to determine the point of performance degradation and a pulsed increase, simulating a sharp surge in activity, similar to a project launch or the release of a major content update. Key metrics at this stage include network latency, throughput, computing power consumption, and error rate [7]. To adequately assess the user experience, it is necessary to analyze not only average latency values, but also their 95th and 99th percentiles, which allows for the consideration of worst-case scenarios for individual players.

The next logical step is stress testing, which aims to investigate the system's behavior beyond its design performance limits [8]. According to the same GOST R 56938-2016, the goal is not only to detect the limit beyond which the system can no longer cope with the load, but also to analyze its behavior under stress and subsequent recovery mechanisms [9]. This checks whether the system is able to properly terminate active sessions, whether its failure does not provoke a cascading disruption of dependent services, such as databases or authentication services, and whether RAM leaks occur. This picture is complemented by stability testing, also known as endurance testing [10]. During this long-term test, the server is subjected to a stable high load for 12-24 hours or more, which allows for the detection of gradually accumulating problems: incremental memory leaks, fragmentation of data in caches, uncontrolled growth of log files, and the accumulation of errors in databases that do not manifest themselves during short-term tests [11].

## Conducting functional testing for game servers

In parallel with performance testing, functional testing under load conditions must be conducted to ensure that all game logic remains correct under massive simultaneous player activity [12]. This includes checking the correctness of calculations in massive PvP battles, the synchronous completion of group tasks, and the integrity of transactions during intensive exchange of virtual items. An equally important component of proactive preparation is security testing, aimed at identifying vulnerabilities that could be exploited by attackers to destabilize the service or gain an advantage [13]. This includes testing resilience to distributed denial-of-service attacks, auditing code for vulnerabilities such as SQL injection, and testing game logic for the possibility of unauthorized operations, such as duplication of in-game assets.

However, even the most thorough proactive preparation cannot guarantee uninterrupted operation in real-world conditions, which are always more complex and diverse than any laboratory model [14]. Therefore, the second fundamental pillar of reliability assurance is the deployment of a continuous monitoring system that performs the function of constantly diagnosing the "health" of the industrial environment. This process can be divided into several interconnected levels [15]. The basic level—infrastructure monitoring—focuses on tracking the state of physical and virtual resources: CPU utilization, RAM

consumption, disk subsystem metrics, and network interface [16]. Tools such as Prometheus in conjunction with Grafana , Zabbix , or cloud monitoring platforms can be used for this purpose.

More significant from a user experience perspective is monitoring the game application itself [17]. This provides data on what players actually experience. Key metrics here include the game state update rate, the processing time of one game "tick," and detailed network metrics such as latency, jitter, and packet loss rate [18]. Monitoring these metrics at a percentile level allows for the identification of issues that may not be apparent when analyzing average values. Simultaneously, business metrics such as the peak and average number of concurrent users are collected and analyzed, serving as the basis for resource planning and automatic scaling [19].

Special attention is given to the analysis of event logs. Centralized collection of logs from all servers into a single system, such as the ELK stack, enables rapid incident investigation [20]. Automated real-time error and exception tracking ensures rapid response to emerging issues [21]. In a microservice architecture, distributed tracing becomes an important tool , allowing the path of a single request to be traced through multiple services, pinpointing the source of a delay or failure [22]. Synthetic monitoring, in which automated scripts launched from various points continuously simulate the actions of a real player according to a predetermined scenario [23], complements the operational picture. This allows for the detection of service degradation or unavailability before users begin to complain en masse [24].

According to the online publication CNews, Kubernetes adoption in Russia is growing at a level approaching the industry standard, with customers expecting guaranteed availability, easy-to-read performance metrics, and a simplified update process [25]. Essentially, customers are seeking the same benefits previously considered necessary for maintaining game servers.

Moreover, according to the findings of CNews Analytics, in 2024 alone, the interest of large customers in Kubernetes in Russia has significantly increased , such as Burger King, AvtoVAZ , Fix Price , Magnit, Rostiks , Gostekh , State Unified Cloud Platform, Post Bank, Rosatom and a number of other significant companies [26]. According to the same data, the main suppliers of Kubernetes platforms are VK Tech », « Yandex Cloud ", "Basis", " Orion soft " and others. As is known, " VK " has a gaming division . Play , which is engaged in publishing and development of games, and also provides a service for the digital distribution of gaming products, it is quite possible to expect the use of Kubernetes in the gaming sector.

Overall, customers require unified management of clusters deployed across various cloud and on-premises environments. The " multicluster " model, where individual clusters are dedicated to specific tasks, is becoming more widespread, increasing the isolation and stability of workloads [27].

At the same time, information security issues are becoming particularly relevant; a key area of development here is process control technologies that allow for the identification of anomalies in the behavior of containers and processes directly at the node level, thereby closing vulnerabilities that are inaccessible to traditional security systems [28].

While cloud solutions remain attractive, on -premises deployments remain preferred by many customers. This is driven by both economic factors, such as reduced cost of ownership with a stable workload, and the desire to avoid the risks associated with potential failures with a single cloud provider. As a result, hybrid and multi-cloud architectures, which provide flexibility and fault tolerance, are becoming an increasingly popular compromise choice [29].

**Dedicated gaming server, VPS or cloud – the problem of choice**

Let's look at the steps for selecting a dedicated server for online games or game hosting, taking into account the nuances that are important for game project owners, developers, and DevOps engineers.

Online games (especially multiplayer games with synchronous interaction) are critical to latency and stability:

- a  y lags and freezes mean lost users;
- a s  rver crash during a tournament or stream is a blow to your reputation;

• high TTFB (time to first byte) in open-world games – degrades the gaming experience.

VPS and cloud VMs don't always provide the desired level of predictability:
• vir ualization often creates "noisy neighbors";
• I/O disk and CPU may be unstable under peak load;
• high RTT (ping) due to shared communication channels.

A dedicated server wins in this regard:
• full control ver the hardware;
• fixed and stable resources (CPU, RAM, I/O);
• the ability to fine-tune the OS kernel and network stacks to suit the needs of a specific game;
• Lower network latency with the right data center selection.
Let's consider which parameters for choosing a dedicated server for gaming projects are the most critical.

Processor (CPU) performance
For PC gaming, single-core performance is more important than multi-threading.
A common mistake is to use a 32-core server with a low frequency for the sake of marketing figures. In reality, 2-4 cores at 4.0-5.0 GHz will provide better gaming performance than 16 cores at 2.2 GHz.
This is especially true for:
• Minecraft (single-thread);
• CS:GO
• s rvers on the Source engine;
• pr vate MMO servers.
The higher the clock frequency, the higher the tickrate, the smoother the gaming process.
Recommendation: Servers based on Intel Core i9, Xeon E-2388G or AMD Ryzen /EPYC with a frequency of 4.0 GHz or higher.

Random Access Memory (RAM)
Memory for game servers is critical depending on the engine and the number of players:
• s all servers (up to 50 slots) – 8–16 GB;
• medi m (up to 100–200 slots) – 32 GB;
• lar e projects (MMO, server streaming) – 64–128 GB and above.
Important: different games have different RAM consumption patterns. For example, Minecraft with mods or large modified Rust servers can require 64–128 GB even for a single instance.
It is also necessary to take into account memory for caching and logging.

Storage type: SSD or NVMe
Game engines actively read and write files to disk:
• w rld maps;
• custom mods;
• logs;
• preservation;
• statistics.
HDDs are absolutely unsuitable for gaming projects in 2025. Even basic SSDs are not the best option. Servers with NVMe drives are recommended, especially for Minecraft, Valheim, ARK, GTA RP, and other disk-intensive games.

Network channel and ping
The most critical parameters include:
• RTT bove 40–50 ms.;
• PvP servers, even + 10ms gives a competitive advantage to some and causes a churn in others;
• dir ct connection to Tier 1/2 providers (so there are no 10 hops to the client);
• ded cated channel 1 Gbps or higher;
• the minimum route to the main regions of Central Asia (EU, CIS, Asia – depending on the game);

• DDoS prot ction (mandatory – gaming projects are regularly attacked).

It is recommended to check the data center's ping in advance for gaming traffic, for example, using the provider's Looking Glass or a real traceroute.

Reliability and stability

Downtime is unacceptable in the gaming industry. A server crash in the middle of a prime evening or tournament means a loss of loyalty. Therefore, it's important:

• Tier III or h gher data centers;
• SLA not lower than 99. % (ideally 99.99%);
• 24/7 sup ort (not during business hours);
• Avail bility of automatic backups or the ability to organize them.

Nuances for game hosting (multi-tenancy)

For gaming hosting (for example, a service selling Minecraft or CS: GO servers), the following characteristics are important:

• high ensity (many small instances per server);
• fast dep oyment automation (via panels such as Pterodactyl, TCAdmin);
• large NVMe storage capacity for snapshots;
• D cker or KVM containers);
• CPU RAM limit control at the OS kernel level.

In this case, it is better to use custom solutions or bare-metal servers with high single-thread-per-core performance.

Regarding the use of VPS, it should be noted that it is advisable to use it when solving the following problems:

• d velopment and testing of servers (not in production);
• s all community servers (up to 10 slots);
• game chats, community sites;
• tickrate requirements.

But for serious gaming projects, only a dedicated server or bare-metal cloud (such as Selectel Metal Cloud or OVH Game) will provide the required level of control and performance.

Table 1 shows the main characteristics of some gaming services [30].

Table 1

**Main characteristics of gaming services**

| | **GFN.RU** | **LoudPlay** | **VK Play Cloud** | **Firewood** | **Playkey** |
|---|---|---|---|---|---|
| Availability | PC, Android and IOS smartphones, smart TV | PC, Android smartphones | PC, Android smartphones, smart TVs | PC | PC |
| Image resolution | HD or Full HD | Full HD | Up to 4K | HD or Full HD | HD or Full HD |
| FPS | up to 60 | up to 60 | up to 120 | up to 144 | up to 120 |
| Cross-platform play support | No | No | No | Yes | No |
| Own library of games | Yes | Yes | Yes | No | Yes |
| What can be launched? | Games only | Any games and programs | Any games and programs | Games only | Any games and programs |

Table 2 presents some results of testing of gaming services [31], which may be useful for game project owners, developers and DevOps engineers.

Table 2

**Results of testing gaming services**

| | LoudPlay | VK Play Cloud | Firewood | Playkey |
|---|---|---|---|---|
| Convenience | 4 | 5 | 3 | 5 |
| FPS | 4 | 5 | 4 | 5 |
| Image quality | 3 | 4 | 3 | 2 |
| Smooth operation | 4 | 5 | 5 | 2 |
| Price | 4 | 4 | 5 | 4 |
| Total | 3.8 | 4.6 | 4.0 | 3.6 |

Thus, it can be concluded that cloud gaming is a good alternative to purchasing a powerful gaming PC if you have a stable high-speed internet connection and the servers of your preferred service are located close to you.

### Conclusion

Thus, it can be concluded that maximum efficiency in ensuring stability and fault tolerance is achieved not by the isolated application of the described methods, but by their integration into the software lifecycle. Data obtained during load testing is used to adjust thresholds in monitoring systems, enabling the creation of accurate and timely alerts. In turn, information on real-world load patterns and user behavior, collected from production servers, constantly adjusts proactive testing scenarios, making them more relevant and realistic. Thus, investing in robust architecture, automated testing, and comprehensive monitoring is a justified investment in the reputation and long-term success of a gaming project, where a stable and responsive server becomes a key competitive advantage.

## REFERENCES


[1] I. D. Udalov, V. A. Dokuchaev, “Efficient hierarchical pathfinding in dynamic and static game worlds,” *Information Society Technologies: Proceedings of the XIX International Industry Scientific and Technical Conference*, Moscow, March 11-13, 2025. Moscow: MTUCI, 2025, pp. 190-193.

[2] V. A. Dokuchaev, I. D. Udalov, “Comparative analysis of heuristic functions for the pathfinding algorithm A,” *Theory and practice of economics and entrepreneurship: Proceedings of the XXII International scientific and practical conference*, Simferopol – Gurzuf, April 24-26, 2025. Simferopol: IP Zueva T.V., 2025, pp. 275-276.

[3] V. A. Dokuchaev, V. V. Maklachkova, I. D. Udalov, “Application of Entity Component System in Game Development,” *Economics and quality of communication systems*. 2025. No. 1 (35), pp. 57-66.

[4] I. Udalov, V. A. Dokuchaev, “Pathfinding algorithms for computer games,” *Synchroinfo Journal*. 2024. Vol. 10, no. 6, pp. 8-16. DOI 10.36724/2664-066X-2024-10-6-8-16.

[5] Changsong Li, Ning Zhao Wu, “Multiple deception resources deployment strategy based on reinforcement learning for network threat mitigation,” URL https://www.nature.com/articles/s41598-025-00348-0 (date appeals – January 2025).

[6] Jie Cao, Haoxiang Wang, Jingru Jiao, Kekun Hu, Ping Qi, “A social network graph partitioning algorithm based on double deep Q-Network,” URL https://www.nature.com/articles/s41598-025-16768-x (date appeals – January 2025).

[7] Alberto Mozo, Ángel González-Prieto, Antonio Pastor, Sandra Gómez-Canaval, Edgar Talavera, “Synthetic flow-based cryptomining attack generation through Generative Adversarial Networks,” URL https://www.nature.com/articles/s41598-022-06057-2 (date appeals – January 2025).

[8] Wei Jiang, Bin Zhang, Qixun Zhu, Conghui Liao, Wenyong Wang, “A Real Network Environment Dataset for Traffic Analysis,” URL https://www.nature.com/articles/s41597-025-04876-2 (date appeals – January 2025).

[9] Syed Aqib Abbas Naqvi, Faraha Ashraf, Ali Ovais, Muhammad Waheed Rasheed, Amir Asif, Abdu Alameri, “Optimizing hybrid network topologies in communication networks through irregularity strength,” URL https://www.nature.com/articles/s41598-025-05631-8 (date appeals – January 2025).
[10] Chester Wai-Jen Liu, Sheng-Feng Shen, Wei-Chung Liu, “On the evolution of social ties as an instrumental tool for resource competition in resource patch networks,” URL https://www.nature.com/articles/s41599-021-00753-6 (date appeals – January 2025).
[11] Manlio De Domenico. More is different in real-world multilayer networks, URL https://www.nature.com/articles/s41567-023-02132-1 (date appeals – January 2025).
[12] Cuihua Zuo, Peihua Xu, Yachen Song, Jianfeng Lu, Cao Yuan, Yaqin Li, “RAIM: three-stage stackelberg game for hierarchical federated learning with reputation-aware incentive mechanism,” URL https://www.nature.com/articles/s41598-025-16830-8 (date appeals – January 2025).
[13] Yi-Fan Sun, Hai-Jun Zhou, “Serving by local consensus in the public service location game,” URL https://www.nature.com/articles/srep32502 (date appeals - January 2025).
[14] I.A. Pushkin, “Development of a fault-tolerant distributed multiplayer game,” URL https://cyberleninka.ru/article/n/razrabotka-otkazoustoychivoy-raspredelennoy-mnogopolzovatelskoy-igry (accessed January 2025).
[15] K. Kaner, D. Folk, E.K. Nguyen, “Software Testing. Fundamental Concepts of Business Application Management,” Moscow: LORI, 2017. 544 p.
[16] R. Culbertson K. Brown, G. Cobb, “Rapid testing,” Moscow: Williams, 2018. 336 p.
[17] D.N. Kolisnichenko, “Linux Server Administration. Performance Monitoring by Percentiles,” *Open Systems. DBMS*. 2020, no. 3, pp. 34-39.
[18] N.I. Kozyreva, M.O. Mukhtulov, S.A. Ershov, S.V. Novoseltseva, D.A. Akhmadullin, “Modern methods of preventing DDoS attacks and protecting web servers,” URL https://cyberleninka.ru/article/n/sovremennye-metody-predotvrascheniya-ddos-atak-i-zaschity-veb-serverov (date of access – January 2025).
[19] A. Petrov, “Distributed data: algorithms for operating modern systems,” Moscow: Publishing solutions, 2021. 450 p. URL https://www.litres.ru/book/aleks-petrov-3142221/raspredelennye-dannye-algoritmy-raboty-sovremennyh-si-66410120/ (accessed January 2025).
[20] V.V. Chekhanovsky, “Distributed information systems,” Moscow: Solon-Press, 2019. 288 p. URL https://www.litres.ru/book/vv-cehanovskiy/raspredelennye-informacionnye-sistemy-66011233/ (date of access – January 2025).
[21] G.B. Gulian, “Distributed Networks: Modern Technologies and Design Fundamentals,” SPb.: BHV-Petersburg, 2020. 384 p. URL https://www.litres.ru/book/gb-guliyan/raspredelennye-seti-sovremennye-tehnologii-i-osnovy-proektirov-5313113/ (date of access – January 2025).
[22] Source Multiplayer Networking, URL https://developer.valvesoftware.com/wiki/Source_Multiplayer_Networking (date appeals – January 2025).
[23] J. Smed, H. Hakonin, “Algorithms And network interaction for computer games = Algorithms and Networking for Computer Games,” Moscow: DMK Press, 2019. 362 p.
[24] N. Zhang, S. Wang, H. Wu, L. Xu, Q. Yin, C. Wang, W. Zhou, “Strategy deployments multiple deceptive resources on basis training with reinforcements for reductions network threats = Multiple deception resources deployment strategy based on reinforcement learning for network threat mitigation,” *Scientific Reports*. 2025. Vol. 15. P. 16830. DOI 10.1038/s41598-025-00348-0.
[25] Market Kubernetes platforms 2025. CNews.ru. 2025. URL: https://www.cnews.ru/reviews/rynok_platform_kubernetes (date accessed: 23.01.2026).
[26] CNews Analytics: The Largest suppliers Kubernetes platforms 2024. CNews.ru. 2025. URL: https://www.cnews.ru/reviews/rynok_platform_kubernetes/table_detail/b00da2f220b169b99bb7965034ad9b88ec3e22dc (date accessed: 23.01.2026).
[27] Kubernetes platforms, Russian customers choose hybrid and multi-cluster strategies. CNews.ru. 2025. URL: https://www.cnews.ru/reviews/rynok_platform_kubernetes/articles/top-5_rossijskih_suppliers_platform (date of access: 23.01.2026).
[28] Runtime scanning in Kubernetes: will containers become safer? CNews.ru. 2025. URL: https://www.cnews.ru/reviews/rynok_platform_kubernetes/cases/rantajm-skanirovanie_v_kubernetes_stanut (date of access: 23.01.2026).
[29] Alexander Chubov, K2Tech: Implementing containerization “for the sake of fashion” is definitely not worth it. CNews.ru. 2025. URL: https://www.cnews.ru/reviews/rynok_platform_kubernetes/interviews/aleksandr_chubov?erid=2W5zFH7JyEg (date of access: 23.01.2026).
[30] How to choose a dedicated server for gaming projects and game hosting. Myslo.News. 2025. URL: https://myslo.ru/News/company/kak-vybrat-vydelennyj-server-dlya-igrovyh-proektov-i-hostinga-igr (date of access : 02/28/2026).
[31] Polyarus M. Cloud Services Test: What to Play on After GFN.RU Departs? Citylink Magazine. 2023. URL: https://journal.citilink.ru/articles/test-oblachnyh-servisov-na-chem-igrat-posle-uhoda-gfnru/ (accessed: 28.02.2026).