International Journal of Engineering & Technology, 7 (4.29) (2018) 113-118

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

A Model for "Mission Critical Multiple Clients and Single Server Operation for Failure Detection and Recovery System"

Syahaneim Marzukhi[1]*, Nursyafeera Azeera Rahim[2], Syed Nasir Alsagoff Syed Zakaria[3]

[1] *Faculty of Science and Defence Technology, National Defence University Malaysia, 57000 Kuala Lumpur, Malaysia*
[2] *Faculty of Science and Defence Technology, National Defence University Malaysia, 57000 Kuala Lumpur, Malaysia*
[3] *Faculty of Science and Defence Technology, National Defence University Malaysia, 57000 Kuala Lumpur, Malaysia*
**Corresponding author E-mail:syahaneim@upnm.edu.my*

Abstract

This paper proposes a model for "Mission Critical Multiple Clients and Single Server Operation for Failure Detection and Recovery System". The proposed model is based on SAPE model with few amendments that consists of five phases: 1) Establish Connection 2) Sense 3) Analyze 4) Plan and 5) Execute. The proposed model is implemented using "Mission Critical Multiple Clients and Single Server System for Failure Detection and Recovery" to simulate various types of failure. In order to test the system, four sets of experiments are conducted: 1) Interval timer between server and clients, 2) Disrupt Client, 3) Disrupt Server, and 4) Disrupt Network. Findings show the system is successfully executed; the system is able to detect and recover from various types of failure as long as there is incoming data between the clients and the server. In future, it is suggested that the system should include the following module: 1) flexible adjustment of default interval timer between clients and server, 2) introduce multiple clients and multiple server, and 3) add more simulation of failures. This enhancement is needed to ensure the system can sustain its capability in detecting and recovering any mission critical operations.

Keywords: *Failure Detection and Recovery System; Mission Critical System; Multiple Clients and Single Server Operation; SAPE Model;*

1. Introduction

Mission critical system is used for monitoring the operations in the organization in order to avoid failure in a system or a subsystem [6]. Here, multiple clients and single server communication is referred to the architecture of multiple clients which is connected to the single server. In the multiple clients and single server system, usually the client will send a request to access the data at the server, whereas the server will wait for any request from the client and respond back via the network [16].

In mission critical system, there are few requirements that is necessary to ensure the system is more reliable which can minimize faults, avoid faults and eliminate faults. Therefore, these requirements are vital to guarantee that the system can support any mission critical in real time. Further, the communication delay between "Mission Critical Multiple Clients and Single Server System" might occurs due to several factors that cause the operation to fail or success. The communication delay can occur by the following factors: the clients and server system crash the client randomly disconnected from server [4], and network disruption which can affect the operation of the mission [15].

In addition, failure detection and recovery is important in every mission critical system to detect and recover fault in the system by maintaining the original purpose of the system even tough facing an unexpected possible degraded state [14]. Several studies have been conducted related to the issues and factors of failures in the mission critical system. Failures in the "Mission Critical System of Multiple Clients and Single Server" can occur due to: increasing number of clients, network connection failure, hardware failure and environmental factors, single point of failure (SPOF), malfunction server and many more related to the system failures.

This paper focusses on "Mission Critical System of Multiple Clients and Single Server" which suffers many problems during the operations. This paper presents the design and implementation for the "Mission Critical Multiple Client/ Single Server Operation Failure Detection and Recovery System". Therefore, a new model for "Mission Critical Multiple Client and Single Server Operation for Failure Detection and Recovery System" is proposed. A new model is designed based on the adaptation from SAPE model with few amendments. The proposed model consists of five phases: 1) Establish Connection 2) Sense 3) Analyze 4) Plan and 5) Execute. Next, the model is verified using the "Mission Critical System of Multiple Clients and Single Server" to simulate various types of failure, such that the system can detect and recover the simulated failures accordingly.

2. Background

2.1. Mission critical system

Mission critical system can be defined as a failure in a system or a subsystem that will cause inability to maintain the whole operational capability for mission continuity [6]. According to this research [6], every component of the system has its own software, thus fault might occur to this software. In addition, resources such as switches, routers, gateways, service platforms, security devices, applications, storage facilities, and transmission facilities must be taken care to ensure that any given mission can be achieved without any faulty. However, all these resources might fail due to improper design, environmental factor, physical defect and operator error. Besides, server disruption, power outage and problems in software also can occur due to improper integration and testing [7].

According to the previous research [11], mission critical system also can be defined as operational to save human lives during critical situation such as tactical operation, unexpected natural disasters and bombings, which requires a high speed data transmission to ensure the real time situational information is updated. The successful distribution of mission critical operations also depends the ability of network; the bandwidth and speed limits of data transfer [5]. Thus, to save human lives, the emergency responder must be straightaway send off to the disaster zones by means of ubiquitous and strong communication to ensure the search and rescue operations can successfully be implemented [6]. In fact, in future, the communication system should consist of: location and status information, video, audio/voice, push-to-talk, real time text messaging (RTT), broadcasting and multicasting, to support the operation [6]. Therefore, the effective coordination of the search and rescue operations can be organized systematically through the communication system.

2.2. Multiple clients and single server

Multiple clients and single server can be defined as a number of clients connected with one server. Generally, the client will send a request to access the data at the server, whereas the server will wait for any request from the client and respond back via the network [16]. There are two types of communication networks: 1) data networks and 2) control networks. Data networks normally involves transmission of large data packets, while control networks involve transmission of small data packets that must meet time assurance with frequent transmission [1]. Figure 1 shows the architecture of multiple clients and single server.

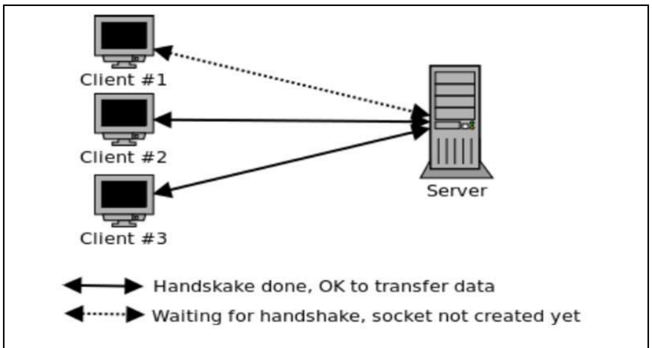

Fig. 1: The communication between multiple clients and single server architecture [8].

Based on Figure 1, the multiple clients and a single server communication is established when a proper connection is linked. Here, the web socket text and data are sent in full duplex mode, where the data transmission can be sent from both directions simultaneously [12]. The communication between the clients and server starts when the client sends a handshake request via web socket protocol and the server responses back by sending a response to the handshake. The web socket transmissions can be represented as "message" where a single message can be a part of few information plans [8]. A concurrent server by means it has an ability to handle multiple clients in parallel. In multiple client environment, each client has to compete with each other for the network bandwidth due to many clients are connected to the server that caused the performance of network traffic to degrade since the clients have to send their messages only once [2]. Yet, this architecture has its own advantages as the client has to maintain a single connection to the server, no matter how many clients are connecting to the server [13].

3. Issues in Multiple Clients and Single Server System

There are number of problems in the multiple client and single server system. For example, increasing number of clients can cause the system's performance drop as the client need to wait in the queue before the task can be sent to the server [1]. In other case, there will be a delay in processing tasks at the server. Another issues in the multiple client and single server system is the network connection failure. This problem is due to the network partition failures, which can be related to omission failures [4]. The omission failure is a situation where the server fails to respond to incoming request [17]. This failure may occur during the network fragmentation; the network is split into two or more sub-network that effect the clients and server communication.

Research related to hostile environments has been conducted using mission critical of multiple clients and single server system [9]. According to this research, there are several factors related to communication issues caused by hardware failure and environment factors such as high radiation, electric disruption and high temperature of the hardware [9]. Besides, single point of failure (SPOF) also affect the communication between the multiple clients and the single server system. SPOF can affect the whole system when one of the system component malfunction during the operation. For example, if the server suddenly fails during operation, then all the clients might be unreachable. This problem caused all the data that is stored in the server to vanish [10].

4. "Mission Critical Multiple Clients and Single Server Operation for Failure Detection and Recovery System" Model

According to [14], fault tolerance system is similar to failure detection and recovery system; a system that has the ability to detect and recover fault in the system by maintaining the original purpose of the system even facing an unexpected possible degraded state. The main objective of the failure detection and recovery system is to minimize mean time to fail (MTTF) that can improve the reliability in computer systems [3]. Here, the "Mission Critical Multiple Clients and Single Server Operation for Failure Detection and Recovery System" Model is based on the "Sense-Analyze-Plan-Execute" (SAPE) Model [3]. This model is previously reformed for autonomic computing. The model consists of four main phases: 1) Sense 2) Analyze 3) Plan and 4) Execute.

However, the "Sense-Analyze-Plan-Execute" (SAPE) model has been modified for the "Mission Critical Multiple Client and Single Server Operation for Failure Detection and Recovery System" in order to suit with the scenario under studied. Figure 2 illustrates the new model, where the model consists of five main phases rather than 4 phases as the following: 1) Establish Connection 2) Sense 3) Analyze 4) Plan and 5) Execute.

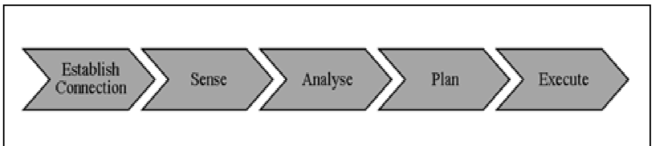

Fig. 2: The "Mission Critical Multiple Client and Single Server Operation for Failure Detection and Recovery System" Model.

In order to understand the process of "Mission Critical Multiple Clients and Single Server Operation for Failure Detection and Recovery System" Model, Figure 3 illustrates the flow of the system in details. In Phase 1, the connection between the clients and the server will be established. Here, the clients will be connected automatically to the server. If there is any failure, the failure will be detected in Phase 2. When the failure is detected, the system will analyze the failure in Phase 3. In Phase 4, the system will take appropriate action for failure recovery based on the findings from Phase 4. In Phase 5, failure recovery process will be executed to ensure the system work successfully between the clients and the server.

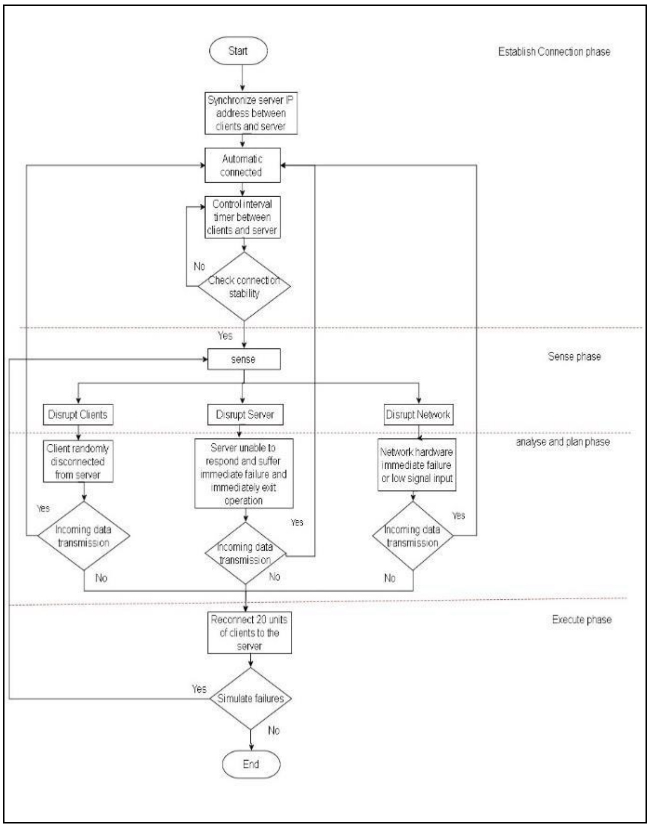

Fig 3: Details flow of the "Mission Critical Multiple Clients and Single Server Operation for Failure Detection and Recovery System" Model.

5. Implementation and Simulation Results

Firstly, there are two hardware that need to prepare and configure (i.e. router and tablet) in order to implement the system. The router acts as a communication device, whereas, the tablet acts as a client and server. In the experimental setup, N750 Wireless Dual Band Gigabit Router TL-WDR4300 is the chosen brand for the router and Microsoft Surface Pro 3 is the chosen brand for the tablet. Here, the router acts as a communication device for the "Mission Critical Multiple Client/ Single Server Operation Failure Detection and Recovery System". The router connects multiple clients and single server in order to simulate various Scenarios for failure detection and recovery using the system.

The router N750 Wireless Dual Band Gigabit Router TL-WDR4300 is used in the experiments (see Figure 4). The router needs to configure correctly before it can connect to the server and the clients. The configuration will follow the TPLINK Router configuration and setup as in Figure 4.

Fig 4: N750 Wireless Dual Band Gigabit Router TL-WDR4300.

There are a few configurations that need to be performed, including: wireless configuration, wireless distribution system (WDS) status, SSID name, wireless channel and Wide Area Network (WAN). In the quick setup page (see Figure 4 (a) and (b)), for instance two wireless configurations 2.4 GHz and 5 GHz is set to SNAKENET24A and SNAKENET5A. However, in the experiments, only the SNAKENET5A is tested for "Mission Critical Multiple Client/ Single Server Operation Failure Detection and Recovery System" as it is lack of interference. Next, WAN port is unplugged as no internet access is required for the experiments.

Fig 4(a): The configuration status of N750 Wireless Dual Band Gigabit Router TL-WDR4300.

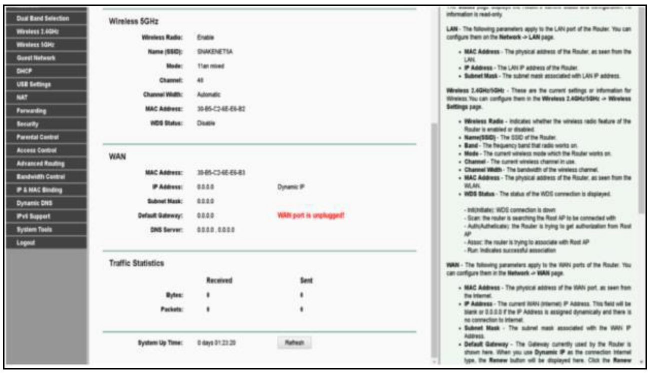

Fig 4(b): The configuration status of N750 Wireless Dual Band Gigabit Router TL-WDR4300.

Next, the tablet acts as the clients and server for the "Mission Critical Multiple Client/ Single Server Operation Failure Detection and Recovery System". The Microsoft Surface Pro 3 tablet is chosen as its' battery life can withstand up to nine hours and its' weight is light as shown in Figure 5. In order to ensure the system is running properly, the tablet will connect automatically to the SNAKENET5A during the experiments.

Fig 5: Microsoft surface pro 3 tablet.

In order to implement the system based on the proposed model, there are 20 clients that were simulated to connect and communicate with a single server using a router. There are three steps to establish the connection between the clients and the server. Firstly, the system will identify the IP address of the server. Secondly, the system will synchronize all the 20 clients with the IP address of the server. Finally, the system will execute the "SenderTestBatch" file. All three steps of the configuration process should be performed before the connection between the clients and the server can establish. The clients and server be able to detect problem failures in the system and reconnect automatically within 5 seconds. All the experiments illustrated below used N750 Wireless Dual Band Gigabit Router TL-WDR4300 router with full gigabit ports for communication to ensure crucial transfer speeds. It also support all the communication protocols between clients and server.

Four sets of experiments and scenarios were performed: 1) interval timer set-up between clients and server, 2) disrupt client, 3) disrupt server and 4) disrupt network. In Experiment 1, the "interval timer" function is used to control the speed of data transmission between the clients and the server. Here, by default the "interval timer" is set to 100 milliseconds. In certain case, the interval timer needs to be set to a certain limit to ensure the stability between the clients and server. For this experiment, the "interval timer" between the client and the server were tested and set to two condition: 1) less than 200 milliseconds and 2) between 200 milliseconds to 1000 milliseconds.

Table 1 presents the results for different setting of the "interval timer". Based on the observation, the results showed that when the "interval timer" was set below than 200 milliseconds, the speed of data transmission between the clients and the server was slower and caused the connection to be unstable. However, when the "interval timer" was increased greater than 200 milliseconds, the data transmission between the clients was stable. In this case, the system must be in a stable state before the following experiments are executed: 1) disrupt clients 2) disrupt server and 3) disrupt network are carried out.

Table 1: Results of the interval timer setting between clients and server using N750Wireless Dual Band Gigabit Router TL-WDR4300.

Interval timer between the clients and the server (milliseconds)	Results
Less than 200	Unstable connection
200> x> 1000	Stable connection

In Experiment 2, the "Disrupt client" function is used to simulate failures by disconnecting the communication between the clients and the server randomly. Before this experiment can be performed, the connection between the clients and the server must be in a stable state. Findings from Experiment 1 showed that the "interval timer" must be set to a certain limit (i.e. more than 200 milliseconds) to ensure the stability of connection between the clients and server. Here, the "interval timer" was set to 500 milliseconds between the clients and the server to control the speed of data transmission between the clients and the server. Figure 6 shows the results after the "disrupt clients" function is executed.

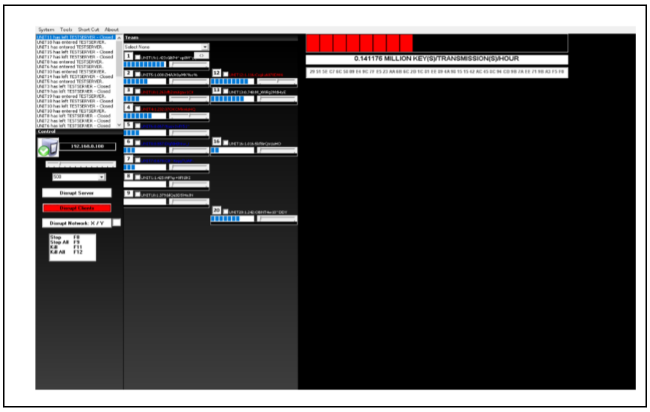

Fig 6: Results when "disrupt clients" function is executed.

When the "disrupt clients" function was executed, a number of clients from the total of 20 clients were disconnected from the server and the system detected the failures. In this experiment, the failures were detected based on the status displayed on: 1) "blue data timer", 2) "slider bar" and 3) "color change for each client". Table 2 shows the results and the systems' status after the "disrupt clients" function is executed.

Table 2: Results and the systems' status after the "disrupt clients" function is executed.

Disconnection Sign	Results
Blue data timer	No movements
Slider bar	Moving to the right end
Color change for each clients	Changing to red color

As the "disrupt clients" executed to the system, the disconnection sign also change. This is because no incoming data transmission between clients and server during operations. The colour change for each client turn to red colour to shows clients disconnected one by one from the server. For example, random clients disconnected become red and the other will remain white colour as long as there is incoming data transmission between clients and server. The system will reconnect the clients to the server again within 5 seconds to recover the failures (see Figure 7). Table 3 shows the results and the system's status after the recovery process.

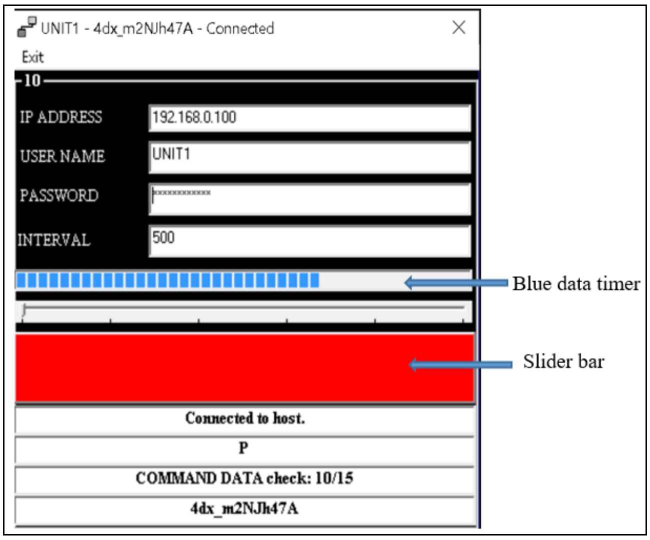

Fig 7: One client connected to the server.

Table 3: Results and the systems' status after recovery process.

Disconnection Signal	Results
Blue data timer	Moving to show the presence of incoming data.
Slider bar	Remain stop
Color change for each clients	No changes

In Experiment 3, the "Disrupt Server" function is used to simulate failures where the server is unable to respond. Here, the server suffers immediate failure and terminates the operation. In this experiment, the "interval timer" was set to 500 milliseconds to control the speed of data transmission between the clients and the server. When "disrupt server" function was executed, a red box was appeared and displayed a message "over flow error arise". Here, the system automatically restarted and reconnected the clients to the server again within 5 seconds to recover the failures. Figure 8 shows the results after the "disrupt server" function is executed.

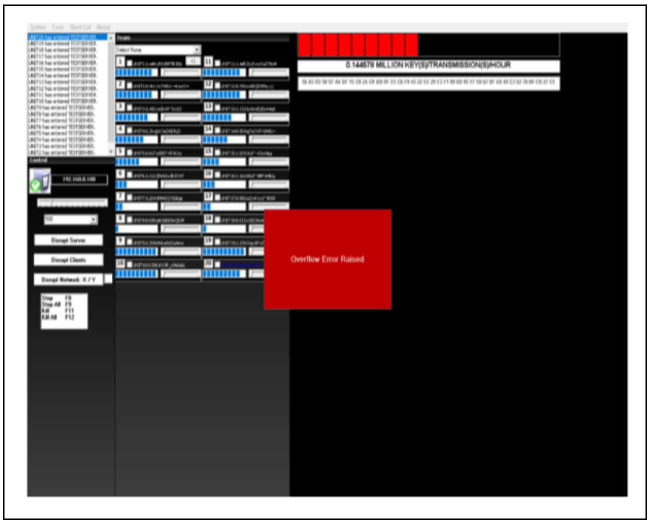

Fig 8: Results when "disrupt server" function is executed.

After the system recover the failures by rebooting and restarting the server, the "interval timer" by default is set to 100 milliseconds. According to previous results, this set-up is inappropriate which caused unstable connection. Therefore, the "interval timer" needs to adjust and set to 500 milliseconds for establishing a stable connection between the clients and the server.

In experiment 4, the "disrupt network" function is used to switch off a network adapter (i.e. "Marvell AVISTA Wireless AC" network controller) that installed in the tablet (i.e. "Microsoft Surface pro 3"). Here, the network adapter was set to zero.

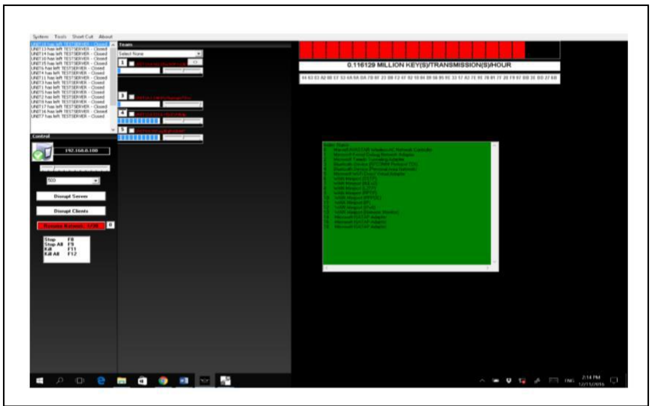

Fig 9: Results when "disrupt network" function is executed.

When the "disrupt network" function was executed, all the 20 clients will be disconnected from the server after 30 seconds (see Figure 9). Next, the network adapter in the tablet was switched off. The system will recover the failures by switching on the network adapter after 30 seconds and will reconnect the clients to the server again within 30 seconds. Table 4 shows the results and the systems' status after recovery process.

Table 4: Results and the systems' status of recovery process.

Network Adapter Set-Up	Time Interval
Time taken to switch off Network Adapter (seconds)	After 30 s
Time taken to switch on the network adapter (seconds)	Within 30s
Time taken to reconnect 20 units of clients (seconds)	Less than 30s

All the four sets of experiment were conducted to cover different scenarios that produced different outcomes such as disrupt clients, disrupt server and disrupt network. As from that, some improvement needs to be considered to overcome the weaknesses.

6. Conclusion and Future Work

The overall research is to propose a new model for the development of "Mission Critical Multiple Client and Single Server Operation for Failure Detection and Recovery System". In this paper a new model for the development of "Mission Critical Multiple Client and Single Server Operation for Failure Detection and Recovery System" is proposed based on SAPE model. In order to suit with the scenario under studied, SAPE model has been modified for this system. The proposed model consists of five main phases: 1) Establish Connection, 2) Sense, 3) Analyse, 4) Plan, and 5) Execute. In the proposed model, an enhancement has been made by adding "Establish Connection" phase. This phase was important to initiate a connection between multiple clients and single server so that the incoming data can exist and the data transmission can form. The advantages of adding "Establish Connection" phase in the proposed model was to ensure the consistency of the simulated system. The consistency of "Mission Critical Multiple Client and Single Server Operation for Failure Detection and Recovery System" are required to ensure every modules and function can be done smoothly during experiments.

In order to test the new proposed model, four sets of experiments were conducted based on four main modules and functions such as: 1) Interval timer between server and clients, 2) Disrupt Client, 3) Disrupt Server, and 4) Disrupt Network. The "Mission Critical Multiple Client and Single Server Operation for Failure Detection and Recovery System" must be in a stable state before the following three experiments can be executed: 1) Disrupt Clients, 2) Disrupt Server, and 3) Disrupt Network. Findings show that the system is able to detect and recover from the three types of failures as long as there is incoming data continuously between the clients and the server.

Based on the current system implementation, there were several works that need to be enhanced for the "Mission Critical Multiple Client and Single Server Operation for Failure Detection and Recovery System" as follows: 1) flexible adjustment of default interval timer between clients and server, 2) introduce multiple clients and multiple server and 3) more simulation of failures in the system. This enhancement is required for the "Mission Critical Multiple Client and Single Server Operation for Failure Detection and Recovery System" to maintain the continuity and capability of the system for mission critical operations.

References

[1] Abhish, K., Rakesh, V. S, "Real-Time Analysis of a Multi-Client Multi- Server Architecture for Networked Control Systems", Halmstad University, Sweden, 2015.

[2] Ahmad, A., Al-Abri, D., Bait-Shiginah, F., Atlanta, J., "Software Testing Education – A Required Knowledge for Software Systems Developers and Managers", Free and Open Source Software Conference (FOSSC-13), pp. 29–33, 2013.

[3] Arshad, N. "A Planning-Based Approach to Failure Recovery in Distributed Systems", University of Colorado, United State, 2006.

[4] Babaoglu, O., Davoli, R., Montresor, A. "Group Communication in Partionable Systems: Specification and Algorithms", IEEE Transaction on Software Engineering, 27(4), 308–336, 1998.

[5] Devalla, B., Sahoo, A., Guan, Y., Bettati, R., Zhao, W., "Adaptive Connection Management for Mission Critical Applications Over ATM-Based Networks", International Journal of Parallel and Distributed Systems and Networks, 3(2), 51–63, 2006.

[6] Fragkiadakis, A. G., Askoxylakis, I. G., Tragos, E. Z., Verikoukis, C. V., "Ubiquitous Robust Communications for Emergency Response Using Multi Operator Heterogeneous Networks", EURASIP Journal on Wireless Communications and Networking, 2011(1), 13, 2011.

[7] Forsyth, P. H., "Design of Dependable Systems on Android", Lund University, Sweden, 2006.

[8] Hole, A., "Data Center High Availability for Integrated Machine Control Platforms", University of Stavanger, Norway, 2014.

[9] Kulkarni, N., "Video Streaming Over Full Duplex Network Using Web Socket and Its Performance Evaluation", Journal of Multidisciplinary Engineering Science and Technology (JMEST), 2(4), 2–5, 2015.

[10] Li, M., "Fault-Tolerant Cluster Management", University of California, Los Angeles, 2006.

[11] Ranjithprabhu, K., Sasirega, D., "Eliminating Single Point of Failure and Data Loss in Cloud Computing", International Journal of Science and Research (IJSR), 3(4), 335–337, 2014.

[12] Rawal, B. S., Berman, L., Ramcharan, H, "Multi-client/Multi-server split architecture", International Conference on Information Networking, pp. 696-701, 2013.

[13] Riihonen, T., Werner, S., Wichman, R., "Comparison of Full-Duplex and Half-Duplex Modes with a Fixed Amplify-and-Forward Relay", IEEE Wireless Communications and Networking Conference, 2009.

[14] Sabadello, M., May, A., "Small group multiplayer games", 2017, https://www.cg.tuwien.ac.at/courses/seminar/ss2001/multiplayer/small_group_multiplayer.pdf.

[15] Saha, G. K., "Approaches to Software Based Fault Tolerance – A Review", Computer Science Journal of Moldova, 13(3), 193–231, 2005.

[16] Sheng, Z., Mahapatra, C., Zhu, C., Leung, V. C. M., "Recent Advances In Industrial Wireless Sensor Networks Toward Efficient Management In IoT", Journal for Rapid Open Access Publishing, 3, 622–637, 2015.

[17] Oluwatosin, H. S., "Client-Server Model", IOSR Journal of Computer Engineering, 16(1), 57–71, 2014.