academicJournals

Vol. 8(33), pp. 1646-1654, 9 September, 2013
DOI: 10.5897/IJPS2012.0893
ISSN 1992 - 1950 © 2013 Academic Journals
http://www.academicjournals.org/IJPS

International Journal of Physical Sciences

Review

On consistency and security issues in collaborative virtual environment systems

Abdulsalam Ya'u Gital[1]*, Abdul Samad bn Ismail[1] and Shamala Subramaniam[2]

[1]Department of Computer Systems and Communications, Faculty of Computer Science and Information Systems, Universiti Teknologi, 81310, UTM Johor, Bahru, Malaysia.
[2]Department of Computer Technology and Network, Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, 43400 Serdang, Malaysia.

Accepted 22 August, 2013

This paper survey security issues in collaborative virtual environments (CVEs) systems. In CVE, multiple users work on different computers which are interconnected through different networks to interact in a shared virtual world. Due to the nature of the geographically disperse users and their connection via different networks, there are numerous security threats that denied fulfillments of most important CVE requirements which have been ignored (e.g. consistency). In this paper, we outlined the types of collaborative virtual environment applications that can be affected by security threats and attacks, it discussed some of the most important CVE systems security requirements, and then discussed the different types of security threats and attacks related to CVE systems security requirements. Finally, we describe the state of the art of CVE system security.

Key words: Collaborative virtual environments (CVE), security requirement, threats, attack.

INTRODUCTION

Currently, because of the explosive growth of the computer and communication technology, many valuable materials in military training systems and manufacturing systems in industries can be shared with each other via internet. Quite a number of researches have been done in computer applications for facilitating collaboration among multiple and distributed users, but rarely people work in the area of security in CVE systems. In CVEs, one of the main research topics is how to efficiently transmit messages to provide scalability, minimized delay, and reliability (Yong et al., 2008). CVEs need to be designed to allow groups of people from a diverse set of organizations and locations to work together easily and securely. Security of such an environment is a crucial issue; this is because of the nature of types of data to be transmitted during collaborative activities. Among many issues in the design and implementation of collaborative virtual environment, the major ones include but not limited to security, scheduling and e-resource discovery (Signh and Signh, 2010).

This paper reviews available relevant literatures on collaborative virtual environments security issues. Most published works consider CVE requirements such as scalability, consistency, reliability, and implement series

*Corresponding author. E-mail: abdulsalamgital@yahoo.com. Tel: +601116380779.

of solutions without considering security issues. Since CVE systems rely solidly on network to perform all transaction, security of these systems are important and weakness in it may lead to unsatisfactory results. All the research conducted on either improving scalability and or reliability exposes the system to many security threats due to the distributed nature of the infrastructures and did not provide security solutions. The paper further introduces collaborative virtual environment and the types of collaborative application that can be affected by security threats, followed by review of CVE security requirements. It further went on to review different types of security threats related to CVE systems. It concludes the work and proposes solutions.

COLLABORATIVE VIRTUAL ENVIRONMENTS

Collaborative virtual environment (CVE) allows participant from distant geographic location to share a common virtual environment including virtual entities and resources maintained by a group of computers which can support effective communication between the users to achieve better coordination tasks. Applications of CVEs include education, massively multiplayer online games (e.g., World of Warcraft), virtual worlds (e.g., Second Life), military training, industrial remote training, and collaborative engineering (Deng and Lau, 2012). As the number of concurrent participants is becoming larger, data exchange between the participants increases, the security of CVE systems is not guaranteed because of the location of the participants which is from different network, and data transmitted must pass through different network before it gets to destination.

There are two types of models mostly use for implementing CVE systems: Client server and Peer-to-peer. Even though client server with a single server cannot scale due to increasing number of users in such a systems (Hu et al., 2011a), but it offer the strongest security, as all important state transitions can be verified and safely stored on the server. The server accepts client input directly. The server has total control over how the CVE state is updated and can take into account any factors deemed relevant (John and Jon, 2010).

In CVE systems, all users share the same virtual space, and each of them is being represented by an entity within the virtual environment. When a user connects to the environment, moves and/or interacts with other entities, the CVE systems require the update to be transmitted in order to update its own state, and to distribute the update of state to other users (Hu et al., 2011b). With the expansion of the scale of applications and the increasing number of users, the security of the systems needs special attention for successful collaboration. Collaboration is often encouraged on the basis that it delivers greater productivity. At the heart of collaboration is the ability of the group to contribute. It is also the case that collaboration is one of a number of different ways of working together and in that sense, it is important to consider its security to protect the integrity and confidentiality of the transmitted data. While it is evident that encouraging collaboration through the use of technology has merit, it is also important to realize that successful collaboration in this day and age requires elements of technology, process and people.

There are two basic foundations of CVEs. At first, 3D virtual worlds provide the three-dimensional view and immersive environment. Second, distributed systems are necessary to offer multi-user and collaborative tools capabilities. CVEs create realistic 3D (virtual reality) displays and provide a rotational capability for views inside, above, beside, or under objects and systems in reduced, normal, or large scale. It makes the significant reduction of the time of new commercial product development and military system operational readiness, and overall development and manufacturing costs (Yong et al., 2008).

Latency which is the time interval from the time a user perform an action to the time other users will noticed the action, represents the quality of service provided to users by the system since it determines how fast changes in the virtual world are noticed to the proper client computer (Reuda et al., 2007). In this case, with the distributed nature of the users, any network that is affected by security threats such as DDoS will notice delay beyond the maximum expected for successful collaboration.

Types of collaborative applications affected by security threats

Collaborative application can be categorized according to the nature of the problem at hand. Most of these collaborative applications fall into the following six groups:

(i) Collaborative work environments (for conducting collaborative work such as military training, engineering design, visualization, documentation, etc.).
(ii) Meetings, seminars and conferences over the internet.
(iii) Simulation of face-to-face contacts where visual quality is critical (such as recruitment interviews, medical diagnoses and remote surgical operations).
(iv) Distance learning environments (for providing course materials, holding a tutorial, carrying out a team project, and conducting an examination).
(v) Networked computer games.
(vi) Leisure and entertainment (including 3D navigation and virtual embodiment) etc.

COLLABORATIVE VIRTUAL ENVIRONMENTS ARCHITECTURE

The most popular architectures used for network virtual collaborative environment design are the well known peer-to-peer architecture and client server architecture with a single server or multiple servers (Mecedonia et al., 1994). These architectures have several drawbacks that require researchers' attention, considering the current types of CVE system handling thousands simultaneously collaborating users. Many researchers contributed a lot to CVE systems in different ways and have achieved a great success for instance, Hu et al. (2011a), Yong et al. (2008), Wang (2011), Hu et al. (2011b), Morillo et al. (2010), Lin et al. (2006), Deng and Lau (2012), Chen and Chen (2006), Li (2011), Lin et al. (2008), Carlini and Ricci (2006), Kulkarni et al. (2007), Sandhu et al. (2011), Ahmed and Shirmohammadi (2008), Chen et al. (2010), Nguyen et al. (2009), Tang et al. (2010), Nguyen et al. (2011), Ta et al. (2010), Shao-Qing et al. (2003) and Hiroki and Yoshitaka (2008), but did not consider security issues which is another factor that may lead to unsatisfactory results in their findings.

Architectures based on networked servers are becoming a de-factor standard for DVE systems (Yong et al., 2008; Reuda et al., 2007). Each client in the system is attached to one of the distributed servers, when a user perform a task, the user computer controlling it sent an update message to the user computer controlling other avatars (Reuda et al., 2007). In order to maintain consistency and update view of the virtual worlds that are linked via different network, security of the link must be guaranteed and free from attack such as DDOS which are common in today's networks. This type of threats can seriously cause inconsistency that may lead to unsatisfactory result as stated previously.

Peer-to-peer architecture

In this communication architecture model, each user sends it update directly to other users. The idea is that all components in the distributed system have the same responsibilities acting both as clients and servers. There is no central server to keep status of the whole system. Each peer maintains its own copy of the virtual environment states and exchanges data directly with other peers (Bu et al., 2007; Berket et al., 2005; Khoury et al., 2007; Pan and Francis, 2004). When a program makes changes to its own database, it sends the update data out so that other programs can update their individual databases (Yong et al., 2008). This architecture has the advantages of low communication latency and fault tolerance capability, for a single client's fault will not cause whole system to crash. Conversely, there is communication complexity with the model as each user has to adopt the filtering algorithm to reduce the consumption of network resources which causes inconsistency of the system (Hu et al., 2011a). These network resources can also be affected by security threats that may result in an inconsistency situation even with the capabilities of the peer-to-peer.

Client server architecture

The client server architecture is classified as either single server or multi server architecture.

Client server with single server

In this model, all the clients' send update to the server; the only common server collects all of the data from the different clients' machine, and sends the results back to each participating client's machine. Each participant's application communicates only with a server that is responsible for passing messages to other clients. Although this model simplifies security implementation, and has a simple data structure to store and handle the data, it is not scalable. Therefore, it is avoided due to increasing number of collaborators. All other models adopted are subjected to a lot of security threats.

Client server with multiple servers

In this model, each client sends updates to the server it is connected to, and the server transmits to other clients and the remaining servers. The management of the virtual environment relies on the several interconnected servers and each server handles a portion of the virtual environment (Hu et al., 2011a). Security of systems implemented using such a model is facing a great challenge; this is because a change or modification by any security treats may lead to a serious error.

From the above description, one notices that servers and the clients execute series of functions to keep the consistency in the virtual environment. At the server side, the server perform the function of receiving the update messages from the clients, updating the whole virtual environment and transmitting updates of the virtual environment to other clients and servers. Moreover, at the clients' side, clients must execute functions of receiving the user's input as the update message, transmitting the update message to the server and receiving the update messages from the server to keep the virtual environment up-to-date. Whenever there are security threats such as Denial of Services (DoS) and or

Distributed Denial of Service (DDoS), the computing and communication resources suffers and the entire system becomes slower and suffers a long time delay, thereby resulting in conflicting virtual world status at a given time.

CVE SYSTEM SECURITY REQUIREMENTS

Computer security can be defined from the aspect of information flow in the networked environment (Corona et al., 2009), currently security in CVE systems in an active research area. Network security is a threat, intrusion, denial of services on a network infrastructure that will analyze your work and gain information to eventually cause the network to crash or to be corrupted. Any network devices that are not being monitored are the main source of information leakage in most organizations (www.ayuverda.hubpages.com). In CVE data such as military attack preparation by group of army from different region, manufacturing system information, etc passed several threads and challenges when it comes to security. Security is a crucial issue when it comes to virtual collaboration because the participants are from geographically disperses location and is connected through different network. Each participant is expected to receive all the transaction. In this transaction, security is a big challenge for reliable and secure transmission of data from and to all participating members during virtual collaboration. The following are security requirements in collaborative virtual environment:

Authentication

This is a mechanism that a user uses to validate data during collaboration with other members of the team. Without this, attackers get access to the data and the data can be modified without the notice of the genuine users. Authentication prevents attackers from getting access to the network and the data on the network (Bullock and Benford, 1999).

Confidentiality

The data transmitted to other members of the collaborating team should only be understood by them. And the system needs to protect the channel transmitting the data so that attackers cannot get access to the data. In this case, both stored data and data in transmission should be protected from attackers (Song et al., 2005).

Integrity

This ensures that the data on transmission process is not deleted or modified by any malicious programs or unauthorized users by the system. The system should be able to inspect viruses and backdoor programs (Salles et al., 2002).

Availability

The network should provide guaranteed services to all participating members at all the time despite attack from attackers. Users should be able to access the system whenever they want to use it because of time critical processes (Yong et al., 2008).

Non-repudiation

The source of all updates or modification should be known and identified by the system. In that case, the system should maintain the origin of the data and the information received (Yong et al., 2008). To ensure all the above security attributes, security assurance has to be put in place. The classification of computer assurance process is as shown in Figure 1.

SOME COMMON NETWORK ATTACK AND SECURITY THREATS THAT CAN AFFECT CVE SYSTEMS

CVE systems share the same internet with all other applications, therefore the different types of attack on the internet forms part of the attacked to be considered while discussing security issues in CVE systems.

Denial-of-service (DoS) and distributed-denial-of-service (DDoS) attacks

Denial of service attack can cause inconsistency that may lead to unpleasant result. A denial of service attack is a type of Internet attack that is aimed at large websites by consuming both computing and communication resources, disruption of routing information and physical network components. These attack results to slow network performance and inability to access any web site among others. A distributed denial of service attack (DDoS) occurs when multiple compromised systems or multiple attackers flood the bandwidth or resources of a targeted system with useless traffic (Gul and Hussaini, 2011).

This type of attack in a time dependent systems such as CVE can cause inconsistency thereby violating the requirement for successful collaboration. In an application like military training, group demonstration of lunching attack and or group study of map area for a mission, it

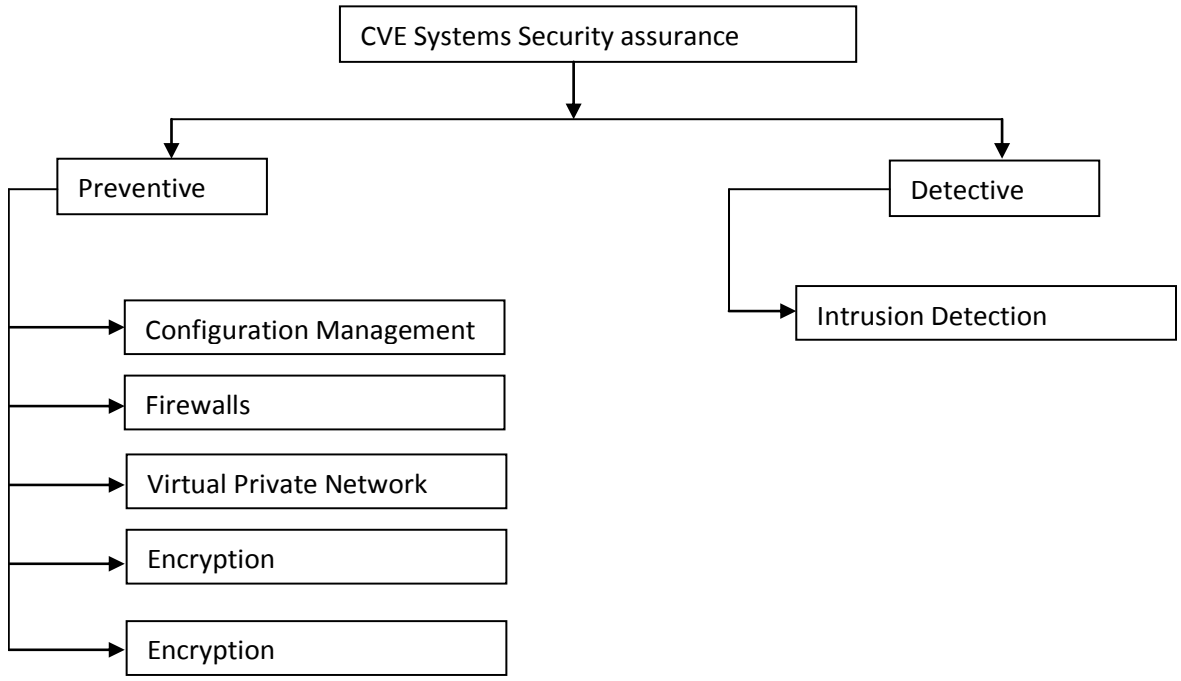

Figure 1. Classification of Collaborative Virtual Environment Security Policy.

may leave among the participant others with out-of-date plan. That may lead to failure or cause serious casualty. Many serious network security problems are caused by Distributed Denial of Service (DDoS) attacks and virus worms-spreading (Davie and Medved, 2009; Desnoyers and Shenoy, 2007). DDoS attacks always paralyze the services which network nodes can provide and occupy the network bandwidth by flooding volumes of traffic to the victims. One attack node may contribute low-rate malicious traffic but attack traffic from widely distributed attack nodes is aggregated toward to the victim (Wang and Huang, 2009; Scaforne, 2007).

Eavesdropping

This is the process of gathering users' machine information such as IP address, the operating system use by the machine and the service the machine is offering in order to launch an attack that is not likely to be noticed by the user. In general, the majority of network communications occur in an unsecured format, which allows an attacker who has gained access to data paths in your network to interpret the traffic (www.ayuverda.com). When an attacker is eavesdropping on your communications, it is referred to as sniffing or snooping. The ability of an eavesdropper to monitor the network is generally a biggest security problem when it cone collaborative military training, manufacturing systems and in Education (On-line examination).

Sniffing

This type of attack generate similar problem or security threat in CVE systems as described in mapping eavesdropping. Packet sniffing is the interception of data packets traversing a network. A sniffer program works at the ethernet layer in combination with network interface cards (NIC) to capture all traffic traveling to and from internet host site. Further, if any of the Ethernet NIC cards are in promiscuous mode, the sniffer program will pick up all communication packets floating by anywhere near the internet host site. A sniffer placed on any backbone device, inter-network link or network aggregation point will therefore be able to monitor a whole lot of traffic.

Most of packet sniffers are passive and they listen to all data link layer frames passing by the device's network interface. There are dozens of freely available packet sniffer programs on the internet. The more sophisticated ones are the once that allow more active intrusion (www.ayuverda.com).

Spoofing

Any internet connected device necessarily sends IP datagram into the network. Such internet data packets carry the sender's IP address as well as application-layer data. If the attacker obtains control over the software running on a network device, they can then easily modify the device's protocols to place an arbitrary IP address into the data packet's source address field. This is known as IP spoofing, which makes any payload appear to come from any source. With a spoofed source IP address on a datagram, it is difficult to find the host that actually sent the datagram.

Hijacking (man-in-the-middle attack)

This is a technique that takes advantage of a weakness in the TCP/IP protocol stack, and the way headers are constructed. Hijacking occurs when someone between you and the person with whom you are communicating is actively monitoring, capturing, and controlling your communication transparently. For example, the attacker can re-route a data exchange. When computers are communicating at low levels of the network layer, the computers might not be able to determine with whom they are exchanging data. Man-in-middle attacks are like someone assuming your identity in order to read your message. The person on the other end might believe it is you, because the attacker might be actively replying as you, to keep the exchange going and gain more information (Anderson, 2007).

STATUS OF CVE SYSTEMS SECURITY

In order to come up with secured, CVE systems, it has been found necessary to evaluate the existing CVE platform base on the security requirements of the CVE systems. Other requirement can also feature for other reference and not for the purpose of this review. The CVE security requirement and other closely related requirements here serve as the evaluation criteria to show whether or not a particular CVE platform satisfies fully these requirements. Many researches to realize scalability, reliability, consistency, responsiveness, extensibility, persistency in CVE have been taking place for more than 20 years, but did not consider security aspect which is vital to any organization. Yet achieving scalability, reliability, consistency, responsiveness in most of the platform is yet to be met. The other entire requirements have effect on the security of the systems. Participant access to CVE objects and information becomes an important topic of discussion because of the growth in the use of CVE. In virtual reality games, storefronts, classrooms, and laboratories for example, the need to control access to spaces and objects is integral to the security of activities in these virtual realms. However, limited access controls are typically available in CVEs (Wright and Madey, 2010). There is a limited number of efforts that deal with security controls in CVE systems (Wright and Madey, 2010).

CVE system such as massively multiplayer online gaming (MMPG) has experienced tremendous growth over the past decade. The number of players, game operators, game designers, and gaming companies with stake in this industry has also increased remarkably (Gupta et al., 2009). As a result, the need for security in MMPGs is becoming increasingly critical. Cheating, virtual frauds, and other security attacks are becoming increasingly widespread in the virtual world (Debbie Jiang, 2011). In 2006, it was estimated that there are more than 10 million people playing MMPGs, with the number doubling every two years (Brian, 2007). One of the most popular MMPGs is Blizzard's World of Warcraft (WoW), which reached a subscriber base of 12 million in October 2010.

Due to the architecture of MMPGs and the large number of participants, there is an inherent lack of security in these games (Figure 1), which creates fertile grounds for cheating. In addition to the inherent security risk, MMPGs are also lacking in terms of legal regulation, security and privacy protection, and other related legislation which can resolve these security issues. As a result, users have taken advantage of this shortcoming and exploited these games through hacks, attacks, and cheats (Debbie, 2011). Virtual Life Network is another system that was designed without security consideration. On the need to secure some basis, security measures are added. That instead led to non-secure solution (Ilja, 2009).

NPSNET-V is a Java-based application with no security beyond the default provided by the Java Virtual Machine (JVM) (Salles et al., 2002). Ernesto et al. (2002) added that an easy assumption upon which to construct networked applications is that any security concern can generally be resolved via existing computer, network and database security mechanisms. Therefore, the desired security level of the application must be ensured by the application itself. Distributed Interactive Simulation (DIS) aims at proposing a common architecture for communication integration and the interconnection of allowing the large scale simulators. After the success of SIMNET (James et al., 1993), DIS (IEEE 1278.1A, 1998) was developed to address the interoperability of heterogeneous simulators. The essence of DIS is the creation of synthetic environment within which humans and simulations interact at multiple networked sites. DIS was not fully distributed; each message must be received

Table 1. State of the art CVE systems.

CVE platform / Evaluation criteria	MMOG	MASSIVE	DIVE	NPSNET	SPLINE	BRICKNET	SIMNET	VLNET	DIS
Scalability	Good	Average	Average	Average	Average	Average	Average	Low	Average
Reliability	Average	Average	Average	Average	Average	Average	Average	Low	Low
Consistency	Good	Average	Good	Average	Average	Good	Low	Average	Good
Responsiveness	Average	Average	Average	Low	Average	Average	Average	Low	Average
Extensibility	Good	Low	Average	Average	Low	Average	Low	Average	Average
Persistency	Good	Low	Average	Low	Low	Average	Low	Low	Low
Security	Low	Low	Low	Average	Low	Low	Low	Low	Average

and treated by each node, which clutter the bandwidth even though not a lot of data is transmitted. DIS does not manage latency and causality that made the reusability of simulations impossible. Latencies were not controlled and no time management service was incorporated which caused data losses due to the rejection of too old packets. That affects the security of the systems because there was systems ware developed without full security considerations.

The current version of MASSIVE is MASSIVE-3. MASSIVE-3 is based on the authors experience on MASSIVE 1,2. According to James et al. (1993), MASSIVE-3 is a multi-user CVE System that supports populated and interactive virtual worlds combining 3D graphics, real time audio and stream video. MASSIVE-3 allows its virtual worlds to be spatially structured as multiple linked locales each of which can be an arbitrary virtual space (e.g. room, building and open region) with its own Cartesian coordinate system. MASSIVE-3 extends the locales by allowing current locales to be linked to recording of other locales.

According to Ta et al. (2010), Distributed Interactive Virtual Environment (DIVE) is one of the most acknowledged Virtual Collaborative System, which is a tool kit for building distributed VR application in a heterogeneous network environment. DIVE allows many users and applications to interact in a real-time through virtual environment. It can also be described as an Internet-based multi-user system that allows remote participants to meet and interact with each other in a virtual 3D space. DIVE was developed at the Swedish Institute of Computer Science. It is one of the early systems that continue to be developed and improved over the years. The DIVE run-time environment consist of a set of communicating processes, running on nodes distributed within a LAN or WAN. The processes which are either a human user or an autonomous application have access to number of databases updating concurrently. The virtual world in DIVE consists of a database containing numbers of description of graphical object. Objects can be added or modified dynamically, and concurrently using a distributed locking mechanism. DIVE uses multicast protocols for the communication simulating a large shared memory for a process group through the network (Chander, 2010; Gupta et al., 2009).

According to Singh et al. (1995), BrickNet enables graphical objects to be maintained, managed, used efficiently, and permits objects to be shared by multiple virtual worlds or clients. A client can connect to a server to request objects of its interest. These objects are deposited by other clients connected to the same server or another server on the network. Depending on the availability and access rights of objects, the server satisfies client requests. BrickNet's object sharing strategy allows users to set-up their own private work-spaces, populated by shared and private objects. BrickNet virtual worlds are not restricted to sharing an identical set of objects. Virtual world manages its own set of objects, some or all of which may be shared with the other virtual worlds on the network. This basic arrangement can be used to implement several types of applications including collaborative, interactive learning systems.

The security of all the above systems lack literature. Table 1 summarizes the state of the art security of CVE systems and other relevant requirements as described earlier. Researchers did less in this area. Now that CVE is applied in many fields to achieve great cost effective group activities even where the participants are far away, the area has gain researchers attention, and it is high time the issues of CVE systems security be researched upon in order to provide workable solutions. In this evaluation, Low implies no security is implemented, Average implies a system with little security consideration and Good represent full implementation of security measures.

Conclusion

Security of CVE systems is becoming a serious topic of

research due to its application in many areas of study. This paper review the general security requirement in CVEs, identify different types of network attacks and security threats related to the different security requirement in CVE systems, and survey the state of the art of some CVE systems security and other requirements. This is because there are some requirements that achieving them without security consideration may lead to unsatisfactory results. However, it is required that a reliable intrusion detection model for CVE systems should be developed and is lacking in literature.

REFERENCES

Ahmed DT, Shirmohammadi S (2008). Performance Enhancement in MMOGs Using Entity Types. pp. 215-229.

Anderson R (2006). Security Engineering: A Guide to Building Dependable Distributed Systems. First Edition. 21:633-678.

Berket K, Essiari A, Thompson MR (2005). "Securing Resources in Collaborative Environments: A Peer-to-peer Approach," in Proc. of the 17th IASTED International Conference on Parallel and Distributed Computing and Systems, 2005.

Brian EM (2007). "Second Life and Other Virtual Worlds: A Road map for Research," International Conference on Information Systems (2007): http://www.bus.iastate.edu/mennecke/CAIS-Vol22-Article20.pdf.

Bu S, Boehm S, Portela M, Jo H (2007). "Collaborative Design Review in Virtual Environment," in Proc. of Korea Computer Congress 2007, C:229-232.

Bullock A, Benford S (1999). "An Access Control Framework for Multi-user Collaborative Environments," In Proc. SIGGROUP Conference on Supporting Group Work, pp.140-149.

Carlini E, Ricci L (2006). Integration of P2P and Clouds to Support Massively Multiuser Virtual Environments. *Science*. www.pap.vs.uni-due.de/MMVE10/papers/mmve2010_submission_3.pd.

Chander VR (2010). An Improved Object Interaction Framwork for Dynamic Collaborative Virtual Environments. IJCSNS.10(9):91-95

Chen J, Grottke S, Sablatnig J (2010). Scalability of a Distributed Virtual Environment Based on a Structured Peer-To-Peer Architecture. Symposium A Quarterly J. Modern Foreign Literatures. pp. 1-8.

Chen Jfa, Chen T.-han. (2006). VON: A Scalable Peer-to-Peer Network for Virtual Environments. Ieee Network, (August), pp. 22-31.

Corona I, Giachinto G, Mazzariello C, Roli F, Sansone C (2009). Information Fusion for Computer Security: State of the Art and Open Issues. Info. Fusion. 10(4):274-284.

Davie B, Medved J (2009). "A programmable overlay router for service provider innovation," in Workshop on Programmable Routers for Extensible Services of Tomorrow (PRESTO), Aug. 2009.

Debbie J (2011). Security Issues in MMOG. ACC 626 Research Paper June, 2011.

Deng Y, Lau RWH (2012). On delay adjustment for dynamic load balancing in distributed virtual environments. IEEE trans. visualization computer graphics, 18(4):529-37. doi:10.1109/TVCG.2012.52.

Desnoyers PJ, Shenoy P (2007). "Hyperion: High volume stream archival for retrospective querying," in *USNIEX*, 2007.

Ernesto J, Sallés J, Bret M, Michael C, Don M, Andrzej K (2002). Security of Runtime Extensible Virtual Environments. *CVE'02*, 2002, Bonn, Germany. ACM 1-58113-489-4/02/0009.

Gul I, Hussaini M (2011)." Distributed Cloud Intrusion Detection Model" Int. J. Adv. Sci. Technol. September, 2011. P.34.

Gupta N, Demers A, Gehrke J, Unterbrunner P, White W (2009). Scalability for Virtual Worlds. Complexity. ICDE '09 Proceedings of the 2009 IEEE International Conference on Data Engineering. pp 1311-1314.

Hiroki O, Yoshitaka S (2008). Assynchronous Collaborative Virtual Environment Support System by Using Revission Tree Presentation Method. IEEE Computer Society, 2008. http://us.blizzard.com/en-us/company/press/pressreleases.html?101007.

Hu X, Liu L, Yu T (2011). A hierarchical architecture for improving scalability and consistency in CVE systems. Int. J. Parallel, Emergent Distributed Syst. 26(3):179-205. doi:10.1080/17445760.2010.49572.

Hu XM, Cai HX, Yu T (2011). A Self-Adaptive Filtering Algorithm Based on Consistency QoS in CVE Systems. Adv. Mater. Res. 225-226:301-306. doi:10.4028/www.scientific.net/AMR.225-226.301.

IEEE 1278.1A-(1998). –"Standard for Distributed Interactive Simulation - Application protocols". E-ISBN 0-7381-0993-2, Print ISBN 07381-0174-5.

Ilja L (2009). VirtualLife Security Infrastructure. Master's Thesis. June, 2009.

James MC, Alan D, Bob G, Paul M, Dale M, Dan O (1993). "The Simnet Virtual world architecture". In VR, pp. 450-455.

John LM, Jon C (2010).. The Near-Term Feasibility of P2P MMOGs. In Proceedings of the 9th Annual Workshop on Network and Systems Support for Games, 2010.

Khoury M, Shen X, Shirmohammadi S (2007). "Peer-to-Peer Collaborative Virtual Environment for E-Commerce," CCECE 2007, pp. 828-831.

Kulkarni S, Douglas S, Churchill D (2007). Badumna : A decentralised network engine for virtual environments. Environments.

Li Y (2011). Determining Optimal Update Period for Minimizing Inconsistency in Multi-server Distributed Virtual Environments. Simulation. doi:10.1109/DS-RT.2011.10.

Lin Q, Zhang L, Ding S, Feng G, Huang G (2008). Intelligent Mobile Agents for Large-Scale Collaborative Virtual Environment. Processing, 7(2):63-72.

Lin Q, Zhang L, Neo N, Kusuma I (2006). Addressing Scalability Issues in Large-Scale Collaborative Virtual Environment. Design, 477-485.

Mecedonia MR, Zyda MJ, Pratt DR, Barham PT (1994). NPSNET: A Network Software Architecture For Large Scale Virtual Environments, 3(4):1-30.

Morillo P, Rueda S, Orduña JM, Duato J (2010). Ensuring the performance and scalability of peer-to-peer distributed virtual environments. Future Generation Computer Systems, 26(7):905-915. Elsevier B.V. doi:10.1016/j.future.2010.03.003.

Nguyen D, Ta B, Zhou S, Cai W, Tang X (2009). Efficient Zone Mapping Algorithms for Distributed Virtual Environments. Technology. doi:10.1109/PADS.2009.10.

Nguyen D, Ta B, Zhou S, Cai W, Tang X, Ayani R (2011). Multi-objective zone mapping in large-scale distributed virtual environments. J. Network Computer Applications, 34(2):551-561. Elsevier. doi:10.1016/j.jnca.2010.12.008.

Pan Y, Francis MT (2004). "A peer-to-peer Collaborative 3D Virtual Environment for Visualization," in Proc. SPIE. 5295:180-188.

Reuda S, Morillo P, Orduna JM, Duato J (2007). A generic approach for adding QoS to distributed virtual environments. Computer Communications. Elservier. 30:731-739.

Salles EJ, Michael JB, Capps M, McGregor D, Kapolka A (2002). "Security of Runtime Extensible Virtual Environments," in Proc. the 4th International Conference on Collaborative virtual environments, P. 97104.

Sandhu UA, Haider S, Naseer S, Ateeb OU (2011). "A Survey of Intrusion Detection & Prevention Techniques" 2011 Int. Conf. Info. Comm. Manage. IPCSIT vol.16 (2011) IACSIT Press, Singapore.

Scaforne KSA (2007). Guide to Secure Web Services. Retrieved from http://csrc.nist.gov/publications/nistpubs/800-95/SP800-95.pdf.

Shao-Qing W, Ling C, Gen-Cai C (2003). A framework for Java 3D based Collaborative Environment. The 8th international Conf. Computer Supported Cooperative Work Design Proc. 2003. pp. 34-36.

Signh M, Signh S (2010). "A Novel Grid-based resource management

framework for collaborative e-learning environments" Int. J. Computer Application. November. 10(4).
Singh G, Serra L, Png W, Wong A, Ng H (1995). BrickNet: Sharing Object Behaviour on the Net. Virtual Reality.
Song J, Kim J, Shin M, Ryu K (2005). "Design and Implementation of Security System for Wargame Simulation System," Korea Information Processing Society, 12-3:369-378.
Ta D, Nguyen T, Zhou S, Tang X, Cai W (2010). A framework for performance evaluation of large-scale interactive distributed virtual environments, (Cit). doi:10.1109/CIT.2010.459.
Tang X, Member S, Zhou S (2010). Update Scheduling for Improving Consistency in Distributed Virtual Environments, 21(6):765-777.
Wang C, Huang C (2009). A Collaborative Network Security Platform in P2P Networks. 2009 International Conference on New Trends in Information and Service Science.
Wang Y (2011). A Fully Distributed P2P Communications Architecture for Network Virtual Environments. Aerospace, pp. 2-5.
Wright TE, Madey G (2010). Discretionary Access Controls for a Collaborative Virtual Environment. 2010. Int. J. Virtual Reality. 9(1):61-71.
www.ayuverda.hubpages.com
Yong S, Moon H, Sohn Y, Fernandes M (2008). A Survey of Security Issues in Collaborative Virtual Environment. IJCSNS. 8(1):14-19.