# A user-layered approach for modelling and simulating terrorist attacks


## Abstract

Modelling and simulation techniques offer a comprehensive method for assessing the impact of a terrorist attack, allowing multiple scenarios to be simulated with ease without imposing extensive time or cost constraints. The aim of this paper is to provide a comprehensive model for the different categories of users who are likely to be involved in the modelling and simulation of a terrorist attack. It focuses on assessing protection measures for implementation within a single building. It takes into account the different types of threat (biological, chemical and explosive) which may occur within or around such a target infrastructure. It considers that a specific company may be required to invest in one or more protection technologies to reduce the risk of an attack occurring, be it a single threat or a combination of threats.



## Keywords

Client-server; Critical infrastructure; Modelling; Scenario analysis; Security system; Terrorist attack


## Introduction

Research into the prevention of terrorist attacks has been a focal point over the past decade, due to the perceived risk of terrorist attacks, both at home and overseas. The capability to assess the risk of these attacks and the effectiveness of security measures within the environment is necessary from both the planning and response perspectives. The recent 'Exercise Forward Defence' drill staged in 2012 in London prior to the Olympic Games, which simulated an explosive attack on the London Underground, provides an example of how one nation perceives the importance of early detection and quick response ("Major exercise in central London - Metropolitan Police Service," 2012). The simulation of attacks in such a manner is limited due to cost, time and cognitive constraints. The aim of this research is to develop an approach for modelling and simulation of terrorist attacks which overcome these constraints, yet allows the involvement and interaction of different stakeholders within the simulation of such an attack.

Various methods have been developed which consider computer modelling and simulation techniques aimed at reducing cost, time and cognitive constraints. This allows prospective attack scenarios to be evaluated quickly and easily, with the potential to take into account terrorist decision-making, considering threats to critical infrastructure resulting from both cataclysmic threats, and gradual and undetectable events that build up over time (Calida and

Katina, 2012). Among such techniques are: agent based modelling, game theory and discrete event simulation.

Agent based modelling (ABM) represents a system as a collection of autonomous decision-making entities (i.e. software objects) called agents. Each agent individually assesses its situation and makes decisions on the basis of a set of rules (Bonabeau, 2002). Agent based models have been used to simulate and analyse interaction-based attack scenarios (Mysore et al., 2006) (Carley et al., 2006) (Bulleit, 2005), by allowing agents to interact with each other and their perceived virtual environment (Lucas et al., 2007). Applications of ABM to counter terrorism include simulating the effect of an attack on a city, focusing on both chemical (Mysore et al., 2006) and biological (Carley et al., 2006) attacks. Multi-agent simulations can be used within the security sector as an insight into the execution of high-resolution simulations, such as war-gaming (Lucas et al., 2007). One of the benefits offered by ABM is that it allows "analysts to quickly build, run and analyse many thousands of simulation experiments over a broad range of input variables" (Lucas et al., 2007). One disadvantage of ABM is that the effects of the interactions between agents are uncertain, meaning that it is difficult to make predictions about the system's interactions. Furthermore both the nature and the outcome of an interaction cannot be determined at the beginning of a request (Jennings and Wooldridge, 2000).

Another use of simulations is achieved by using human beings to recreate (to varying levels of granularity) the circumstances and decision-making process both prior to, during and in the aftermath of a terrorist event (Abhayaratne and Ackerman, 2004). Abhayaratne and Ackerman (2004) (working on the Weapons of Mass Destruction (WMD) Terrorism Research Project at the James Martin Center for Nonproliferation Studies in the US) completed a literature review of manned gaming and simulations of terrorist threats that involve WMD. They investigated roleplaying for entertainment purposes (and how these provide valuable contributions for simulating terrorism in general), wargaming the terrorism threat (towards simulating terrorist attack planning and implementation rather than response efforts) and terrorism response exercises (the best practices of these). Brennan (2002) highlighted one limitation of wargaming as its lack of applicability for analysing the effectiveness of deploying resources, and how this preparation would reduce the consequences of such attacks. He suggested that wargames established the effects that attacks on homeland would have on the deployment process only temporarily, but that very little was done to examine the second- and third-order consequences.

Further studies have investigated modelling terrorist attacks as a discrete event simulation (DES) (Wan, 2007)(Sung et al., 2005). DES consists of a collection of techniques that when applied to a discrete-event dynamical system, generates sequences called sample paths that characterize its behaviour (Fishman, 2001). It assumes that each event occurs at an instant in

time and marks a change of state in the system. Wan (2007) extends this to cover dynamic, asymmetric threat behaviours, enabling the simulation of a wide range of maritime threat scenarios. He uses a discrete-event simulation to simulate a typical port threat response model, testing the adaptive response of asymmetric threats in reaction to port security procedures. He supplements this with a multi-agent system to provide complex adaptive behaviours for the different threats.

Modelling and simulation (M&S) techniques offer a comprehensive method for assessing an attack, allowing multiple scenarios to be simulated with ease without imposing extensive time or cost constraints. However, to date they do not consider the different types of user that are likely to access the software. Sung et al. (2005) do however, suggest an approach which considers two layers of developers within the creation of a war game model: M&S experts, and domain experts. They aim to address the fact that M&S experts do not generally have a deep knowledge of the system to be modelled. Sung et al. provide a framework which allows the M&S experts to develop a war game using DES, representing abstract behaviour of an object, and the domain experts to use an object model to detail the object. It provides a mean for allowing domain experts to first analyse the system specification using UML and then M&S experts to transform this for DES.

Popken (1990) creates a computer model that is designed to be used over time to address a fixed set of repetitive problems for airbase logistics within a simulation environment. It ensures that users do not have to recreate a new model for every problem, by adjusting out the input data and certain parameters. Military systems and terrorist attacks alike are complicated to model because of their large scale and scope. In order to address this problem Popken uses the four-level user concept which assumes that at one end of the scale are pure model users (with no training in M&S, but who can complete simple analysis with a given model) and at the other end are model developers ("with the requisite background to use models, build models, or given enough time, modify existing simulation models" (Popken, 1990)). Level 1: the user level, we assume requires the least amount of expertise. It allows the user to perform single parametric studies with a predefined simulation model and statistical analysis tools. Level 2: the analyst level allows specific objects to be selected from an object-oriented component library. Level 3: the developer level allows building, testing and validation of new M&S objects for lower levels. New objects are defined as specializations of already existing object classes. Level 4: the utility level provides access to the underlying software environment (i.e. the program coding). Popken focuses on military problems and the large, complex systems surrounding this. His M&S environment is based solely on the level of M&S expertise that a user has, and does not account for the different interactions that they may wish to experience with the software. For example, his work does not consider how different individuals interacting with the system may require different information and tools relating to a military problem. Furthermore there is no detail about the interactions between each of the levels of user, assuming that they act on a standalone basis.

To our knowledge there exist no M&S software publications on terrorist attacks concerning potential different user groups of the software, each with different requirements. The focus of this paper is therefore to identify such an approach, based on the development of the tool-target approach for modelling a terrorist attack, outlined by Le Sage, Borrion and Toubaline (2012). The tool-target approach implemented by the authors combines DES with ABM techniques to simulate various sequences of discrete events. Inspired by the principle of action and reaction, the methodology proposes that every change in state happens because a tool (an item used to achieve a tool, without being consumed in the process) impacts upon a target (the object of attention), and that the system evolves into another state according to a probability distribution (Le Sage et al., 2012). The target environment is divided into tri-dimensional zones. Interfaces connect two contiguous zones, and paths are defined between them. The typological network is modelled by introducing nodes in the middle of these paths. At each node, a number of pre-specified actions (processes) can occur. The model can be used to provide relatively realistic determination of the risk of potential terrorist attacks modelled as dynamically changing scenarios. It allows existing and hypothetical security systems to be integrated easily, and allows the determination of risk of a malfunctioning security system to be estimated.

The methodology presented within this paper is based on a user-layered approach which has been applied to the model described above. The aim is to provide a comprehensive model for the different levels of users who are likely to be involved in the M&S of a terrorist attack. The proposed design is a multi-layered approach which uses client-server architecture. A major advantage of a client-server architecture is the separation of affairs, thus enhancing modifiability as the "presentation, business and data management logic are all clearly encapsulated. Each can have its internal logic modified in many cases without changes rippling into other tiers" (Gorton, 2011) (modifiability), high performance and scalability. However upon the implementation of such an approach the security threats to the client-server system must be considered. Client-server security problems can arise due to (i) physical security holes (when an individual gains unauthorised access to an individual's computer by obtaining their user name and password); (ii) software security holes (resulting from a bug in the software which may compromise the system into giving wrong performance); or (iii) inconsistent usage holes (when two different usages of a system contradict over security point) (Yadav and Singh, 2009). Threats to the server should be considered within this implementation, and ways in which these threats can be minimised effectively addressed.

The Transport Layer Security (TLS) protocol was designed to allow client-server applications to communicate in a way that is designed to prevent eavesdropping, tampering or message forgery (Dierks, 2008). The protocol uses data encryption and integrity checks to provide

connection security, and can be used for encapsulation of higher level protocols. Since the introduction of this protocol much work has been completed into ensuring the security of client-server architectures (Wu and Tseng, 2010) (Jain et al., 2012) (Bloch et al., 2008). Lien et al. (2011) aim to integrate security considerations in client-server architectures of health information systems. This they achieve by specifying logical rules for security system requirements. Based on these rules they incorporate security control components at different points in the system architecture.

The remainder of this paper considers the design of a user-layered approach within the tool-target based methodology developed by Le Sage et al. (2012) for modelling and simulating a terrorist attack. It will first seek to identify the requirements of the individuals associated with the M&S process, and then propose a solution to incorporate these individuals and their individual requirements through a client-server platform.

**The modelling and simulation software user groups**

The proposed methodology focuses on assessing protection measures for implementation within a single building. It takes into account the different types of threat (biological, chemical and explosive) which may occur within or around such a target infrastructure. It takes into account the nodal network of the building, which can be modelled according to the building floor plan, and the objects within the structure. It considers that a specific company may be required to invest in one or more protection technologies to reduce the risk of an attack occurring, be it a single threat or a combination of threats.

Scenario-based design was employed to allow users and developers to work together to determine the requirements of a system. Scenario-based design is "a family of techniques in which the use of a future system is concretely described at an early point in the development process. Narrative descriptions of envisioned usage episodes are then employed in a variety of ways to guide the development of the system that will enable these use experiences" (Rosson and Carroll, 2002). A scenario describes an expected or supposed sequence of actions. The scenario-based design approach focuses on the operational modelling of the system.

The framework that has been adopted for the scenario-based design is represented in Figure 1. It was adapted from Rosson and Carroll (2002). The initial step in the proposed framework was to analyse stakeholders' needs and their claims about current practice, to enable the problem scenarios to be generated. Once this problem scenario is created, detailing the context and a set of goals, it can be combined with a collection of background information and used as an input to generate the activity scenarios. Activity scenarios focus on how to

address the goals of the actors in the problem scenario, and are generated by the stakeholders of the project. These activity scenarios can then be combined with analysis of the information security requirements and used to develop conceptual models. After the conceptual models are established, the user interface can be developed, through the use of information and interaction scenarios. Upon the completion of the user interface development the prototype can be created and evaluated by stakeholders against the usability specification.

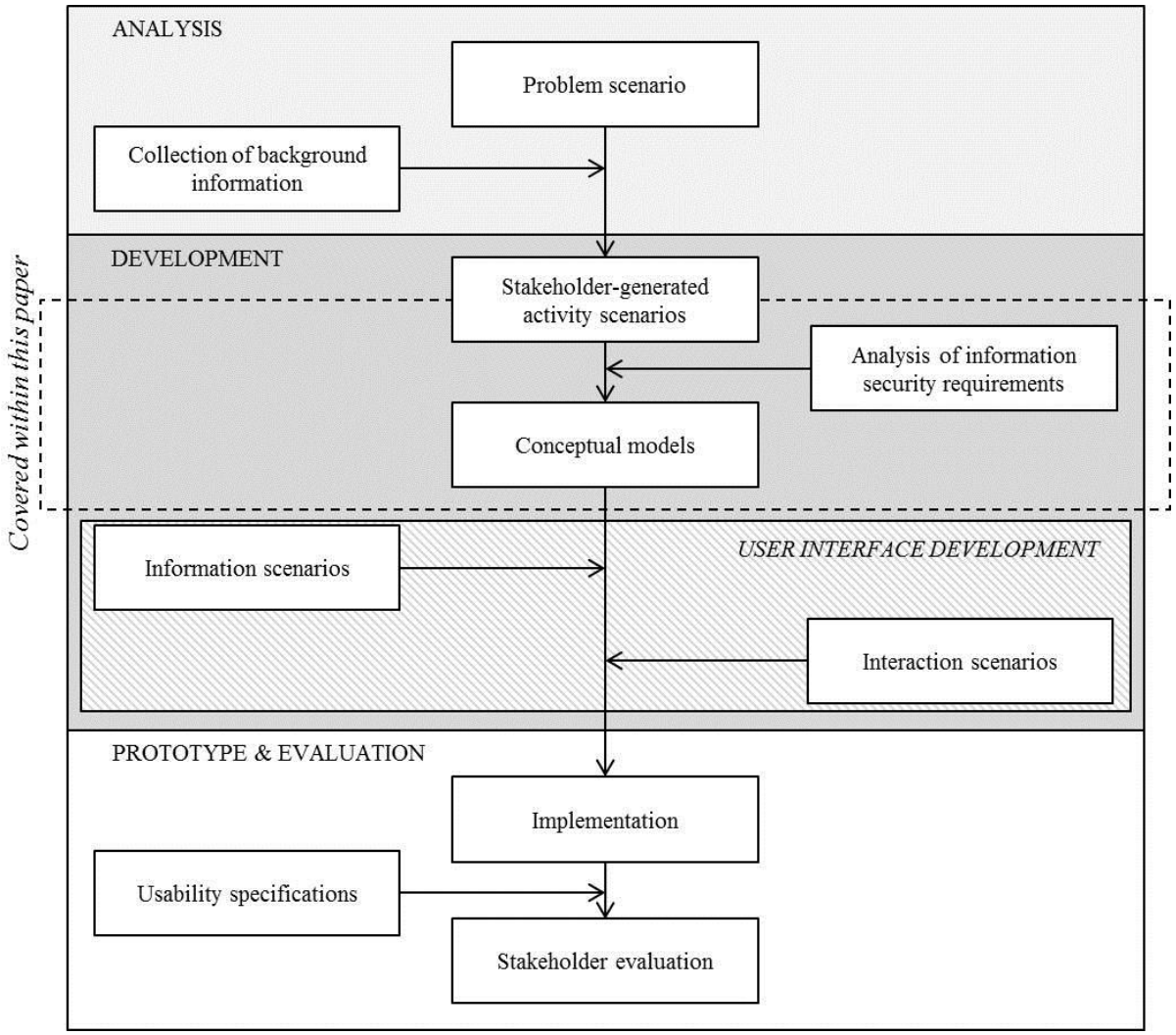


Figure 1: Adaptation of the Structured Scenario-Based Design Method model.

The scenarios used as a basis for this method of design consist of the following; a setting or situation state, one or more actors with personal motivations, knowledge and capabilities, and tools and objects that the actors can manipulate (Rosson and Carroll, 2002).

The user groups were selected such that they broadly represented the potential users of attack M&S software. The groups included the head of security for Company X, security consultants, security design and maintenance teams and software developers. The head of

security for an organisation aims to maintain a safe and secure environment for all users of the building by establishing and enforcing security policies and procedures, and overseeing the processes and their functioning. Security consultants specialise in security analysis, determining the needs of organisations on an individual basis. They are responsible for tailoring security systems to suit an individual organisation, and propose the overall system architecture for the security. Security system designers are responsible for designing and manufacturing technological security systems, which can be tailored to individual organisations, and maintenance teams are responsible for the installation and carrying out any planned or unplanned maintenance required. Software developers are responsible for developing information systems for the analysis of security risk in an organisation, i.e. designing, developing and possibly installing software solutions.

Three scenarios were drawn up to highlight the requirement for the user layered approach:

Activity scenario 1: The head of security for Company X wishes to install a new security system to meet the requirements of his organisation. He wants to install the optimal product to ensure control over the assets of the organisation. As such he employs a security consultant to analyse which product he should install. The security consultant must liaise with the product developer to ascertain the specification of each product, thus allowing him to analyse the conformity to the organisation's requirements.

Activity scenario 2: A new premises is being developed and the security consultant must determine what security measures should be installed to provide the "optimal" security, without compromising the organisation's requirements. They must liaise with the product developers to obtain the system specifications, and must analyse how the different measures can be used in cohesion.

Activity scenario 3: A problem has arisen with the security system in place at Company X. The head of security makes the decision to call out the maintenance team. The maintenance team must assess the security system and produce a report highlighting the problems. The report must be analysed by a security consultant to determine whether Company X should keep the current security system in place, or replace it with an alternative security solution.

**The requirement for a database management system**

In order to achieve their goals the different users must be able to interact with, and access information provided by, the other users. Figure 2 illustrates how the software developer (who is part of a software development company) must be able to interact with the external security consultant. The software developer will request additional generic components which are required within the software.

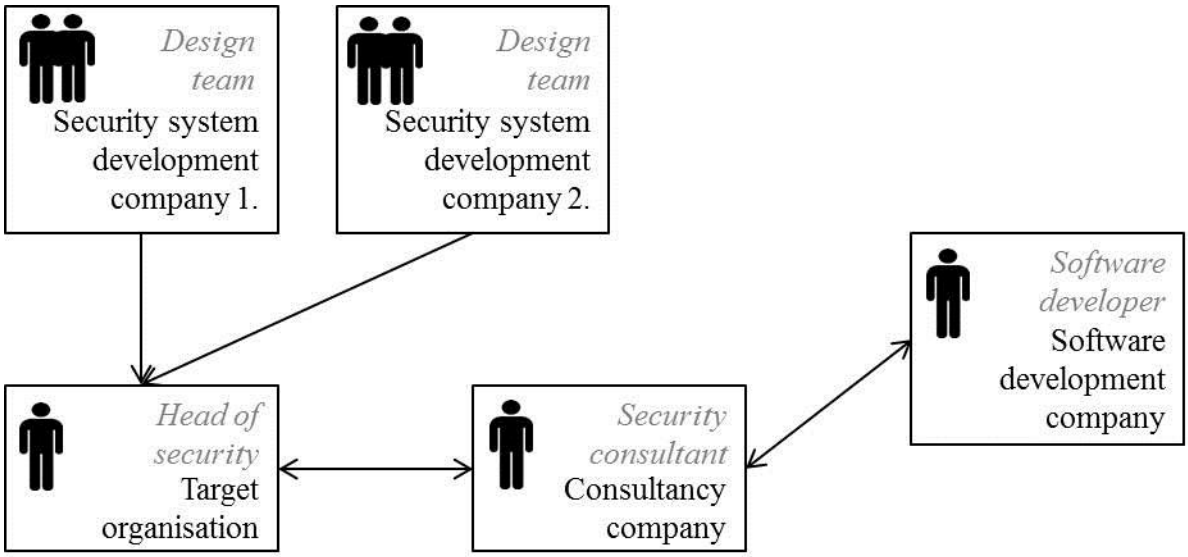


Figure 2: Interaction between the different user groups.

In turn the target organisation communicates with the security consultant at the consultancy company to specify any further information that they require within the model. The security system development companies provide the head of security with specifications for actual or conceptual security designs, which they would like to assess the effectiveness of. The head of security communicates with the security system development company in order to obtain the information required with regards to the security system specifications. The information provided by the consultancy company and the security system design teams are combined to allow the head of security to simulate the risk of an attack occurring.

To facilitate this process and the communication between the different levels, the information may be stored and accessed from a central server. The information required throughout the process includes elements of the tool-target approach, the building floor plan, threats, the network model of the building, the building ecosystem and script progression probabilities, and can be seen in Figure 3. The developer is required to have access to the elements of the tool-target based approach, including the different classes and the attributes and functions associated with each class. The security consultant needs access to the target specific floor plan, and a database containing the threats which they can configure for the assignment. They use this information to aid in the construction of the nodal network of the building, define the

objects within the building and the state of these objects (for example, all doors are left open), and determine the probability of progression along the script. Ultimately, this information allows the security system design team to choose the appropriate security solution for the specific scenario. The same information is required by the head of security for Company X, but with an updated nodal network of the building, building ecosystem and script progression probabilities, to include the proposed security solution.

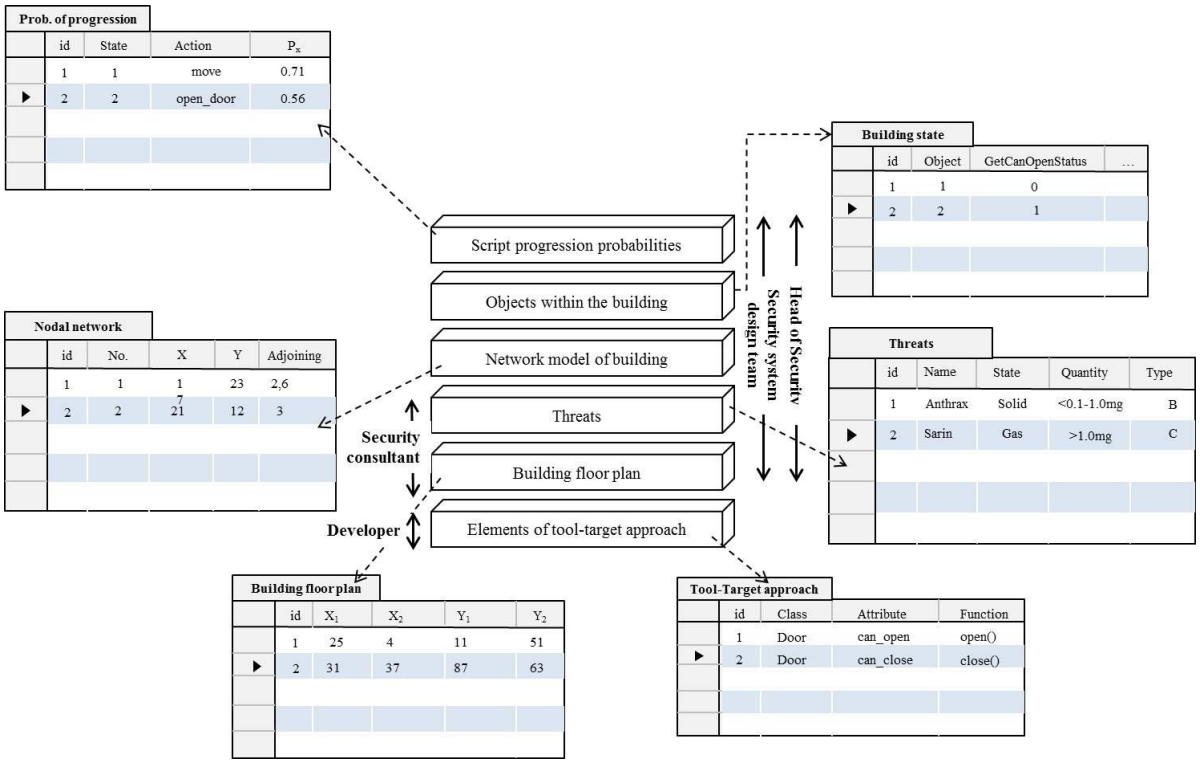


Figure 3: Information required by each user group

## Analysing the information security requirements of the system

The information security requirements of the system were analysed concurrently with the problem scenario to obtain security classifications for each level. "Information security classification is a fundamental step in protecting against the risks associated with unauthorized disclosure, use or loss of […] assets. All security countermeasures to protect the information are determined by its security classification level" (*Information Security Classification Framework*, 2010). Three levels of sensitivity have been defined within this paper, adapted from, and in line with the Information Security Classification Framework (*Information Security Classification Framework*, 2010): High sensitivity, medium sensitivity and low sensitivity.

Information ranked as having a "High" security classification is deemed to provide essential information for a terrorist planning an attack. Such information would include crisis

communication during emergencies and provincial response plans and logs, target specific intelligence and information systems used for testing essential services. Information ranked as having a "Medium" security classification would provide information to terrorists which may be useful in planning an attack, but which would not be of much use as a standalone component without additional resources to accompany it. For example, evacuation and emergency response plans for a specific target would only be useful if the offender could relate this to a specific target infrastructure. The lowest level of security classification would contain information which could reasonably be expected to provide little or no assistance to an offender. This would include material suitable for public release, operational procedures related to non-critical activities and public education materials.

The above provides a platform for determining the most critical aspects of information to protect in the client-server design. Threats to the server include eavesdropping (silently listening to the data over the network in order to obtain a transcript of the network activity and thus obtain sensitive information, such as password, data and procedures for performing functions) and denial of service (damaging a resource such that it cannot be used) (Yadav and Singh, 2009). A client-server architecture must be implemented which takes into consideration these threats when considering the building floor plan and the nodal network of the building.

**Designing a client-server architecture**

The model is based on a 2-tier client-server architecture pattern, which allows the client to talk directly with the server. It separates the user interface and data logic onto one computer (tier 1) and the database server onto another computer (tier 2). Data logic is defined as "the set of rules and specifications that data or queries must meet before being submitted to the database server. The data logic tier may also include calculation, data analysis or anything that does not specifically relate to the user interface" (Hemmer, 1997). The server stores four individual databases as illustrated in Figure 4; the main, target specific, security system and high-level repositories.

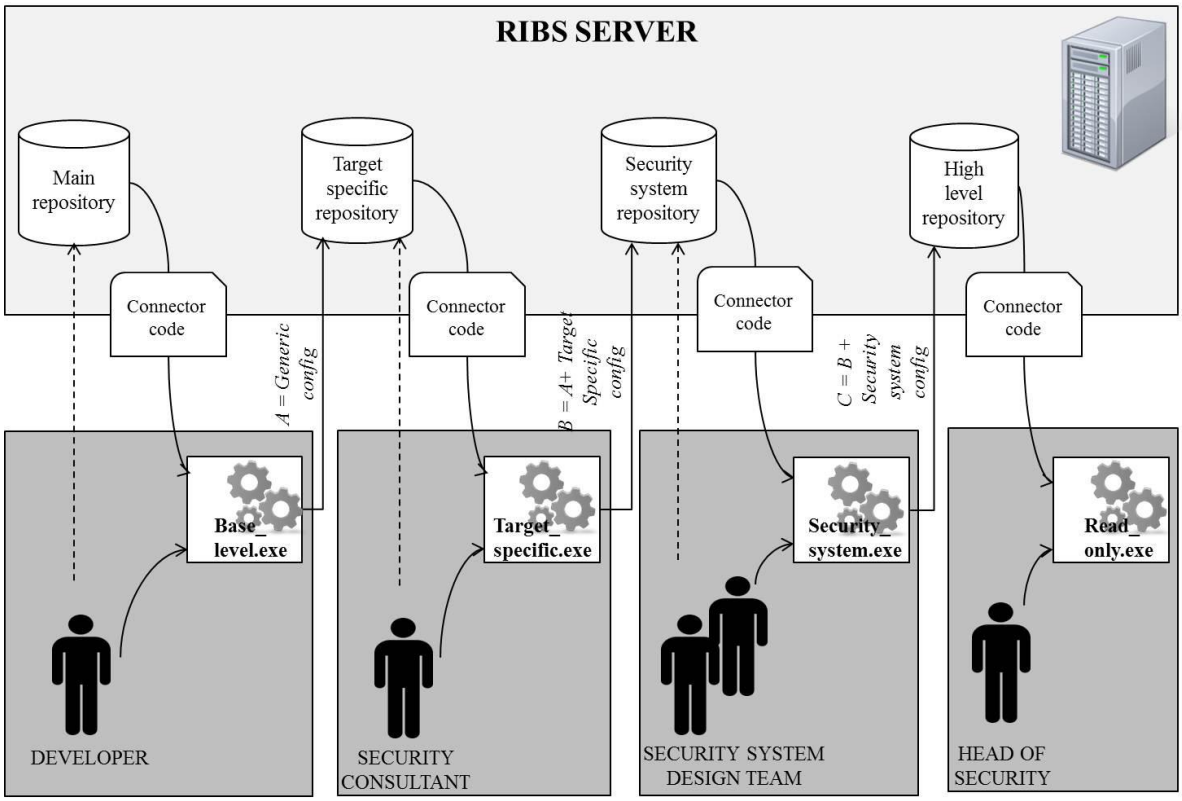


Figure 4: Client-server approach for attack modelling and simulation software.

The developer client has access to the main repository, which stores information relating to the generic components of the software. The main repository is not considered to be security sensitive, as it does not detail any information pertaining to individual structures – this level provides the platform for the implementation. The information from the main repository is fed into the base level executable. The executable allows the developer to format the structure of the HMI, and to implement and make adjustments to the objects within the tool-target based approach. For example, the developer may feel that it is necessary to create a new class entitled "alarm", along with associated attributes and functions. This would allow scenarios to be simulated encompassing these attributes and functions. An example scenario could be as follows; an individual will attempt to park a motorcycle near an assembly point. A few minutes later, they will enter into the building and attempt to trigger the fire alarm in order to cause an evacuation of the building. The motorcyclist intends to trigger the device when a crowd will have gathered at the assembly point.

The results from the base level executable are passed to the target specific repository. The security consultant also inputs information into the target specific repository. The security consultant is responsible for inputting information with regards to the target infrastructure, including the layout of the building and the types of threats which are to be considered. The target specific repository contains sensitive information with regards to the individual

infrastructure to be studied. Dependent on the sensitivity of this information it is therefore recommended that the server is created as a local server, with access limited to one PC. In order to achieve this, the results from the base level executable would have to be stored in an intermediary on the server, allowing the security consultants access to this, such that they can compile this within their on-site target specific repository. The data stored in the target specific repository is passed to the target specific executable. This software allows the security consultant to calibrate the target system, and to mark out the nodal network which the crime scripts are to be simulated on. The target specific configuration, combined with the general configuration, is passed to the security system repository, which is accessible by the security system design team. At this level the design team input data with regards to the security systems which are to be implemented within the target building. The security system repository may also be stored on a local server to prevent unauthorised access to the data.

Finally, all of the above information is transferred into the high level repository, which is connected to a read only executable. Using the repository and the executable the head of security is able to simulate crime scripts to provide insight with regards to the implementation of varying security systems within their building, and furthermore the effects of the potential malfunctioning of each of these security systems. The architecture described must be converted into a platform for implementation, taking into consideration how the system will cover security implementation measures. The outline for the implementation of this platform will be discussed in the following section.

**A platform for implementation**

The user will log themself into the client through the RIBS login interface component by entering an ID number and password. If they are successful in passing the authentication function then they proceed to the interface of the attack simulation software as outlined in Figure 5, otherwise their login is rejected. The attack simulation software component then accesses data from a database located in the server site, dependent on the access level granted to the individual. This server site may be situtated within a local host to limit the potential security breaches within the system. This would only be effective if the different users are located within a similar geographical area, or for high-value targets the different users are willing to travel to a central location to compile and exchange repositories. In the event that the users are not located within a similar geogrpahical area then the security of transmission between the client and the server must be considered. It is envisaged that the client-server transmission will be protected with TLS, and additional privacy controls  including audit trails and encryption.

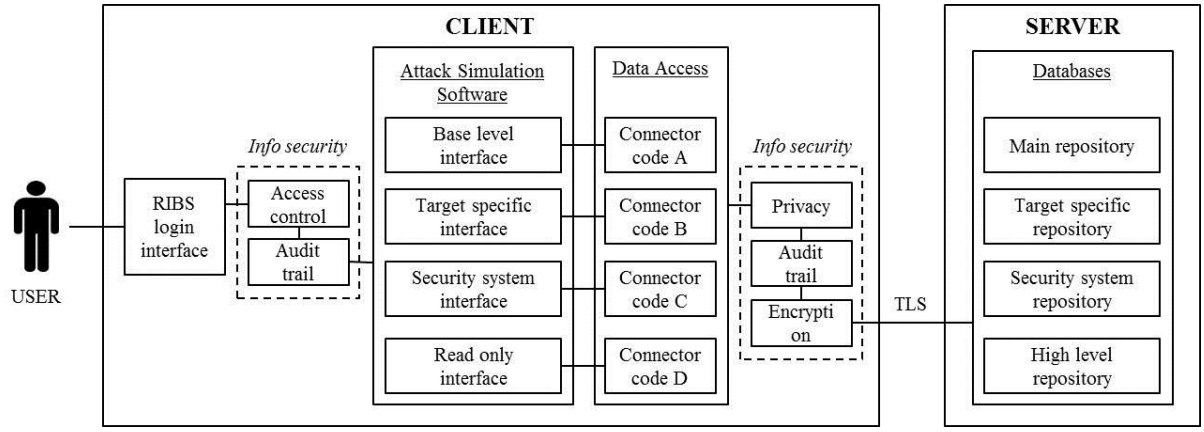


Figure 5: Client-server platform for implementation

## Conclusion

The method proposed within this paper describes a user-layered approach developed for the implementation of terrorist attack M&S software. It allows multiple stakeholders involved in the development and implementation of security systems to interface with a single modelling environment, each with their own prescribed needs. It accounts for the different levels of technical expertise which exist amongst the different stakeholders, and the interactions that must be facilitated between the groups. It considers opportunities for assessing how new security systems could reduce vulnerability, and the effect that a malfunctioning security system may have on the risk of an attack occurring. It addresses the problem that one user is unlikely to have all the knowledge required to implement such a platform, aiming to create a comprehensive and representative solution.

Four different stakeholders of the software have been identified: the software developers, the security consultant, the security product manufacturer and the head of security for a particular company. It has been proposed that in order to effectively assess the implementation of new security technologies, all four groups of stakeholders are required to have an input. As such the software developer would be responsible for ensuring that the underlying software environment is configured correctly. The security consultant would configure the software to ensure that it is specific to a given company. The security product manufacturer would provide the specifications required for the new security system implementation which is to be tested, and the head of security would be responsible for assessing the overall effectiveness of this new security system within his building. It thus aids in the organisation's acquisition of appropriate security solutions, and that they are procured at the best possible cost to meet the needs of the organisation (in terms of quality and quantity, time, and location). In order to ensure that such technology is tested effectively, without actual implementation, communication is required across the different layers of stakeholders. This is enabled by the implementation of a client-server approach which accounts for the possible different geographical locations of the stakeholders. The aim of the platform is to integrate security

concerns into the client-server architecture. One limitiation of the approach is the perceived security threat to the stakeholders involved due to the fact that they are sharing their information across a server. A solution has been proposed which would mean that the most sensitivie information is not shared on the server, and is made available locally to the required parties. Further work required will include the implementation and testing of the platform. User testing will be set up to validate the efficiency of the proposed system.

## References


Abhayaratne, P., Ackerman, G., 2004. Manned gaming and simulation relating to terrorism and weapons of mass destruction: a review of the literature. Defense Threat Reduction Agency.

Adler, R.M., 1995. Distributed Coordination Models for Client/Server Computing. Computer 28, 14–22.

Bloch, M., Narasimha, R., McLaughlin, S.W., 2008. Network Security for Client-Server Architecture Using Wiretap Codes. IEEE Transactions on Information Forensics and Security 3, 404 –413.

Bonabeau, E., 2002. Agent-Based Modeling: Methods and Techniques for Simulating Human Systems. Proceedings of the National Academy of Sciences of the United States of America 99, 7280–7287.

Brennan, R., 2002. Protecting the Homeland.

Bulleit, W., 2005. An agent-based model of terrorist activity.

Calida, B.Y., Katina, P.F., 2012. Regional industries as critical infrastructures: a tale of two modern cities. International Journal of Critical Infrastructures 8, 74.

Carley, K.M., Fridsma, D.B., Casman, E., Yahja, A., Altman, N., Chen, L.-C., Kaminsky, B., Nave, D., 2006. BioWar: scalable agent-based model of bioattacks. Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on 36, 252 – 265.

Dierks, T., 2008. The Transport Layer Security (TLS) protocol.

Fishman, G.S., 2001. Discrete-Event Simulation, 1st ed. Springer.

Gorton, I., 2011. Essential Software Architecture, 2nd ed. 2011. ed. Springer.

Hemmer, F., 1997. Two tier client/server database development for alignment data at the reletavistic heavy ion collider and alternating gradient synchrotron. Presented at the 5th International Workshop on Accelerator Alginment, Argonne, Illinois.

Information Security Classification Framework, 2010. . Province of British Columbia.

Jain, M., Jain, G., Vasavada, J., Patel, P., Ojha, V., 2012. Information security and auditing for distributed network, in: 2012 International Symposium on Instrumentation Measurement, Sensor Network and Automation (IMSNA). Presented at the 2012 International Symposium on Instrumentation Measurement, Sensor Network and Automation (IMSNA), pp. 364 –367.

Jefferson, T., Harrald, J., Fiedrich, F., 2012. Linking infrastructure resilience to response requirements: the New Madrid Seismic Zone case. International Journal of Critical Infrastructures 8, 22.

Jennings, N.R., Wooldridge, M., 2000. On agent-based software engineering. Artificial Intelligence 117, 277–296.

Le Sage, T., Borrion, H., Toubaline, S., 2012. A tool-target based approach for simulating a terrorist attack. Presented at the IEEE Conference on Technologies for Homeland Security.

Lien, C.-C., Liu, C.-W., Chen, Y.-A., 2011. Integrating Security Considerations in Client Server Architectures of Health Information Systems Development, in: 2011 Fifth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS). Presented at the 2011 Fifth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), pp. 527 – 531.

Lucas, T.W., Sanchez, S.M., Martinez, F., Sickinger, L.R., Roginski, J.W., 2007. Defense and homeland security applications of multi-agent simulations, in: Simulation Conference, 2007 Winter. pp. 138 –149.

Major exercise in central London - Metropolitan Police Service [WWW Document], 2012. URL http://content.met.police.uk/News/Major-exercise-in-central-London/1400006754161/1257246745756 (accessed 4.19.13).

Mysore, V., Narzisi, G., Mishra, B., 2006. Agent modeling of a sarin attack in Manhattan. Presented at the Autonomous Agents and Multiagent Systems, Japan.

Popken, D.A., 1990. Object-oriented simulation environment for airbase logistics. Air Force Human Resources Laboratory, Air Force Systems Command.

Rosson, M.B., Carroll, J.M., 2002. Scenario-based design, in: Jacko, J.A., Sears, A. (Eds.), The Human-computer Interaction Handbook. L. Erlbaum Associates Inc., Hillsdale, NJ, USA, pp. 1032–1050.

Sung, C., Hong, S.-Y., Kim, T., 2005. Layered approach to development of OO war game models using DEVS framework. Presented at the Summer Computer Simulation Conference, Philadelphia.

Wan, C., 2007. Discrete-Event Simulation with Agents for Modeling of Dynamic Asymmetric Threats in Maritime Security (Master's Thesis). Naval Postgraduate School, Monterey.

Wu, T.-Y., Tseng, Y.-M., 2010. An efficient user authentication and key exchange protocol for mobile client–server environment. Computer Networks 54, 1520–1530.

Yadav, S.C., Singh, S.K., 2009. An Introduction to Client/Server Computing. New Age International Pvt Ltd Publishers.