MODELLING INTRUSION DETECTION IN SECURITY SYSTEMS

Zenon Chaczko[(a)], Ryszard Klempous[(b)], Chris Chiu[(a)],

[(a)] Faculty of Engineering and Information Technology, University of Technology, Sydney,
15 Broadway, Ultimo, NSW, Australia, 2007.
[(b)] Institute of Computer Engineering, Control and Robotics, Wroclaw University of Technology,
11/17 Janiszewskiego Street, 50-372 Wroclaw, Poland

[(a)]Zenon.Chaczko@uts.edu.au, [(b)]Ryszard.klempous@pwr.wroc.pl, [(a)]Chris.Chiu@uts.edu.au

ABSTRACT

With the ever increasing wireless connectivity and complexity of infrastructure-oriented systems, security is a very important issue for all network-based infrastructures in a modern enterprise environment. There are many examples of intruders and attackers who made successful attempts to seriously affect assets of high-profile organisations and companies. In some cases, the entire infrastructure of a company was brought down, resulting in a prolonged disablement of business, provided services, loss of money and reputation. Many methods, techniques and tools have been developed to secure the assets, network infrastructure and communication of various organisations. Intrusion detection is a relatively new addition to such methods, hence the tools that use advanced intrusion detection techniques started to appear only recently. Using intrusion detection methods, we are able to collect and use information from known types of attacks and find out if someone is trying or even attempting to penetrate our infrastructure or attack our assets. The information collected and accumulated in this way can then be used to harden the infrastructure/network security of the organisation/enterprise and possibly can be used for legal cases. This paper aims to present various issues related to security services, describes methods, techniques used in intrusion detection that can be useful when building and designing effective security systems.

Keywords: modelling methodology, intrusion detection, security system, decision support system

I. INTRODUCTION

There are two main methods of intrusion detection based on the approach to event analysis, signature-based detection and anomaly based detection. Brief descriptions of their functionalities are explained, along with supporting advantages and disadvantages in the subsequent sub-sections.

1.1. Signature-based Detection

This methodology is similar to the way many anti-virus programs incorporate virus signatures to recognise and restrict access to infected files, programs, or active web content from entering a computer system. The attack signatures in this case are comprised of network traffic and activity patterns relating to known attacks. Signature detection is the most widely used approach in commercial intrusion detection systems.

1.1.1. Disadvantages

Signature-based detection scrutinizes ongoing traffic, user activity or transactions, and behaviours looking for matches with known patterns of events specific to known attacks. This intrusion detection system requires access to a current database of attack signatures, and is able to actively compare and match current behaviour against the large collection of signatures. This technique works extremely well for categorised and previously known attacks. Signature databases will require constant updates to be more reliable. Contrary to this, if signature definitions are too specific, this will induce another disadvantage in missing variations of known attacks. This technique is similar to the approach taken to modify polymorphic viruses, where hackers in this case, create new attacks by changing steps in existing known attacks rather than creating entirely new ones. In addition, signature-based detection causes bottlenecks in performance, when current behaviour matches multiple or numerous attack signatures, either in whole or in part.

1.2. Anomaly-based Detection

This methodology operates in a similar fashion as anti-virus heuristics checking, but in this case it uses rules or predefined concepts about “normal” and “abnormal” system activity. These rules can also be referred to as heuristics, and can distinguish anomalies from normal system behaviour. They monitor report on, or block anomalies as they occur. Some intrusion detection systems only support limited types of anomaly detection. Most experts 0, 0 believe this type of detection methodology will become more popular as artificial intelligence becomes more prominent 0, 0, 0 Anomaly based detection examines ongoing traffic, user activity or transactions, and behaviour looking for anomalies on networks or systems that may indicate attack. Based on the premise that “attack behaviour” differs considerably from “normal user behaviour”, detection of intrusions can be carried out by cataloguing and identifying the differences involved. This makes it possible to create baselines of normal behaviour, and allow the anomaly based IDS to observe when current behaviour deviates statistically from the norm. The advantage of this, in theory, provides anomaly based intrusion detection systems the ability to detect new attacks for which signatures are yet to be established.

1.2.1. Disadvantages

A major drawback to anomaly based detection stems from the fact that normal behaviour is dynamic and changes readily and easily. This method of detection is prone to false positives – where attacks may be reported based on simple changes to the norm rather than representing real threats. Another disadvantage is that the intense analytical characteristics of this methodology often impose extreme processing overheads for systems they are running on. In addition, anomaly based systems require considerable time to create statistically sounding baselines; during this period they are vulnerable to intrusion.

2. PHYSICAL SECURITY DISRUPTION

The purpose of this paper is to demonstrate the versatility of adapting both probabilistic and evolutionary (biomimetic) paradigms in development of security services within a software system infrastructure. There is an extensive literature that covers various aspects of biomimetic computational models such as Genetic Algorithms (GA) and Artificial Immune System (AIS) and their applications 0, 0, 0, [8]. To validate the models and understand its limitations we have built a middleware security system framework with genetic and immune-computing paradigms. It is anticipated the applicability of the biomimetic paradigms becomes evident in the experiment 0.

3. A VIRUS ATTACK SCENARIO

A new unknown virus enters the software system and its undesirable presence or activities have been detected. The source(s) of the virus spread (carrier files) has to be located first. After performing some virus containment activities - a security expert has to distribute the code in order to detect the signature (pattern) which could help into provide a unique identification of the software virus. Consecutively, to heal (patch) infected networks, subsystems or files an effective antivirus can be developed. The signature of the virus is then be stored in a data-store (database), to allow specialised antiviral programs to detect and prevent known viruses from attacking the computer networks. Finally, data consistency and integrity on the system has to be verified, and further precautions, if required are taken. A similar procedure process has been repeated for all new variants of the virus (strains).

4. PROPOSED SOLUTION

Currently, there are several biomimetic metaphors being actively investigated by researchers in the engineering and informatics communities. Among the most popular bio-inspired models that are being adopted for a new generation of security systems are: protein pathway mapping, neural feedback loops and gene behaviour 0, 0. Our research focuses on constructing the architectural framework of the ISS that draws on maturity of de-facto security industry standard EASI (Estimation of Adversary Sequence Interruption) model 0 and taking an advantage of biomimetic methodologies. The architectural model of the proposed Biomimetic Security System (BISS) extends the EASI model by incorporating a set of biomimetic concepts and algorithms operating within a dedicated software infrastructure. The design of an Intrusion Detection services component in the BISS software infrastructure attempts to exploit the best aspects of probabilistic and evolutionary techniques. The architectural driver requirements for the proposed BISS have to ensure: autonomics (self-organisation and adaptation), interoperability, distributiveness and its lightweight footprint. By design, the various network components (nodes) are required to assess the situation, cooperate to defend and pool available resources and assist in decision making by adopting genetic and immune-computing principles.

5. EASI MODEL

Garcia's Estimation of Adversary Sequence Interruption (EASI) framework 0 is based on a probabilistic model that describes all elements of the system security system in terms of their safety and possibility of improvement. These aspects are expressed in probability estimates for item failure and possibility of security interruptions. The analytical work, including adjustment of various parameters, is still largely based on human intuition and experience. This is in order to obtain a meaningful interpretation of the results. The element of the security system, pertinent to EASI model includes physical properties and probabilities. Among physical properties are: adversaries (the opponent and his/her skill level), sensors and actuators (cameras, microwave, IR, fence sensors, switches and other devices), barriers (doors, walls, screens, etc.) located at points along the adversary path, communication devices, response force (the guards or other teams that protect the facility), the protected assets, path(s) taken by an adversary to reach the asset, points along the path and other elements. The calculation of probabilities involves the following parameters:

- $P(D)$, Probability of detection (associated with each sensor).
- $P(A)$, Probability of the alarm.
- $P(C)$, Probability that communications (in the facility) will be available.
- $P(I)$, Probability of interrupting the adversary along his or her path.
- $P(\frac{R}{A})$, given an alarm has sounded, the probability (conditional) that the response force will arrive.
- t_i, time that a barrier at a given point might be disrupted. The time value is dependent on the

adversary's skill level and tools possessed. This time can be thought of as the time the adversary takes to complete a task, such as breaking through a fence.

- RTF, the time a response force might take to respond to a given threat.
- TR, the time remaining for the adversary to reach the end point (the asset) once an alarm sounds.

The model uses the following equations:

$$P(A) = P(D)P(C) \text{ and } P(\frac{R}{A}) = \overset{\infty}{\underset{0}{f}} \frac{1}{\sqrt{2\pi\sigma_x^2}} e^{-[\frac{(x-\mu_x)^2}{2\sigma_x^2}]dx} \quad [1]$$

and where the mean is defined as:

$$\mu_x = E(TR) - E(RFT) \quad [2]$$

and variance:

$$\sigma_x^2 = Var(TR) + Var(RFT) \quad [3]$$

In practice, the tables are to store calculations of the mean and variance values. These tables show the times it may take an adversary with different set of tools to penetrate a barrier. The terms in the mean ux are:

$$E(TR)_p = E\left(t_{AD_p}\right) + \sum_{i=p+1}(t_i) \quad [4]$$

Where $E(TR)p$ is the expected time from any point p to the end point (the asset) n; the $tADp$ is the time after the detection (the intrusion was detected) at the point p; and the $E(t_i)$ value is the expected time to perform a task at a given time i along the intrusion path.

$$E\left(t_{AD_p}\right) = \begin{cases} E(t_i) & \text{if detection is at beginning} \\ \frac{E(t_i)}{2} & \text{if detection is in the middle} \\ 0 & \text{if detection is at the end} \end{cases} \quad [5]$$

The TR and RFT are independent and normally distributed. The normal distribution is approximated by letting both TR and RFT be sums of random variable which satisfy the conditions of the Central Limit Theorem. Thus, the final equation that calculates the probability of the interrupting the intruder given all definitions of probabilities above is:

$$P(I) = \\ P(D_1)P(C_1)P(R|A_1) + \\ \prod_{i=2}^{n}[P(D_i)P(C_i)P(R|A_i)\prod_{j=1}^{i-1}(1 - P(D)_j)] \quad [6]$$

The below table shows probabilities of detection $P(D)$ for selected sensors as the adversary can cut, crawls, walks, run, cuts the wire or break the sensor that we used in experiments. These probabilities values give an indication of the level of efficiency of the sensors in detecting an adversary. Currently, the data is empirical, depending on the research data-set used as a reference. The second table indicates probabilities of intrusion detection when adversary actually attempts to defeat various sensors when performing different modes of movements/activities as indicated in Table 1. On the other hand, Table 2 indicates probabilities of intrusion detection when adversary actually attempts to defeat the sensors. Table 3 below indicates action (penetration) times for an intruder travelling on foot, carrying explosives and metal cutting tools. The action time would vary depending on the intruder's skill level, weight carried, prior knowledge of the surround and possible inside support. Penetration time is also called a delay time, as the barriers *pr* action causes delays to the adversary.

Table 1: Probabilities of Intrusion Detection in Intrusion Modes

Sensor	**Slow Walk**	**Run**	**Crawl**	**Cutting**
Continuity	-	-	-	0.75
Light Beam	0.9	0.9	0.75	-
Mechanical Switch	0.9	0.9	-	-
PIR	0.75	0.75	0.5	-
Microwave	0.75	0.75	0.5	-
Press. S/W	0.75	0.75	0.75	0.75
Strain	0.75	0.75	0.75	0.25
Light level	0.25	0.25	0.25	0.25
CCTV	0.9	0.9	0.9	?

Table 2: Probabilities of Intrusion Detection for Various Locations of Sensors

Sensor	$\boldsymbol{P(D)}$
Combined badge reader	0.75 – 0.85
Officer at a check-point	0.5
Detectors on all walls	0.85 – 0.99
Exterior microwave detector	0.75 – 0.95

Table 3: Penetration Times for Various Actions of an Intruder.

Action	**Slow walk**	**Run**
Climb gate/fence	10s	±30% of expected *t*
Doors	12s	"
Badge reader station	8s (if all correct biometrics possessed)	"
Getting pass a checkpoint officer	30s	"
Door 6in metal	60s	"
30cm reinforced concrete walls/floors	3min	"
Door 3in metal	30s	"
1in interior wooden doors	60s	"

6. APPLICATION WITH SPRING TENSOR MODEL

The EASI method to determine the task probability of penetration and interruption activities can be alternatively applied by a globalized trajectory mapping method to analyse the magnitude and change of an intruder's movement.

The global trajectory method used in the experiment uses the Spring Tensor Model (STEM), a model that analyses protein fluctuation dynamics. As discussed in depth by Lin and Song 0, 0, the premise of the spring tensor model is to determine conformational changes in proteins using for calculating second-order partial derivatives (Hessian matrices) as indicated in Table 4. Conformational change is the transition of macro-molecular structures in proteins as a result in a change of acidity, temperature, voltages and so forth. The spring tensor model is an enhancement of anisotropic modelling and Gaussian modelling methods, as while the former determines fluctuations of an atom's direction, the latter is better at determining the prediction of magnitudes of direction 0. Thus by combining the two methodologies, the spring tensor model can be applied to the EASI method as follows:

6.1. Anisotropic Modeling

This modeling denotes the determination of conformational variation or fluctuation in direction between elements.

- ***Adaptation:*** This is suitable for determining how the interactions between an intruder and the asset will result in the degree of directional fluctuation. The variation of potential direction will indicate what possible directions an intruder will travel if they are in proximity with a security sensor, such as a motion detector.
- ***Meaning:*** Smaller anisotropic values indicate a smaller potential in which an intruder will move, while larger values indicate a larger potential for the intruder to move.

6.2. Gaussian Modelling:

This modeling is described n terms of determination of conformational variation or fluctuation in magnitude between elements.

- ***Adaptation:*** This is appropriate to ascertain how interactions between an intruder and a sensor will result in the magnitude or total range of the fluctuation. The variation of potential magnitude indicates the possible maximum range the intruder will travel if it is near the proximity of the sensor.
- ***Meaning:*** Smaller magnitudes values indicate a smaller potential of the intruder to alter their distance, while larger magnitudes indicate a larger potential for the intruder to alter their movements.

The STEM model's fourth term is of interest as it examines the global interactions of the elements. The final term, examined in Table 4, is shown with its Taylor expansion form [6]. The final non-local derivation is adopted from Lin and Song's calculations, which is used as a point of reference in this research project 0. Using the parameters stated by Clementi 0, the value of epsilon (ε) adopted is 0.36 as per conformation observations of macro-molecular protein structures using X-ray crystallography.

Table 4: Elaboration of the STEM Modelling Approach

$Y(X, Y_0) = \Sigma_{V1Bond} + \Sigma_{V2Angle} + \Sigma_{V3Dihederal} + \Sigma_{V4Non-Local}$	The $V(X, Y_0)$ values are sum of radial lengths, bonding angles and dihedral angles of consecutive objects i and j. Non-local contacts are used.
$V_4 = [5\left(\frac{r_0, ij}{r_{ij}}\right)^{12} - 6(\frac{r_0, ij}{r_{ij}})^{10}]$	The final non-local contact term is derived from the Go-like potential as discussed by Lin & Song's theoretical work.
$Y(X, Y_0) = \Sigma_{V1Bond} + \Sigma_{V2Angle} + \Sigma_{V3Dihederal} + \Sigma_{V4Non-Local}$	Taylor expansion of initial non-local contact term yields the equation, where r_{ij} and $r_{0,ij}$ are consecutive long-term values for objects i and j.
$\frac{\delta^2 V_4}{\delta X_i \delta Y_j} = \frac{240\varepsilon}{r_{0,ij}^2}(X_j - X_i)(Y_j - Y_i)^2$	As focus is on the equilibrium fluctuations, r_{ij} is equal or approximately equal to $r_{0,ij}$ at equilibrium; thus the derivative of V_4 can be simplified.

7. APPLYING THE METHODOLOGY

The proposed methodology is to be executing in several steps as follows:

STEP 1: In this step from site diagrams/maps we need to construct the possible adversary paths. The time to complete the task is called the Expected penetration time denoted as $E(t_i)$. Along the path, the adversary encounters different set of sensors which must be overcome in order to reach the asset. Associated with of the sensors is a Probability of detection $P(D)$ as indicated in the adversary path diagram for a selected site (Fig.1).

STEP 2: In this step all tasks are tabulated, and we use look-up tables to the allocated probabilities of detection $P(D)$ and expected barrier penetration times $E(ti)$. A standard deviation of ±30% is applied throughout the experiment. The values in Table 5 are determined using Tables 1, 2 and 3.

STEP 3: In this step we require to make the assumptions for the values of expected reaction time $E(RFT)$, the standard deviation $Var(RFT)$ and the probabilities for communication in the facility $P(C)$.

Research by Garcia 0 has indicated that appropriate values for $E(RFT) = 300s$, $Var(RFT) = 30s$ and the probabilities for communication in the facility $P(C) = 0.95$. This indicates the communication between security staff and all other parts of the facility are 0.95 probable. As follows, we need to calculate the values of Mean Variance parameters defined as:

$$\mu = E(TR) - E(RFT) \qquad [7]$$

$$\sigma^2 - Var(TR) + Var(RFT) \qquad [8]$$

and $P(\frac{R}{A})$, and finally the value of $P(I)$ using the previously defined formulas.

STEP 4: For each task indicated in the table defined in Step 2, locate tasks that have associate sensors in the neighbourhood. We need to decide if the adversary can be detected before the tasks begins, during the task or the completion of the task. If for example, we take task 6 as the adversary walks into the room, he or she can be detected by the PIR. In this case, the detection can occur before the task begins, such as crawling towards the next door. Hence, the following conditions can be considered (see Table 6):

- If detection occurs before a task begins, assign a $'B'$.
- If detection occurs in middle of task, assign an $'M'$.
- If detection occurs at end of task, assign an $'E'$.

STEP 5: In this step, the probability of interruption is calculated. The outputs of all the calculations used are presented in Table 8.

Thus the task probabilities for various penetration and interruption activities can be updated, with the optional application of the STEM model applied at this stage. Adjustments according to various detection conditions are established, due to deployment of various detectors and sensory systems as indicated in Table 7.

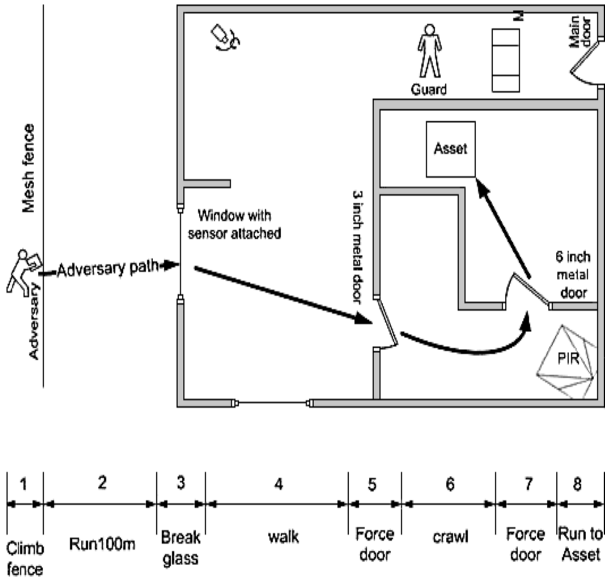

FIGURE 1: An adversary path for a given site.

Table 5: Allocated Probabilities and Expected Barrier Penetration Times for Listed Tasks.

Task	**Description**	**Any Sensors**	$\boldsymbol{P(D_i)}$	$\boldsymbol{E(T_i)}$ **in sec**	$\boldsymbol{Var(T_i)}$ **in sec**
1	Climb fence	Fence Sensor	0.9	10s	3s
2	Run (100m)	None	0	60s	18s
3	Break Glass	Continuity Sensor	0.75	10s	3s
4	Walk	CCTV (PTZ)	0.9	10s	3s
5	Force Door (3in metal)	None	0	30s	9s
6	Crawl (10m)	PIR	0.5	20s	6s
7	Force Door (6in metal)	Pressure, Biometrics	0.9	60s	18s
8	Run to asset	No	0	5s	1.5s

Table 6: Conditions for $\boldsymbol{E(TR)}$ and $\boldsymbol{Var(TR)}$.

The Conditions for $\boldsymbol{E(TR)}$	**The Conditions for** $\boldsymbol{Var(TR)}$
If 'B' then $E(TR) = E(t_p) + \sum_{i=p+1}^{n} E(t_i)$	If 'B' then $Var(TR) = Var(t_p) + \sum_{i=p+1}^{n} Var(t_i)$
If 'M' then $E(TR) = \frac{E(t_p)}{2} + \sum_{i=p+1}^{n} E(t_i)$	If 'M' then $Var(TR) = \frac{Var(t_p)}{4} + \sum_{i=p+1}^{n} Var(t_i)$
If 'E' then $E(TR) = \sum_{i=p+1}^{n} E(t_i)$	If 'E' then $Var(TR) = \sum_{i=p+1}^{n} Var(t_i)$

Table 7: Adjusted Values for Probabilities and Barrier Penetration Times for Listed Tasks

Task	**Description**	**Any Sensors**	**Det**	$\boldsymbol{P(D_i)}$	$\boldsymbol{E(T_i)}$ **in sec**	$\boldsymbol{Var(T_i}$ **in sec**
1	Climb fence	Fence Sensor	B	0.9	10s	3s
2	Run (100m)	None	E	0	60s	18s
3	Break Glass	Continuity Sensor	B	0.75	10s	3s
4	Walk	CCTV (PTZ)	B	0.9	10s	3s
5	Force Door (3in metal)	None	E	0	30s	9s
6	Crawl (10m)	PIR	B	0.5	20s	6s
7	Force Door (6in metal)	Pressure, Biometrics	B	0.9	60s	18s
8	Run to asset	No	B	0	5s	1.5s

Table 8: Calculation of the Probability of Interruption $P(I)$.

	Adversary	**Probability of Guard**		**Response Force Time (s)**	
	Sequence	**Communicate**		**Mean**	**Std Dev.**
	Interruption	**0.05**		**300**	**90**
				Delays (in secs)	
Task	**Description**	**P(Detection)**	*Loc*	**Mean**	**Std Dev.**
1	Climb fence	0.9	B	10	3
2	Run (100m)	0	B	60	18
3	Break Glass	0.75	B	10	3
4	Walk	0.9	B	10	3
5	Force Door (3in metal)	0	B	30	9
6	Crawl (10m)	0.5	B	20	6
7	Force Door (6in metal)	0.9	B	60	18
8	Run to asset	0	B	5	1.5
	Probability of Interruption	0.137438069			

8. CONCLUSION

In the proposed method, each of the tasks is independent. However it is envisaged that future models can include some task dependencies. The probabilities of communication and reaction time have fixed values. Evolutionary algorithms can be used to find the desired optimal values, as desired. Instead of the user manually entering the values for different paths into the computer program, or run a spread sheet program, an algorithm can be used to search the security space for the path with the greatest or smallest probability of interruption. Further, we should be able to find variations showing most vulnerable paths in the facility. This would be a path or paths that have a low probability of interruption. A user then can decide to change the placement of a sensor device, or add new once to improve the security. Both recent developments and our research in the project's field provide very encouraging results. However, more investigation is required before full confidence and wider acceptance of the approach is to take place in the ICT security industry.

ACKNOWLEDGMENTS

The authors would particularly like to thank Dr Perez Moses for his valuable ideas, observations and comments related to this work.

REFERENCES

Boukerche, A., Juca, K.R.L, Sobral J.B., Notare M.S.M.A.: An artificial immune based intrusion detection model for computer and telecommunication systems. Parallel Comp. 2004; 30(5–6), pp. 629–46.

Bruzzone, A.G., Longo, F., 2005. Modeling & Simulation applied to Security Systems. *Proceedings of Summer Computer Simulation Conference*, pp. 183-188. July 24-28, Philadelphia (Pennsylvania, USA).

Chaczko, Z. and Moses, P.: Neuro-Immune-Endocrine (NIE) Models for Emergency Services Interoperatibility, Lecture Notes in Computer Science, Computer Aided Systems Theory EUROCAST '07, Springer Berlin/Heidelberg, Vol. 4739, 2007.

Chaczko, Z.: Autopoietics of Biomimetic Middleware System, private correspondence, November, 2007.

Clementi, C., Nymeyer, H., Onuchic, J.N.: Topological and Energetic Factors: What Determines the Structural Details of the Transition State Ensemble and En-route Intermediates for Protein Folding? An Investigation for Small Globular Proteins, Journal of Molecular Biology, 298: pp. 937–953, 2000.

Garcia, M.L.: The Design and Evaluation of Physical Protection Systems, Elsevier, 2001.

Goel S. and Bush S.F.: Biological Models of Security for Virus Propagation in Computer Networks, LOGIN, Dec. 2004; 29(6): 49–56.

Lin, T.L. and Song, G.: Generalized Spring Tensor Models for Protein Fluctuation Dynamics and Conformation Changes, Computational Structural Bioinformatics Workshop, Washington D.C., USA, 2009.

Low, K.H., et. al.: Task Allocation via Self-organizing Swarm Coalitions in Distributed Sensor Networks, 19th Artificial Intelligence Conference, pp. 28–33, 2004.

Rice, J. and Martin, N.: Using biological Models to Improve Innovation Systems: The Case of Computer Anti-viral Software, European Journal of Innovation Management, Volume: 10, Issue: 2, pp. 201 – 214, 2007.

Shimonski**,** R.J.: What You Need to Know About Intrusion Detection Systems, 2004.