Association for Information Systems
AIS Electronic Library (AISeL)

PACIS 2006 Proceedings

Pacific Asia Conference on Information Systems (PACIS)

2006

Computer Immunodeficiency: Analogy between Computer Security and HIV

S Daskapan
Delft University of Technology, Semird@tbm.tudelft.n

Computer Immunodeficiency: Analogy between Computer Security and HIV

S. Daskapan
Delft University of Technology
Delft, The Netherlands
Semird@tbm.tudelft.n

Abstract

Current security systems are designed to prevent foreseeable attacks. Those security systems do not prevent effectively the more emergent types of attacks, like a botnet, whose presence and behavior is difficult to predict. In order to predominate those types of attacks, we advocate an adaptive security approach based on the animal immune system. But since those sophisticated attacks can also be directed at the security systems themselves, leading to computer immunodeficiency, like HIV, in this paper we propose a protocol that protects the immune system itself. This approach discriminates between attacks on the security systems, which are part of the computer immune system, and attacks on other vital computer systems in an information infrastructure.

Keywords: Self-organization, complex adaptive systems, information security, immune system.

1 Introduction

In large-scale information systems complexity is more and more challenging security policies. Not only are those systems large, but also distributed and interwoven. Besides that they are also rapidly changing and have thus a dynamic structure due to organizational and technological factors. As such changes in one infrastructure may have effect on the other infrastructures. It is even so that initially small flaws in one infrastructure could result in amplified problems in other dependent infrastructures (butterfly effect) and bounce back. The consequence is that complexity increases, manageability decreases and vulnerability of such interwoven infrastructures increases. Those large conglomerates of infrastructures are sometimes that complex that their total reaction to certain local distortions becomes even unpredictable (Amin 2000). This is a growing concern, since societies depend in business and in private occasions more than ever on those conglomerate information systems, i.e. information infrastructures. Information infrastructures that provide critical services to physical critical infrastructures, like roads and electricity, are referred to as critical information infrastructures (CII).

A CII could be an extranet of a bank, a WAN of a multinational, traffic control information systems or even (a part of) the Internet. The term critical infrastructure is here used to identify a chain of (sub)systems, the failure of which might cause high direct and/or indirect social, economical or ecological damage. A CII consists of many (sub)systems, or computing entities (CE's), with different functions. The role of a

security CE's in CII is crucial since all the security services that make the rest of the infrastructure dependable, arise in these nodes. We will call those security CE's that contribute to providing one or more of the security services, i.e. identification, authentication, confidentiality, integrity and non-repudiation, security distribution centers (SDC's).
Current SDC's, like virus scanners, intrusion detection systems, authentication servers and firewalls, are designed to repel foreseeable, i.e. "known" and "definable" attacks. Known means that its defense strategy is based on ex-post data and historically determined anomalies. Definable means that attacks are determinable in time (when?) and number (how many?). Current SDC's in such CII do not meet in time the required adjustments to unforeseen and emergent security attacks. Acknowledging this shortcoming, many organizations assure availability of their vulnerable ICT systems, and thus also their SDC's, by employing redundancy techniques (Barbour et al 1989; Hiltunen et al 2003; Reiter 1994; Veríssimo et al 2000;Gong 1993). Those measures are a posteriori and do not prevent attacks.

Because those traditional approaches are not designed to repel those complex attacks, in this paper we want to explore unconventional and risky alternatives instead of building upon existing and convenient security architectures: emergent complex problems require adaptive solutions. Since biology is an inexhaustible source of inspiration for many researches in many disciplines that apply or deal with emergent behavior, our security approach is, like many others (Kephart 1995; Forrest et al 1997; Goel et al 2003), also based on the animal immune system (AIS). This work differs however from those works on at least two points. First, those other works exploit the AIS to protect other functional CE's (cells) of the CII (body) than the SDC's. Sophisticated emergent attacks directed at SDC's of the AIS self, called HIV attacks, are not prevented. The in this paper proposed protocol does take them also into consideration. Second, those other works usually exploit the recognition technique of the AIS to develop better intrusion detection systems and as such can block more attacks effectively, but still not all attacks. Our protocol complements those AIS-based approaches, since it also takes care of the missed, and thus successfully penetrated, attacks. Conclusively, other works either do not focus on security systems (Amin 2000; George et al 2003; Kephart 1995; Forrest et al 1997; Goel et al 2003) or do not clear effectively the missed attacks (Kephart 1995; Forrest et al 1997; Goel et al 2003) or are not adaptive (Barbour et al 1989; Hiltunen et al 2003; Reiter 1994;Veríssimo et al 2000; Gong 1993).

2 Complex adaptive systems

Complexity in computer science can be defined as the level in difficulty in solving mathematically posed problems as measured by the time, number of steps or arithmetic operations, or memory space required. Understanding this complexity is not always possible as it goes beyond our mental capacities, so that solutions to control the conglomerate infrastructure seem rather unachievable. Consider the Internet: despite that the fact that it is the result of human effort, we are still not capable of understanding its behavior, not to mention to master it (Van Best 2005).

But complexity, as one of the culprits that increases vulnerability, has also a positive side effect. The many 'dumb' bees, for example, are able to architecture a most complex honeycomb with hexagonal cells. More complex life forms like the human body depict a conglomerate of many bio-compartments. Those biological systems in the human body contain countless cells with many differentiated tasks to maintain the numerous functions of the body, like self-healing of wounds, the AIS against pathological diseases and so on. Obviously, by letting a sample of specialized cells perform their specific and yet simple tasks they manage together an aggregated complex task. Although the result of this complex and intelligent behavior is not intended by each single cell and none of them might be aware of this meta- goal, yet it is the result of their collaboration. We consider therefore a *complex adaptive system* (CAS) as a collection of interdependent rule-following agents with interactions resulting in system-wide patterns across the group. The richness and volume of these interactions allow a complex system as a whole to undergo spontaneous self-organization. Self-organization is the emergence of a patterned outcome that no individual had planned, i.e. emergent behavior of the system. A characteristic is that no agent needs to be aware of the existence of the total space. Each CE knows at most what kind of capabilities it has and how it can look for relevant information in the environment. Properties of a CAS are (Wilinsky et al 1999): emergent behavior, adaptation, specialization, dynamic change, decentralization, competition and cooperation. Our positive perception of complexity, in which complexity is rather exploited to solve problems, is supported by other groups like the Santa Fe Institute (Langton et al 1992; Dooley et al 1995; Dooley 1997). As such in the next sections we will discuss a case where the AIS serves as input for information security problems.

3 An analogy from the animal immune system

Given the more sophisticated type of attacks within the increased complexity of infrastructures we foresee more opportunities in using complexity by means of a CAS to oppose security threats. We expect that especially in the field of computer security we can learn from the self-healing property of the human body to deal with many complex security problems in distributed systems. Particularly, the AIS will be considered to inspire us towards solutions as the main function of the AIS is to withstand foreign micro-organisms and to counter attack those who managed to penetrate the body.

The animal immune response system

Acknowledging that the body of vertebrates is immensely complex and the fact that it is capable of controlling this complexity autonomously is something we can learn from to master our problems. Given the motivation of one cell to survive (Dawkins 1989) and the aimed behavior of an SDC, also to survive, we see reasons to learn from the AIS.

The body of vertebrates consists of a conglomerate of many bio-compartments. Those biological systems in the body contain countless cells with many differentiated tasks to maintain the numerous functions of the body, like self-healing of wounds, the AIS against pathological diseases and so on. By letting a sample of specialized cells to perform their specific and yet simple tasks they manage together an aggregated complex task. Although the outcome of this complex and intelligent behavior is not aimed by each

single cell and none of them might be aware of this meta- goal, yet it is the aggregated result of their collaboration. The analogy would be that clustered computers clear failures like the AIS is doing with antigens by seamless collaboration. In this section we aim at finding ideas and requirements for achieving the research goal to build a security defense system.

The AIS is a complex network of closely cooperating cells and molecules, performing their functions with other organ systems. Its primary task is the induction and regulation of immunity to pathogens, such as bacteria, viruses and other micro-organisms, and to tumors (Roitt et al 2001). Host defences include both physiologic barriers and immunological responses. Skin and mucous membranes provide the first line of defence. Immune defences consist of innate and adaptive response system. The mechanism for the *innate immunity* is always present and comes rapidly into action. It provides a coarse-grained defence system that usually does not hold for a long time. Because of its general recognition system many pathogens are able to pass by this weak access control. The *adaptive immunity* system in contrast is capable of recognizing many specific pathogens and forms as such a fine-grained access control. Besides that, it is capable of building new antibody cells against new pathogens, although this process costs time. While both responses are not mutually exclusive, they provide distinctly different advantages for dealing with pathogenic organisms. As a description of the specific working of the overall AIS would surpass the goal of this section we will limit us to one specific type of collaboration between some specific cells (among others from (Perham 2000; Roitt et al 2001).

> *Many types of white blood cells, like the small T- and B-cells (lymphocytes), originated in the bone marrow, play during their maturation in the body an important role in dismantling the pathogens. After recognition of the antigens the B-cells are cloned and dispatched through the lymphatic system to inform and to mobilize other lymphocytes. As such the lymphatic system facilitates the communication service by recirculating the lymphocytes and antibodies through the body. In the meanwhile other cloned B-cells bind to the antigens to mark them and a few of them do nothing as they function as memory cells for immediate recognition in future invasions. The killer (or cytotoxic) T-cells trace those marked pathogens, perforate infected cells and kill the body's own cells that have been evaded by activating programmed cell death. Ultimately, the phagocytic cells, like macrophage and neutrophil, trace the remaining marked micro-organisms. After opsonization (coating) or neutralization they are absorbed and destroyed.*

The seamless and dependent collaboration between the different cells appears in many phases, like for example between the T- and B-cells. Through a process of cloning, specific B-cells are stimulated to proliferate and differentiate to bind the antigens. Although B cells are able to recognize antigen, they are unable to proliferate and differentiate unless triggered by the action of T-cells.

> *In order for the T-cells to become stimulated to release lymphokines, they must also recognize specific antigens. However, while T-cells recognize antigen via their T-cell receptors (T-helper cells $CD4^+$), they can only do so in the context of the antigen-presenting cells (APCs). Several types of cells may serve the APC function, like the*

macrophage and dendritic cells, but also the B-cell self. When APC bind antigen, the antigen becomes internalized, processed and expressed on the surface of the APC. This specific expression is now recognized by T-helper cells so that they can release a trigger for the B-cells.

There two main types of T-cells: the helper T-cell and the killer cytotoxic T-cell. There two main types of T-helper cells: those that help the B-cells (Th2) and those that help the cytotoxic T-cells (Th1). The effect of HIV on the immune system is the result of gradually eliminating the Th1 and Th2 helper T-cells. The immune system gradually looses its functionality until it is malicious and/or malfunctioning.

The application

The body consists of many organ systems to keep the body healthy. Think of the straining function of toxic substances by the liver or the skin and mucous membranes that provide the first line of defense against pathogens. Above this, the immune response system provides the innate and adaptive response system. The T- and B- cells are analogously the SDC's in CII. Similar to the elimination of the Th1 and Th2 helper T-cell sub-populations by the HIV in a body, in CII the SCD's can be eliminated by malware with emergent behavior (HIV attacks).

Focusing on the last one, the immune response system is apparently also specific, adaptive and has a memory for protecting and healing the body. From the previous we can derive some ideas about realizing the aimed defense system as a CAS. The immune response system depicts most of all a sophisticated way of *self-organization by collaborating individual cells*. Requirements for achieving this self-organization principle can be derived from the working of the cells in the AIS. The cells

- have specific internal motivators to act, that can be triggered by external events,
- are selfish (Dawkins 1989),
- are not necessarily aware of the consequence of their behavior,
- have limited knowledge of other cells,
- perceive other cells either as correct or compromised,
- are able to find and communicate with helper cells,
- are able to detect and isolate intruders and defect cells,
- can be increased in number by cloning,
- are able to distinguish between common and rare invasions,
- can rely on a collective memory about intruders.

Considering this adaptive, specialized, decentralized and cooperative behavior of cells, we claim that: *attributing SDC's with the similar characteristics as the animal immune cells will enable them together to build innate and adaptive immunity such that HIV attacks can resisted.*

4 Overview of the approach

In this section the so-called 'escapability' behavior of an endangered SDC will be introduced. We will draw an impression of how a distributed defense system, as a result of the collaboration of the individual nodes, should deal with infected SDC's. In this approach the adaptive system takes also care of HIV attacks.

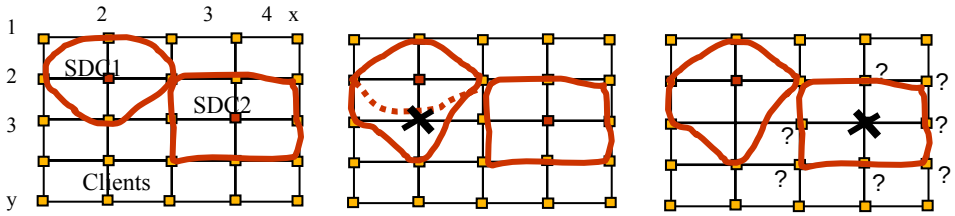
The body

Assume a space in which dozens of entities want to collaborate with each other (universities, naval, etc), but are unknown and thus by default initially do not trust each other. That means that they are logically tied to each other in a kind of grid, but due to the prisoners dilemma they do not act. A lot of effort is put therefore to establish a trust centre and to point out one of the members as a trusted point of reference to mediate security services (authentication/ key server), i.e. SDC. The essential requirement is that all the other members agree on and authorize this particular member. Once this is done all the members can rely on the trust centre for confirming identities and distributing keys. When this trust centre collapses or starts malfunctioning this would be disastrous for the whole group. The establishment of a new trust centre would be inevitable when the trust centre cannot be recovered properly or/and on time. Establishing a trust centre between unknown entities is one of the most difficult and costly things in the security domain, as it requires not only technical means but also social and sometimes political means (agreements, treaties). Therefore, a once established trust service should ideally be carried on without any process disjunction, regardless the vitality of the hosting trust base.

The innate response system

Assume now that there are two SDC's in that space that issue trust (i.e. certificates or keys), like Kerberos does (Neuman 1994). An SDC knows only one operation mode: common session security mode. In the common session mode, an SDC reacts as a leader of clients immediately on known requests from clients and known security attacks from outside by common security measures (firewalls, IDS, etc). Each SDC takes care of the distribution of keys within its own group. If an object wants to communicate with another object, the SDC mediates trust by distributing keys. In figure 1 two groups are depicted, lead by SDC1 and SDC2. In fig a object (x,y) = (2,3) can start therefore communication with object (3,2) or with an object from another group like (3,3). In the latter one it is required that both SDC's have a trust relationship. In figure b a situation is depicted in which trust center (2,2) has formed a new group after a leaving member (2,3) and new joining members ((1,3);(2,4)). However, when a trust center like (4,3) collapses due to unknown and unexpected failures then all the group members become useless orphans (at least for a certain crucial moment) as they are not trusted and secured. Only CE (x,y) = (3,2) remains secure and trusted due to his subscription to a second SDC, i.e. SDC1. Any requests for interaction will be rejected, since there is no trust center to verify their identity and to check permissions. This is depicted in figure c.

The three figures depict how a conventional SDC's react on a) common client requests b) client group behaviour and c) SDC failures. In the latter, most clients become orphans.



a. SDC's mediate *b. Clients leave/join* *c.SDC2 collapses*

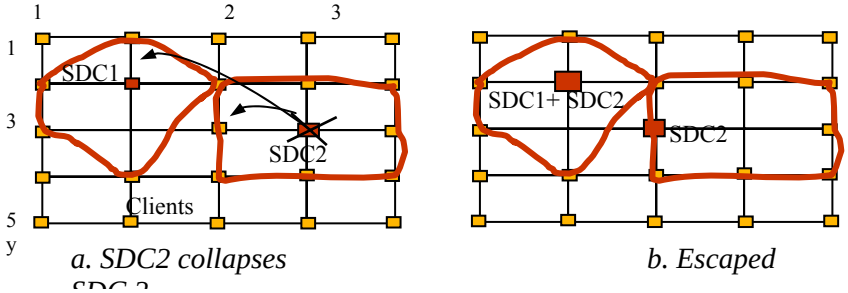
Figure 1. Conventional SDC's

The adaptive response system

Assume now that an SDC distinguishes two operation modes: *common session security* mode and the *survivability* mode. In the survivability mode, the SDC serves clients but participates also in a resource sharing pool. As such it reacts to unknown failures (internally or from outside) also immediately, but now by using superfluous capacity of one of the other CE's in this trust pool. Assume now that SDC1 has formed a pool with SDC2 and CE(3,3); see Figure 2. According to the metempsychosis principle the trust authority must not bind permanently to any hardware architecture so that the SDC can continue his tasks on another pool member. This mechanism should subsequently take care of consolidating the state of trust, packaging and launching the trust package. This trust package is the trust token that should contain the essential unique secret characteristics. In figure 2b the trust authority at (4,3) escapes to one of its neighbors (3,3) and recovers (reincarnates) there, like is explained for object volatility, so that the group of clients remains undisturbed after all. By doing so, this mechanism takes care of the reliability and availability of the trust service. The interesting thing is that with this approach trust remains during the security session centralized, but on attack it distributes and benefits from a decentralized approach as it can practically hop to any collaborating peer.

It is obvious that the larger the network and thus the more collaborating CE's there are, the metempsychosis principle and the principle of self-organization can provide perpetual availability of security services by continuously hopping away and reincarnation. Besides that in this mode the SDC escapes it builds internally also resistance by learning from the failure or type of attack. As for example, in a simple form a virus detection system is updated with the new type of virus. The corrupted SDC is isolated by the other SDC's. They avoid any relation further with the corrupted SDC and multicast to the clients his state of corruption.

The figures show how SDC's, should react on failures, so that all clients remain served. Collapsing SDC2 can continue his service by moving to one of his clients or to another SDC.



a. SDC2 collapses *b. Escaped*

Figure 2. Escaping SDC's

5 Definition of security as a CAS

In order to characterize a SDC as a CAS we need to define the assumptions and the instruction set of each agent. In this case we choose a key distribution centre (KDC) as the SDC, although others are instances possible.

Assumptions:

- A CE is an agent.
- An agent is either a client agent (Ac) or a KDC agent (Akdc).
- A Akdc can deliver key distribution services to Ac's.
- An agent sees a limited set of other agents in the total space.
- An Akdc shares with two ore more other Akdc's a trust relationship.
- An agent is not aware of the consequences of his actions for the overall system.
- An agent perceives only availability and trustworthiness of other agents.
- An agent has a memory system, i.e. it can store basic information about other agents.
- An agent is able to send and receive messages.
- An agent is either triggered by an explicit message or by a faulty expected message from other agents.
- An external certification authority (CA) is ad hoc present (possibly also as a result of previous case).
- The Akdc have signed certificates to verify their asymmetric key pair and their trustworthiness[4].
- Each Akdc is frequently suffering from denial of service attacks, but not all at the same time.

Instruction set

Three instruction sets keep the organism healthy: the innate, the adaptive and the common response system. The three instruction sets are called Medusa. The innate refers in our information infrastructures to the front end security systems, like intrusion detection and firewalls. The common security system refers to the service of the regarded SDC, i.e. a KDC. Both, front end security systems and KDC, will not be discussed further, since they are exiting technologies. The instruction set for the adaptive response systems, however, is not existing and will be given in the following instruction set. The next autonomous actions for each agent separately in the CAS enable the SDC service to be resilient by continuously hopping away from the attacked Akdc.

0. Most Ac's: subscribe by sending a secret *s* to one of the Akdc's (using his public key) based on his trustworthiness.
2. Akdc: creates and sends on request symmetric keys to Ac's.
3. Akdc: frequently creates tokens *t* of those secrets *s* [5].
4. Akdc: frequently sends *t* and a list of his preferred successors (SL) to those successor Akdc's (SAkdc's).
5. SAkdc's: check availability of suffering Akdc.

While an Akdc is DOS attacked do:

6. Successor Akdc's: send declaration of death of harmed Akdc to each other.

If majority agrees on death:

[4] In the so-called extension block of the certificate (X.509 vs.3 or SPKI format).

[5] Token = share, according to Shamir's secret sharing algorithm

7. SAkdc's: send their *t* to first ranked SAkdc on SL.
8. SAkdc: reconstructs the secrets of the clients.
9. SAkdc: refreshes and sends the new secrets to the clients.
10. Go to 2 with Akdc = SAkdc.

This instruction set takes care of continuously replicating the security service and letting it resurrect on another host Akdc after a DOS attack. The new host Akdc functions as a temporary carriage, i.e. execution platform, until he is also attacked. This mechanism lets the security service to be independent of the resources of a particular Akdc. The clients are not necessarily aware of this host transition, since the trust relationship is based on the shared secret s and not on the identity of the Akdc. A detailed description of an applied instruction set can be found in (Daskapan 2006).

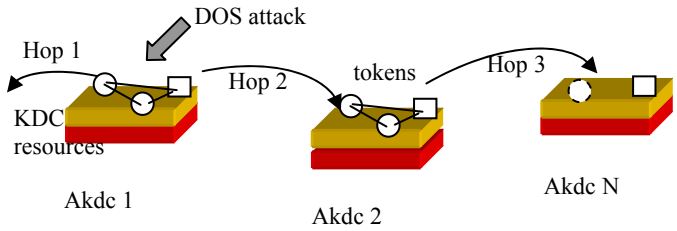


Figure 3. Continuous hopping away security services

6 Test

Preliminary tests have been conducted using also here a discrete event simulator NS+ with C++/TCL on Linux (Fall et al 2000). The aim is to see if resilience of front-end SDC's by applying CAS approach of participating SDC's is improved.

The test

The tests were performed up to 100 nodes of a CII in a WAN topology. In our test model we assume an existing botnet that tries to jeopardize our CII. The SDC's in the CII were exposed to multiple distributed denial of service attacks (DDOS) from the botnet to cause system failures. The particular type of DDOS attack used for this scenario was the buffer overflow DDOS (Chang 2002). In the simulation model we assume that the SDC is a key distribution centre where the leader has a buffer for session key requests. This buffer will be exploited in this scenario. Normally, the leader will process requests faster than it receives but for this scenario, clients are instructed to send requests at a much faster rate so that the leader inevitably will collapse. To test survivability multiple consecutive DDOS attacks were planned.

The result

The test was first run with 6 participating SDC's. Each SDC had a relation with two other SDC's, such that all the SDC's were directly or indirectly connected. DDOS attacks were executed then at any first one and subsequently at his successor and so on. Medusa was in this case able to resurrect the security service three times. The reason for that is that a majority of SDC's must remain honest to clear the embedded voting algorithms reliably. We have conducted this test with larger numbers. It appears that when we increase the number of participating SDC's also the number of attacks it can resist increases.

Many other tests were also carried out with different types of networks and different levels of trustworthiness of the agents. Also the simulation model was validated and security of the protocol was verified. Those detailed tests are out of the scope of this paper, but can be found in (Wiechers et al 2004; Wiechers et al 2005).

7 Conclusions and future work

In this paper we have derived ideas from the animal immune system to build an adaptive security defense system. This defense system discriminates between attacks on the security systems, which are part of the computer immune system, and attacks on other vital computer systems in an information infrastructure. The preliminary tests depicted that the claim of the defence system to be adaptive and to resist multiple attacks, even on the security systems self, can be met. In our future work we intend to apply more concepts from biological systems and to improve this initial protocol.

References

Amin, M. "National Infrastructures as Complex Interactive Networks," in Automation, Control, and Complexity: New Developments and Directions, T. Samad and J.R. Weyrauch (ed), John Wiley & Sons, NY, 2000.

Amin, S.M. "Toward Self-Healing Infrastructure Systems," IEEE Computer Magazine (33:8), 2000, pp. 44-53.

Barbour, A. E., and Wojcik, A. S. "A General Constructive Approach to Fault-Tolerant Design Using Redundancy," IEEE Trans. on Computers (38:1), 1989, 15-29.

Chang, R. K. C. "Defending against Flooding-Based Distributed Denial-of-Service Attacks: A Tutorial," IEEE communications magazine (40:2), 2002, pp.42-51.

Daskapan, S. "Building governments in e-government: settlement of dependable trust roots," Int. Conference on Availability, Reliability and Security (in DeSeGov), Austria, 2006.

Dawkins, R. "The Selfish Gene". Oxford, Oxford University Press, 1989.

Dooley, K., Johnson, T., et al. "TQM , Chaos and Complexity." Human Systems Management (14:4), 1995, pp. 1-16.

Dooley, K. "A Complex Adaptive Systems Model of Organization Change." Nonlinear Dynamics, Psychology and Life Science (1:1), 1997, pp. 69-97.

Fall, K., and Varadhan, K., "NS Notes and Documentation", T. V. Project, UC Berkeley, LBL, USC/ISI and Xerox PARC, 2000.

Forrest, S., Hofmeyr, S., and Somayaji A. "Computer immunology," Communications of the ACM (40:10), pp. 88-96, 1997.

George, S., Evans, D., et al. "Biological Programming Model for Self-Healing," ACM Workshop on Survivable and Self-Regenerative Systems, 2003.

Goel, S., and Bush, S. F. "Kolmogorov complexity estimates for detection of viruses in biologically inspired security systems: A comparison with traditional approaches," Complexity (9:2), 2003, pp.54–73.

Gong , L. "Increasing Availability and Security of an Authentication Service," IEEE Journal on Selected Areas in Communications (11:5), 1993, pp. 657-662.

Hiltunen, M. A., Schlichting, R.D., et al. "Building Survivable Services Using Redundancy and Adaptation." IEEE Transactions on Computers (52:2), 2003, pp 181-194.

Kephart, J.O. "Biologically inspired defenses against computer viruses," Proceedings of IJCA, Montreal, 1995, pp. 985-996.

Langton, C. G., Taylor, C., et al. (Ed.), "Santa Fe Institute Studies in the Sciences of Complexity". Artificial life 2, Addison-Wesely, 1992.

Neuman, B. C. and Ts'o T. Y. "Kerberos: An Authentication Service for Computer Networks," IEEE Communications (32:9), 1994, pp 33-38.
Parham, P. "The Immune System," Garland Publishing, 2000.
Reiter, M. "Secure Agreement Protocols: Reliable and Atomic Group Multicast in Rampart," 2nd ACM Conf. on Computer and Communications Security, 1994.
Roitt, I. M. and Delves P. J. "Essential Immunology," Blackwell Science Inc., 2001
Van Best, J.P. "Unraveling internet infrastructure," Eburon, Delft, 2005.
Veríssimo , N. F. N., and Correia, M. "The middleware architecture of MAFTIA: A Blueprint," IEEE Third Information Survivability Workshop, Boston, 2000.
Wiechers, W., Daskapan, S. "Validating the Security of Medusa: A survivability Protocol for Security Systems," Workshop on Security In Information Systems, Miami, 2005.
Wiechers, W., Daskapan, S., Vree, W. G., "Simulating the Establishment of Trust Infrastructures in Multi-Agent Systems", 6th International Conference on Electronic Commerce, Delft, 2004.
Wilensky, U., and Resnick, M. "Thinking in Levels: A Dynamic Systems Perspective to Making Sense of the World," Journal of Science Education and technology, 1999.