DOI: http://dx.doi.org/10.15446/ing.investig.v35n1.44323

Basic definitions for discrete modeling of computer worms epidemics

Definiciones básicas para el modelado discreto de epidemias por gusanos informáticos

P. Guevara[1], G. Delgado[2], J. Audelo[3], J. S. Valdez[4] and H. M. Pérez[5]

ABSTRACT

The information technologies have evolved in such a way that communication between computers or hosts has become common, so much that the worldwide organization (governments and corporations) depends on it; what could happen if these computers stop working for a long time is catastrophic. Unfortunately, networks are attacked by malware such as viruses and worms that could collapse the system. This has served as motivation for the formal study of computer worms and epidemics to develop strategies for prevention and protection; this is why in this paper, before analyzing epidemiological models, a set of formal definitions based on set theory and functions is proposed for describing 21 concepts used in the study of worms. These definitions provide a basis for future qualitative research on the behavior of computer worms, and quantitative for the study of their epidemiological models.

Keywords: Epidemic, computer worm, basic definitions, hosts, network, state.

RESUMEN

Las tecnologías de la información han evolucionado de tal manera que la comunicación entre computadoras o *hosts* se ha vuelto algo común, hasta el punto que la organización a nivel mundial (gobiernos y grandes empresas) depende de esto; lo que pasaría si estas computadoras dejaran de funcionar por un periodo de tiempo largo es catastrófico. Desgraciadamente, las redes de *hosts* son blanco de ataques por *malware* como virus y gusanos informáticos que podrían colapsar el sistema. Esto ha servido como motivación para el estudio formal de los gusanos informáticos y sus epidemias con el fin de idear estrategias de prevención y protección; por ello en el presente artículo, y previo al análisis de modelos epidemiológicos, se proponen un conjunto de definiciones formales basadas en la teoría de conjuntos y funciones que permiten describir 21 conceptos utilizados en el estudio de los gusanos informáticos. Estas definiciones servirán de base para futuras investigaciones cualitativas sobre el comportamiento de los gusanos informáticos, y cuantitativas para el estudio de sus modelos epidemiológicos.

Palabras clave: Epidemia, gusano informático, definiciones básicas, red de computadoras, estado.

Received: July 8th 2014
Accepted: November 4th 2014

Introduction

According to (Audelo *et al*., 2013), the origin of computer worms can be traced back to 1979, when scientists in XEROX PARC laboratories found that their equipment did not turn on, restart or collapsed the system. John Shich and Dave Boggs wanted to observe the traffic behavior patterns on networks under certain workloads. They realized that results would not be real enough due to the lack of data traffic, so they developed a program to simulate a high workload, which resulted in the first computer worm prototype. The first recorded computer worm was created by Robert Tappan Morris, a student at Cornell University, in 1988. Apparently, the intention of his program was not malicious, but it dispersed with such speed that approximately 60,000 UNIX systems collapsed (15% of the Internet) (Erbschloe, 2005).

According to (Audelo *et al*., 2013; Erbschloe, 2005), a computer worm is a malicious code (malware) with the ability to spread itself

[1] Pedro Guevara López. Ph. D. in Computer Science, Instituto Politécnico Nacional, México D. F. Affiliation: Titular Professor, Real-Time Mechanical and Electrical Engineering school, Instituto Politécnico Nacional, México D. F.
E-mail: pguevara@ipn.mx

[2] Gustavo Delgado Reyes. Doctoral Student in Communications and Electronics, M. Sc. Microelectronics, Instituto Politécnico Nacional, México D. F. Affiliation: Mechanical and Electrical Engineering school, Instituto Politécnico Nacional, México D. F.
E-mail: dengue_mgs4@hotmail.com

[3] Jesús Audelo González. Ph. D. in Communications and Electronics, M. Sc. in Microelectronics, Instituto Politécnico Nacional, México D. F. Affiliation: Mechanical and Electrical Engineering school, Instituto Politécnico Nacional, México D. F.
E-mail: jaudelo@ipn.mx

[4] Jorge Salvador Valdez Martínez. Doctoral Student in Communications and Electronics, Instituto Politécnico Nacional, México D. F. Affiliation: Professor, Industrial Mechanic Academic Department, Emiliano Zapata Technological University, México.
E-mail: jorgevaldez@utez.edu.mx

[5] Hector Manuel Pérez Meana. Ph. D. in Electrical Engineering from Tokyo Institute of Technology, Tokyo, Japan. Affiliation: Titular Professor, Microelectronics, Mechanical and Electrical Engineering school, Instituto Politécnico Nacional, México D. F.
E-mail: hmperzm@ipn.mx

How to cite: Guevara, P., Delgado, G., Audelo, J., Valdez, J. S., & Pérez, H. M. (2015). Basic definitions for discrete modeling of computer worms epidemics. *Ingeniería e Investigación*, *35*(1), 79-85. DOI: http://dx.doi.org/10.15446/ing.investig.v35n1.44323

when using operating system processes that are generally invisible to the user. A worm does not change file systems, but rather resides in memory and replicates itself, often causing problems to the network (e.g., using the bandwidth available or system resources until tasks are slow or not executable). Another important feature is its capability to spread in a network; worms are able to send copies of themselves between network terminals without the user intervention.

Since the first computer worms appeared, they have caused extensive damage to government institutions, universities and companies, generating numerous economic losses (Audelo *et al.*, 2013). Recent technological advancement has allowed the development of worms that are increasingly more difficult to counter-attack, hence the importance of modeling the dynamics of spread using epidemiological models to generate new techniques and tools to counter-attack in a fast and effective way (Nazario, 2004). Figure 1 shows a flowchart with computer worm action, from a host in a network to epidemic.

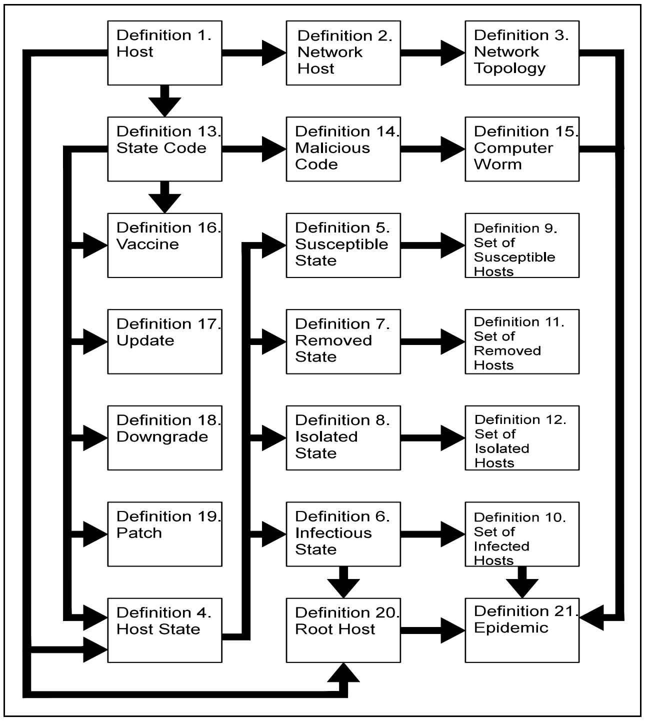
Definition 1. Host
Definition 2. Network Host
Definition 3. Network Topology
Definition 13. State Code
Definition 14. Malicious Code
Definition 15. Computer Worm
Definition 16. Vaccine
Definition 5. Susceptible State
Definition 9. Set of Susceptible Hosts
Definition 17. Update
Definition 7. Removed State
Definition 11. Set of Removed Hosts
Definition 18. Downgrade
Definition 8. Isolated State
Definition 12. Set of Isolated Hosts
Definition 19. Patch
Definition 6. Infectious State
Definition 10. Set of Infected Hosts
Definition 4. Host State
Definition 20. Root Host
Definition 21. Epidemic

Figure 1. The 21 definitions developed in this paper are linked in order to explain how the computer worm spread evolves through a network. In this sense, definition 1 is where the infection starts, and definition 21 is where the epidemic occurs.

Most known computer worms are spread in one of the following ways: via files sent as email attachments, via a link to a web or FTP resource; via a link sent in an ICQ or IRC message; via P2P (peer-to-peer) file sharing networks; and some worms are spread as network packets. Computer worms can exploit network configuration errors (for example, copy themselves onto a fully accessible disk) or exploit loopholes in operating system and application security. Many worms will use more than one method in order to spread copies via networks (Kaspersky labs, 2014). In this sense, future worm epidemics might spread at unprecedented rates in high-speed networks (Chen & Robert, 2004). A comprehensive automated defense system will be the only way to contain new threats but could be too risky to implement without more reliable detection accuracy and better real-time traffic analysis.

In this paper we propose a set of formal definitions based on set theory, and functions are proposed for describing 21 concepts used in the study of computer worms. These definitions provide a basis for future qualitative research on the behavior of computer worms and quantitative for the study of their epidemiological models.

Commonly used concepts in modeling computer worms epidemics

Computer worms have the capability to spread themselves without any intervention from the user. This feature allows an analogy with biological diseases. Mathematical models based on Kermack-Mckendrick (Kermack-Mckendrick, 1927) can be used to describe the dynamic behavior of the disease spread. These models can also describe the spread behavior of computer worms. Usually, they are referred as dynamic systems represented by differential equations; for example, in the cases shown in (Changchun *et al.*, 2002; Yang-Chenxi, 2003; Tao *et al.*, 2007; Onwubiko *et al.*, 2005; Juan *et al.*, 2010; Hincapié-Ospina, 2007; Tassier, 2005) they are used to represent *SI*, *SIR* and *SIRS* models (Hincapié-Ospina, 2007). However, before considering the dynamic modeling, it is necessary to explain commonly used concepts in specialized literature.

Host: A computer whose programs access another computer through a network (Downing *et al.*, 2009).

Network: A system of computers, and often peripherals like printers, that are interconnected (Downing *et al.*, 2009). See Figure 2.

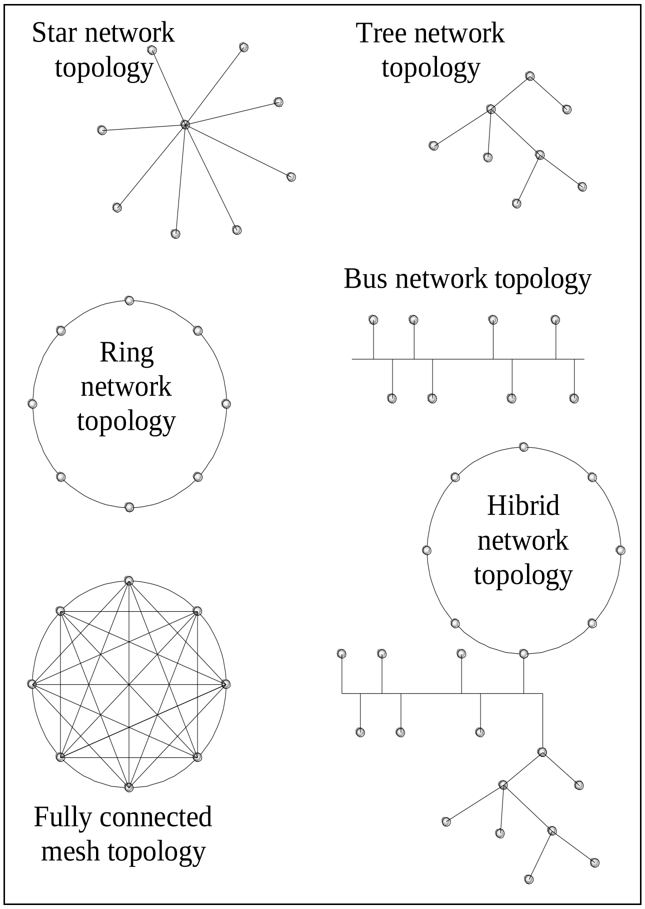

Figure 2. Network and Network Topologies.

Network Topology: The communication link used by the nodes in a network for communication (Downing et *al.*, 2009). Network topologies, shown in Figure 2, are classified according to the network architecture, or how they interconnect the different nodes or users on the network.

Malicious Code (Malware): Malicious software with the main goal to infiltrate, damage or make use of a computer resources without the owner's consent, according to (Sabin, 2011). In the information field, technology malware is also often referred to as hostile, intrusive or annoying.

Computer Worm: According to (Cohen, 1992), a computer worm is a program, whose main feature is the capability to self propagate through a data network without the need of explicit participation of any user. Once it is released by its creator, the code is designed to propagate autonomously, allowing it to exploit failures in the network administration politics and vulnerabilities of their services, and depending on the purpose for which the worm was designed, using the operating system processes automatic and invisible for the user, so they are not easily detected. However the damage caused by worms is perceptible, causing great instability of the systems.

Epidemiology: The study of the general laws of infectious diseases distribution (including characteristics of the infection source, the transmission mechanism, the susceptibility to infection, etc.) and the general principles of prevention and control of disease (Changchun et *al.*, 2002; Kephart et *al.*, 1993).

Epidemiological State: Classical epidemiological models consider three states (Kermack-Mckendrick, 1927; Hincapié-Ospina, 2007): susceptible state S, infectious state I and removed or recovered state R:

Susceptible State (S): For individuals who have no immunity against the infectious agent.

Infectious State (I): For individuals already infected who can transmit the infection to susceptible individuals.

Removed or Recovered State (R): For individuals who are already immune to infection, and therefore do not affect the dynamics of the transmission in any way, even when they enter in contact with other individuals.

Updates: Programs whose main objective is to repair flaws for the vulnerabilities in the first versions of operating systems. In some applications, there are also updates with new features for the operating system (Microsoft, 2013).

Patches: Programs that contain applicable changes for a program, usually security fixes for bugs. They may increase the functions in applications or change the defined language of a program (Downing et *al.*, 2009; Sabin, 2011).

Antivirus Software: To prevent some programs or applications classified as harmful for the system, there are programs that must be installed in operating systems. They run in the background to detect programs, and notify the user when they block, delete or contain malware (Downing et *al.*, 2009; Sabin, 2011).

The next section presents a set of formal definitions based on discrete mathematics involving concepts related to computer worms.

These definitions will be used to analyze classical models as *SI*, *SIR* and *SIRS,* which in fact are mathematical models to describe the dynamic behavior of epidemics and propose new models of epidemics by worms, considering that the cardinality of hosts sets is integer and not fractional.

Basic definitions for computer worms epidemic modeling

This section will define the set of basic concepts employed in the epidemiological models based in approximations. It also presents the worm environment and the definitions of each element involved in the epidemiological process: host, network, worm, vaccine, update, downgrade, set of susceptible hosts, set of infected hosts and set of removed hosts.

Definition 1 (Host). Every host $h_{i,k}$ is a triad compound for a computer o_i, an state $E_{i,k}$and a set of computers neighboring next to V_i to o_i; this is:

$$h_{i,k} = (o_i, E_{i,k}, V_i) \tag{1}$$

where $i, k \in \mathbf{N}^+$ are the index that identifies the host and the index of evolution, respectively. Thus a host is a computer whose programs access other computers through a network and can have several states and several computer neighbors. An example of a host can be a computer connected to a network with an operating system like Windows® or Mac®, vulnerable to attacks. In the case of Figure 2, it can be any of the nodes shown.

Definition 2 (Network). Every network N is a set of n hosts:

$$H = \quad \{h_{1,k},...,h_{i,k},...,h_{n,k} | i,k,n \in N^+\} \tag{2}$$

And a set of *m* connections between hosts:

$$C = \{c_1,...,c_i,...,c_m | l,m \in N^+\} \tag{3}$$

Such as it may be represented by a graph:

$$N = (H, C) \tag{4}$$

In this sense, every network host is a set of hosts where each one is connected by a physical medium and a protocol with at least another host. Hosts can be connected by WIFI, Ethernet, coaxial or even Bluetooth. An example of a network can be a set of interconnected computers on a corporate office or cyber site.

Definition 3 (Network Topology). Every topology T of a network N is defined by the arrangement between a set of hosts H and a set of connections C; it describes the shape of way between hosts $h_{i,k}$and $h_{j,k}$ with $i \neq j$ and $i, j, k \in \mathbf{N}^+$

Every Network Topology is an orderly array of a host set, independently of the medium or protocol used. For example, a star topology is present on a cyber site with several hosts connected to a single access point. See figure 3.

Definition 4 (Host State). The state $E_{i,k}$ of a host $h_{i,k}$ is defined as the situation that has a host and it is described by:

$$E_{i,k} = X \tag{5}$$

with:

$$X \in \{X_1,...,X_g,...,X_z | g,z \in N^+\} \tag{6}$$

Where the classic epidemiological models of a worm *X* is given by:

$$X \in \{S, I, R\} \tag{7}$$

where S is the susceptible state, I is the infectious state and R is the removed or recovered state. Here I denotes a host infected by a computer worm and R the state of a host after the worm is removed.

An example of this definition is a PC with Microsoft Windows® without any antivirus installed, and therefore vulnerable to an attack by malware; hence, its state is susceptible S.

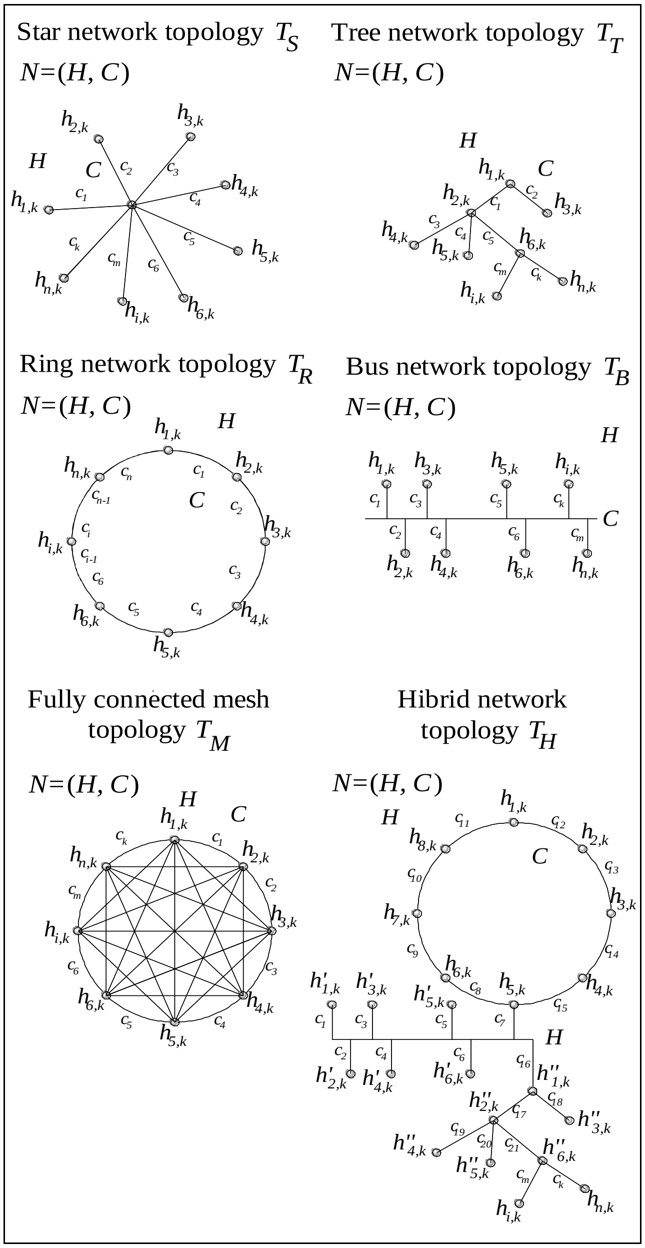

Figure 3. Network Topology.

Definition 5 (Susceptible State). Every host $h_{i,k}$ is in a susceptible state S if and only if it can change its status to infectious state I or to removed or recovered state R in $h_{i,k}$. See figure 4.

This is a first state in a host; in this condition, a host is vulnerable to any computer worm or state code. In the example of definition 4 it was explained that the state of a Windows® PC without an antivirus is always susceptible, as a consequence it will be exposed to a virus or computer worm attacks.

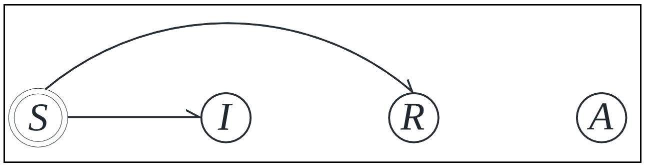

Figure 4. Susceptible State *S*.

Definition 6 (Infectious State). Every host $h_{i,k}$ is in infectious state I if and only if it can change its status to removed or recovered state R in $h_{i,k}$. See figure 5.

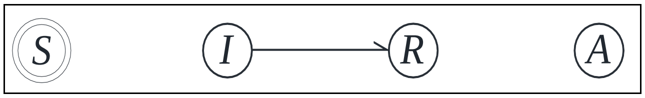

Figure 5. Infected State *I*.

The Infectious state is result of action by a computer worm in a susceptible host. As an example of an infected host is a Windows® computer without antivirus and connected to a network, which has been attacked by the worm *Sasser* or *I love you*.

Definition 7 (Removed or Recovered State). Every host $h_{i,k}$ is in removed or recovered state R, if and only if it can change its status to susceptible state S or to isolated state A in $h_{i,k}$. See figure 6.

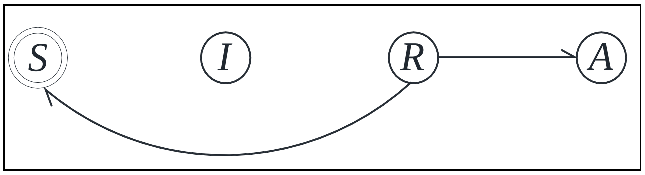

Figure 6. Removed or Recovered State *R*.

The recovered state is result of action by a vaccine in a infected host, in this sense considering the example in the previous definition, the infected computer by *Sasser* or *I love you* will be in removed state or recovered if it is applied correctly vaccine using McAfee®, Norton® or Kaspersky®.

Definition 8 (Isolated State). Every host $h_{i,k}$ is in isolated state A if and only if it cannot change its status to any other state in $h_{i,k}$.

The isolated state is result of action by a patch in a removed or recovered host. An example is a recovered computer by the action of an antivirus which was updated or patched, for example Windows® Service Pack 3.

Definition 9 (Set of Susceptible Hosts). Every set $\overline{S_k}$ of susceptible hosts in a network N is defined by the hosts set $h_{i,k}$ with state S in the index of evolution k, this is:

$$\overline{S_k} = \{h_{i,k} \mid E_{i,k} = S\} \qquad (8)$$

Every network without action of a vaccine is vulnerable or susceptible to a computer worm action. If one or more hosts are infected, then they can infect all susceptible hosts. An example of this is a set of computers in a cyber site without any antivirus installed.

Definition 10 (Set of Infected Hosts). Every set $\overline{I_k}$ of infected hosts in a network N is defined by the set of hosts $h_{i,k}$ with state I in the index of evolution k, this is:

$$\overline{I_k} = \left\{ h_{i,k} \middle| E_{i,k} = I \right\} \tag{9}$$

This set is the cardinality or number of elements infected by a computer worm action into a network. An example of this is a computers set in a cyber site affected by a computer worm.

Definition 11 (Set of Removed or Recovered Hosts). Every set $\overline{R_k}$ of removed or recovered hosts in a network N is defined by the set of hosts $h_{i,k}$ with state $\boldsymbol{R}$ in the index of evolution k, this is:

$$\overline{R_k} = \left\{ h_{i,k} \middle| E_{i,k} = R \right\} \tag{10}$$

This set is the cardinality or number of elements recovered by action of a vaccine into a network host. An example of this is a set of computers in a cyber site to which a vaccine has been applied.

Definition 12 (Set of Isolated Hosts). Every set $\overline{A_k}$ of isolated hosts in a network N is defined by the set of hosts $h_{i,k}$ with state $\boldsymbol{A}$ in the index of evolution k, this is:

$$\overline{A_k} = \left\{ h_{i,k} \middle| E_{i,k} = A \right\} \tag{11}$$

This set is the cardinality or number of elements isolated to computer worm action into a network. The isolated action is obtained by patch or upgrade software. An example of this is a set of computers in a cyber site with an updated operating system or a patch.

Definition 13 (State Code). Every state code is such when its action e changes the state $\boldsymbol{X}$ of a host $h_{i,k}$ to the state $\boldsymbol{Y}$ such as $\boldsymbol{X,Y} \in \{\boldsymbol{S, I, R, A}\}$ in the index $k+1$ in a time interval τ, which is the speed of action of the state code expressed in *card* (*affected-host*)*/second*, and Δt is the size of the sampling interval in seconds. This is:

$$e(h_{i,k}) : \left(h_{i,k} \middle| E_{i,k} = X \right) \rightarrow \left(h_{i,k+1} \middle| E_{i,k+1} = Y \right) \tag{12}$$

in time $\tau\Delta t$.

An example of state code can be a computer worm, vaccine, update, downgrade or virus; it may be any software that changes the state of a host.

Definition 14 (Malicious code). Every malicious code is such when its action m changes the susceptible state $\boldsymbol{S}$ of a host $h_{i,k}$ to the infectious state $\boldsymbol{I}$ in the index $k+1$ in a time interval $\beta\Delta t$ where $\beta$is the speed of action of the malicious. This is:

$$m(h_{i,k}) : \left(h_{i,k} \middle| E_{i,k} = S \right) \rightarrow \left(h_{i,k+1} \middle| E_{i,k+1} = I \right) \tag{13}$$

in time $\beta\Delta t$.

Examples of malicious codes are worms, viruses, trojans, spyware, etc.: any code that changes the state of a host to infected.

Definition 15 (Computer Worm). Every computer worm is a malicious code when its action ω changes the susceptible state $\boldsymbol{S}$ of a host $h_{i,k}$ to the infectious state $\boldsymbol{I}$ in $k+1$ and changes the susceptible state $\boldsymbol{S}$ of a host $h_{i,k+1}$ (immediate neighbor of $h_{i,k+1}$) to the infectious state $\boldsymbol{I}$ in $k+2$. This is:

$$\omega(h_{i,k}) : \left(h_{i,k} \middle| E_{i,k} = S \right) \rightarrow \left(h_{i,k+1} \middle| E_{i,k+1} = I \right) \tag{14}$$

in a time interval $\beta\Delta t$ such as:

$$\omega(h_{j,k+1}) : \left(h_{j,k+1} \middle| E_{j,k+1} = S \right) \rightarrow \left(h_{j,k+2} \middle| E_{j,k+2} = I \right) \tag{15}$$

in a time interval $2\beta\Delta t$ with $h_{j,k+1}$ immediate neighbor of $h_{i,k+1}$with βas the speed of action and propagation of the worm.

Every worm is a malicious code (malware) that attacks any host in order to change its state to infected; worms spread from host to host in a network without human action. Examples of computer worms are: *I Love You*, *Melissa*, *Sasser*, *Blaster*, etc.

Definition 16 (Vaccine). Every vaccine is a state code, and its action v changes to the infectious state $\boldsymbol{I}$ of a host $h_{i,k}$ to the removed or recovered state $\boldsymbol{R}$ in the index $k+1$ in a time interval $\gamma\Delta t$ where γ is the speed action of the vaccine. This is:

$$v(h_{i,k}) : \left(h_{i,k} \middle| E_{i,k} = I \right) \rightarrow \left(h_{i,k+1} \middle| E_{i,k+1} = R \right) \tag{16}$$

in the time $\gamma\Delta t$.

Vaccines are state codes with the capability of changing the state of a host to recovery. Examples of vaccines are: Panda®, Kaspersky®, McAfee®, Avast®, etc.

Definition 17 (Update). Every update is such when its action a changes the susceptible state $\boldsymbol{S}$ of a host $h_{i,k}$ to the removed or recovered state $\boldsymbol{R}$ in the index $k+1$ in a time interval $\sigma\Delta t$ where σ is the speed action of the update. This is:

$$a(h_{i,k}) : \left(h_{i,k} \middle| E_{i,k} = S \right) \rightarrow \left(h_{i,k+1} \middle| E_{i,k+1} = R \right) \tag{17}$$

in the time $\sigma\Delta t$.

The updates are software modules that protect hosts before computer worms or another malware changes their states to infected. Usually, operating systems like Microsoft Windows® or Apple OS® release updates periodically. An example of this is the update from MacOS IX® to MacOSX®, or Windows7® to Windows 8®, and it is important to mention that in each update new safety standards are added.

Definition 18 (Downgrade). Every downgrade is such when its action d changes the removed or recovered state $\boldsymbol{R}$ of a host $h_{i,k}$ to the susceptible state $\boldsymbol{S}$ in the index $k+1$in a time interval $\alpha\Delta t$ where $\alpha$is the speed action of downgrade. This is:

$$d(h_{i,k}) : \left(h_{i,k} \middle| E_{i,k} = R \right) \rightarrow \left(h_{i,k+1} \middle| E_{i,k+1} = S \right) \tag{18}$$

in time $\alpha\Delta t$.

Downgrade refers to reverting software back to an older version; downgrade is the opposite of upgrade. The disadvantage of this action is that it keeps hosts in state susceptible to computer worms. An example of this is the migration from MacOS X® to MacOS IX® orWindows8® to Windows 7®; its purpose is to ensure compatibility with older software.

Definition 19 (Patch). Every patch is a state code, and its action p changes the removed or recovered state $\boldsymbol{R}$ of a host $h_{i,k}$ to the isolated state A in the index $k+1$ in a time interval $\delta\Delta\, t$ where δ is the speed action of patch. This is:

$$p(h_{i,k}):\left(h_{i,k}\middle|E_{i,k}=R\right)\rightarrow\left(h_{i,k+1}\middle|E_{i,k+1}=A\right) \quad (19)$$

in the time $\delta\Delta t$.

A patch is a piece of software designed to update a computer program or its supporting data, to fix or improve it. This includes fixing security vulnerabilities isolating hosts. An example of this is the update for service pack in Microsoft Windows®.

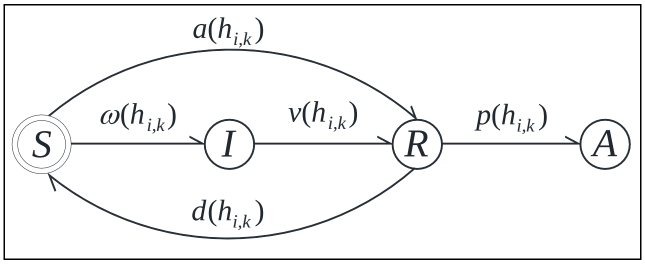

Figure 7. Evolution of the host´s states caused by a computer worm.

Definition 20 (Root host). In a network N, the root host $h_{1,1}$ is the first host affected by an action of the worm such that:

$$\omega(h_{1,1}):\left(h_{1,1}\middle|E_{1,1}=S\right)\rightarrow\left(h_{1,2}\middle|E_{1,2}=I\right) \quad (20)$$

in the time interval $\beta\Delta t$ such as:

$$\omega(h_{j,2}):\left(h_{j,2}\middle|E_{j,2}=S\right)\rightarrow\left(h_{j,3}\middle|E_{j,3}=I\right) \quad (21)$$

in a time interval $2\beta\Delta t$ with $h_{j,2}$ immediate neighbor of $h_{i,2}$.

In a practical case, the root host is the first infected host, the host where the infection starts. An example of this is the first computer infected at a cyber site due to the insertion of a USB stick with a computer worm.

Definition 21 (Epidemic). In a network N, an epidemic is defined as the change of the states $E_{i,k}$ of the nodes $h_{i,k}$ of S to I from the root host $h_{1,1}$ due to the action $\omega(h_{1,1})$ of a worm, following the pattern of propagation in accordance to the topology T of the host network N.

Finally, an epidemic is the rapid spread of a computer worm to a large number of susceptible hosts set in a known interval of time. Based on the example of the previous definition, *epidemic* is the spread of computer worm content on USB memory for all computers that have not been vaccinated in the cyber site.

Conclusions

In this paper, we presented a brief history of computer worms, malware with the capability to self propagate across network and a set of concepts involved. The theory of epidemics by computer worms and these concepts are expressed verbally without any formal notation, which is not useful for mathematical models and simulations. This paper presented 17 formal definitions based on discrete mathematics involving concepts related to computer worms. An additional contribution to computer worm theory is the definition of isolated state A of a host, or a host with a unique state that cannot be affected by the computer worm.

The 21 definitions developed in this work provide a theoretical foundation that serves as a basis for mathematical modelling of computer worms spread, which, when implemented on computers, can help to prevent the computer worms spread; this is explained in (Guevara *et al.*, 2014) where the creation and implementation of computational algorithms that relate the variables involved in the developed formal definitions is described. This constitutes one of the main contributions of this work, since the existing mathematical models to describe the dynamic behaviour of the spread of computer worms lack these.

References

Audelo, J., Castañeda, A., & Guevara, P. (2013). Gusanos Informáticos: de los inicios a su primer impacto en los gobiernos. *Revista Aleph Zero*, *17*(64), 22-25. Retrieved February 20, 2013 from http://www.comprendamos.org/az/alephzero/aleph64.pdf

Changchun, C., Gong, W., & Towsley, D. (2002). Code red worm propagation modelling and analysis. In *Proceedings of CCS'02*, (pp. 1-10). Retrieved February 20, 2013 from http://www.unix.ecs.umass.edu/~gong/papers/codered.pd

Chen, T. M., & Robert, J. M. (2004). Worm epidemics in high-speed networks. In *IEEE Computer* (pp. 48-53). Retrieved October 2014 from: http://dl.acm.org/citation.cfm?id=998510

Cohen, F. (1992). A formal definition of computer worms and some related results. *Computers & Security*, *11*(7), 641-652. Retrieved February 20, 2013 from http://game.all.net/books/tech/wormdef.pdf

Downing, D., Covington, M., & Mauldin, M. (2009). Dictionary of computer and internet terms (10th Ed.) (pp. 243, 335). New York: Barron's Educational Series, Inc. Retrieved February 20, 2013 from http://www.turuz.info/Sozluk/0329-Dictionary%20Computer%20Internet.Terms.pdf

Erbschloe, M. (2005). Trojans, Worms, and Spyware: A computer security professional's guide to malicious code (1st Ed.) (pp. 33). Massachusetts: Butterworth-Heinemann (Elsevier). Retrieved February 20, 2013 from http://edc.tversu.ru/elib/inf/0110.pdf

Guevara, P., Valdez, J., Audelo, J., & Delgado, G. (2014). Numerical approaching of SI epidemic model for spreading of computer worms, simulation and error analysis. *Revista Tecnura*, *18*(42), 12-23. Retrieved December 19, 2014 from http://tecnura.udistrital.edu.co/ojs/index.php/revista/issue/view/61/showToc

Hincapié, D., & Ospina, J. (2007). Bases para la modelación de epidemias: el caso del síndrome respiratorio agudo severo en Canadá. *Journal of Public Health*, 9(1), 117-128. Retrieved February 20, 2013 from http://redalyc.uaemex.mx/pdf/422/42290111.pdf

Juan, W., Cengyi, X., & Qifeng, L. (2010). A novel model for the internet worm propagation. In Proceedings of VI ICNC 2010 (pp. 2885-2888). Retrieved February 20, 2013 from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5584495

Kaspersky lab (2014). Retrieved October 25, 2014, from http://usa.kaspersky.com/internet-security-center/threats/viruses-worms

Kephart, J., & Chess, D. (1993). White Steve, computers and epidemiology. *Spectrum IEEE*, *30*(5), 20-26. Retrieved February 20, 2013 from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=275061

Kermack, W. O., & Mckendrick, A. G. (1927). A contribution to the mathematical theory of epidemics. *Proceedings of the Royal Society of London Series A*, *115*(772), 700–721. Retrieved February 20, 2013 from http://www.math.utah.edu/~bkohler/Journalclub/kermack1927.pdf

Microsoft TechNet (2013). Retrieved February 20, 2013 from http://technet.microsoft.com/es-mx/updatemanagement/bb245736.aspx

Nazario, J. (2004). Defense and detection strategies against internet worms (1st Ed.) (pp. 2-8). Massachusetts: Artech House. Retrieved February 20, 2013 from http://es.scribd.com/doc/59508754/Defense-and-Detection-Strategies-Against-Internet-Worms-2004#download

Onwubiko, C., Lenaghan, A. P., & Hebbes, L. (2005). An improved worm mitigation model for evaluating the spread of aggressive network worms. In *Proceedings of EUROCON 2005* (pp. 1710-1713). Retrieved February 20, 2013 from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1630303

Sabin, W. (2011). Appendix D: Glossary of computer terms (1st ed.) (pp. 16). New York: McGraw-Hill Companies. Retrieved February 20, 2013 from http://www.mhhe.com/business/buscom/gregg/docs/appd.pdf

Tao, L., Zhi-Hong, G., Zhengping, W., & Xianyong, W. (2007). Stability analysis of a delayed model of the spread of worms. In: *Proceedings of 2007 IEEE International Conference on Control and Automation* (pp. 3188-3190). Retrieved February 20, 2013 from http://ieeexplore.ieee.org/xpl/articleDetails.jsp?reload=true&arnumber=5234704

Tassier, T. (2005). SIR Model of Epidemics (pp. 1-9). New York: Department of Economics, Fordham University. Retrieved February 20, 2013 from http://www.fordham.edu/images/Undergraduate/economics/faculty/SIR.pdf

Yang, W., & Chenxi, W. (2003). Modeling the effects of timing parameters on virus propagation. In *Proceedings of WORM'03* (pp. 1-6). Retrieved February 20, 2013 from http://www.thehackademy.net/madchat/vxdevl/papers/avers/worm.pdf