ISSN 1392 – 1215 ELECTRONICS AND ELECTRICAL ENGINEERING 2008. No. 6(86)
ELEKTRONIKA IR ELEKTROTECHNIKA

T 120 SYSTEM ENGINEERING, COMPUTER TECHNOLOGY
SISTEMŲ INŽINERIJA, KOMPIUTERINĖS TECHNOLOGIJOS

Malware Propagation Modeling by the Means of Genetic Algorithms

N. Goranin, A. Čenys
Information security laboratory, Information System Department, FMF, Vilnius Gediminas Technical University,
Saulėtekio al. 11, Vilnius, Lithuania; phone: +370 6 5656541; email: ngrnn@fmf.vgtu.lt

Introduction

According to report [1] almost 60% of all companies in the United Kingdom have faced different kinds of information security breaches in 2006, almost 50% of which were caused by malware, i.e. viruses, worms, Trojans, etc, i.e. software that was created with the aim to harm computer software or to infect it without the permission and knowledge of a legal user [2]. In Lithuania according to RRT [3] approximately 70% of information security breaches that affected companies and home users were caused by malware. In 2006 the total number of new malicious programs was 41% up from 2005 [4] and even up to 172% according to [5] research. One more 2006 year trend marked by [5] is the use of worms as a means of propagation for other malware. Data for year 2007 is incomplete but according to [6] prediction a 60% percent increase from 2006 in unique malware is expected. Despite the differing percent numbers presented by different antivirus companies [4-5] and incomplete data for 2007 it is obvious that the rate of malware usage by e-criminals has the tendency to increase and protection against it is a crucial task.

In this article we analyze Internet worm propagation strategies and correlated propagation rates after the satiation phase, since this type of malware remains a significant and important part of all modern malware. Leaning on the described tendency of worm usage for transmission of other malware types (e.g. creating botnets) we predict that future worms will use more resistant propagation strategies, that ensure rather stable population (low population size decrease rates) of infected hosts after satiation phase. This challenge will require a model for evaluating propagation rates of newly originating worms after satiation phase. Propagation strategy is one of the most descriptive malware characteristics. Malware propagation rate prior to satiation can be evaluated by probabilistic and time-consuming evaluations of propagation strategy. On the other hand, only the analysis of prior worms' strategies with corresponding propagation rates (population size decrease) can be used for estimating the rates of propagation after the satiation.

We define malware propagation strategy as a combination of methods and techniques, used by malware to achieve tasks assigned to it by malware creator. So strategy suitable to achieve one specific task (e.g., infecting computers of home users) may be not useful for another (e.g., disrupting Internet functioning). Modern worms are usually created on a modular basis and may contain all or some of the following parts [7]: reconnaissance module, that scans the Internet for vulnerable hosts; attack module, that may exploit from one to many known vulnerabilities at potentially vulnerable host; communication module that allows worms to communicate between themselves or to transfer information to the worm management center; command module, that allows to accept commands; and intelligence module, that insures functioning of the communication module, since contains information how to find a neighbor worm for communication. Specific methods used in each of the modules are called patterns and strategy can be also defined as a combination of patterns. Strategy is also dependent on worm introduction techniques, i.e. method used to release worm to the wild, connection protocol used (e.g. TCP or UDP), etc.

Prior and related work

Existing malware propagation models mainly concentrate to forecasting the number of infected computers in the initial propagation phase (timeline from 1 to 9, Fig. 1).

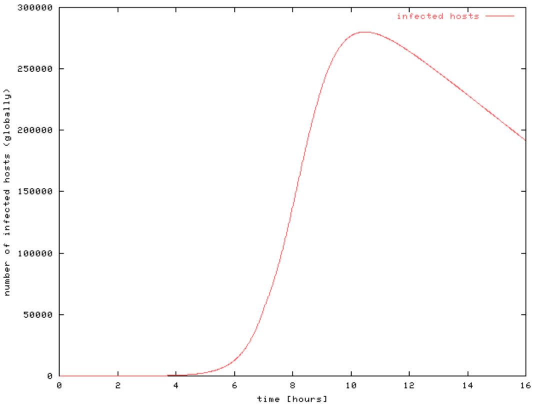


Fig. 1. Internet worm propagation graph

The first epidemiological model to computer virus propagation was proposed by [8]. Epidemiological models abstract from the individuals, and consider them units of a population. Each unit can only belong to a limited number of states. A SIR model assumes the Susceptible-Infected-Recovered state chain and SIS model – the Susceptible-Infected-Susceptible chain.

In a technical report [9] Zou et al. described a model of e-mail worm propagation. The authors model the Internet e-mail service as an undirected graph of relationship between people. In order to build a simulation of this graph, they assume that each node degree is distributed on a power-law probability function.

Malware propagation in Gnutella type Peer-to-Peer networks was described in [10] by Ramachandran et al. The study revealed that the existing bound on the spectral radius governing the possibility of an epidemic outbreak needs to be revised in the context of a P2P network. An analytical model that emulates the mechanics of a decentralized Gnutella type of peer network was formulated and the study of malware spread on such networks was performed.

Botnet propagation modeling using time zones was proposed by Dagon et al. [11]. The model uses diurnal shaping functions to capture regional variations in online vulnerable populations.

The Random Constant Spread (RCS) model [12] was developed by Staniford et al. using empirical data derived from the outbreak of the CodeRed worm. It assumes that the worm has a good random number generator that is properly seeded. The model assumes that a machine cannot be compromised multiple times and operates several variables: K is the constant average compromise rate, which is dependant on worm processor speed, network bandwidth and location of the infected host; a(t) is the proportion of vulnerable machines which have been compromised at the instant t, N·a(t) is the number of infected hosts, each of which scans other vulnerable machines at a rate K per unit of time. But since a portion a(t) of the vulnerable machines is already infected, only K·(1-a(t)) new infections will be generated by each infected host, per unit of time. The number n of machines that will be compromised in the interval of time dt (in which a is assumed to be constant) is thus given by:

$$n = (Na) \cdot K(1-a)dt \,. \tag{1}$$

N is assumed to be a large constant address space so the chance that worm would hit the already infected host is negligible. From this hypothesis:

$$n = d(Na) = Nda \,. \tag{2}$$

It is also possible to write

$$Nda = (Na) \cdot K(1-a)dt \,. \tag{3}$$

From this

$$\frac{da}{dt} = Ka(1-a) \,, \tag{4}$$

where

$$a = \frac{e^{K(t-T)}}{1+e^{K(t-T)}} \,. \tag{5}$$

So the model can predict the number of infected hosts at time t if K is known. The higher is K, the quicker the satiation phase will be achieved by worm. As [7] states, that although more complicated models can be derived, most network worms will follow this trend.

Other authors [13] propose the AAWP discrete time model, in the hope to better capture the discrete time behavior of a worm. However, according to [14] continuous model is appropriate for large scale models, and the epidemiological literature is clear in this direction. The assumptions on which the AAWP model is based are not completely correct, but it is enough to note that the benefits of using a discrete time model seem to be very limited.

On the other hand Zanero et al in [14] propose a sophisticated compartment based model, which treats Internet as the interconnection of autonomous systems, i.e. subnetworks. Interconnections are a so-called “bottlenecks”. The model assumes, that inside a single autonomous system (or inside a densely connected region of an AS) the worm propagates unhindered, following the RCS model. The authors motivate the necessity of their model via the fact that the network limited worm Saphire which was using UDP protocol for propagation was following the RCS model till the “bottlenecks” were flooded by its scans.

Zou et al in [15] propose a two-factor propagation model, which is more precise in modeling the satiation phase taking into attention the human countermeasures and the decreased scan and infection rate due to the large amount of scan-traffic. The same authors have also published an article on modeling worm propagation under dynamic quarantine defense [16] and evaluated the effectiveness of several existing and perspective worm propagation strategies [17].

Propagation rate estimation model

In the model proposed the worm’s propagation strategy S(worm name) is described by the following attributes:

1. OS_PLATF - describes the OS platform the worm can function on;
2. EXPL_1 … EXPL_N - describes the first exploit to be included in worm’s body. The first exploit is compulsory, since at least one exploit is necessary for worm’s propagation. Exploits from 2 to 8 may or may not be included;
3. IP_GEN - describes potential victim’s IP address generation algorithm;
4. TRANSF - describes worm’s body transfer mechanism;
5. MEM - describes type of memory the worm uses;
6. HIER - describes worm’s network hierarchy;
7. COM - describes worms‘ communication algorithm;
8. EXEC - describes remote worm management features;

9. ADD - describes additional worm functionality features;

10. EVOL - describes worm's evolution.

Since it is not possible to describe all attribute value ranges due to space limitations we present a sample strategy representation for a worm with a complex propagation strategy ("-" marks attributes that were not included):

S(Ramen)=("Linux", EXPL_1="FTP port", EXPL_2="RPC port", EXPL_3=" LPR attack", EXPL_4-EXPL_8="-", IP_GEN="Random, range from 128/8 to 224/8", TRANSF="TCP/IP", MEM="-", HIER="Centralized hierarchy", COM="child-management host", EXEC="http interface", ADD="deface, protect against reinfection", EVOL="-").

A data file with records of the following structure "S(worm name), propagation rate" was generated. Data file size – 100 data records. Propagation rate values described the change of population size (in fact propagation with negative sign) of infected hosts compared to population size at satiation phase during one month. Value assignment was based on worm descriptions from different antivirus software vendors, such as Symantec, MacAfee, etc. and information collected from security forums. The created data set cannot be treated as absolutely precise, since propagation rates from different information sources were used and this is challenge for further model improvements. Anyway, this does not minimize the value of the method for propagation rate estimation after the satiation phase since model approach was tested to be correct. The propagation rate values are described in Table 1:

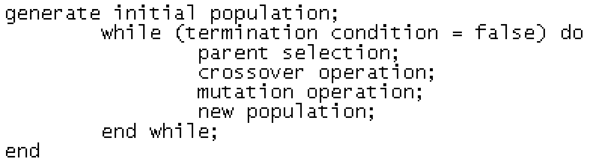
Table 1. Propagation values

Propagation value	Population decrease
Low	0-20%
Medium-low	20%-40%
Medium	40%-60%
Medium-high	60%-80%
High	80%-100%

That means, the lower is the propagation value, the more stable population of infected hosts is created by the corresponding propagation strategy. The generated data file was supplied to the GAtree [18] program for decision tree generation.

The main task of the decision tree created in our experiment is to assign the propagation value to the supplied for estimation propagation strategy. This task is very important when a new virus epidemic starts, the worm's propagation strategy is already known and the estimation of worm's propagation after satiation phase is needed.

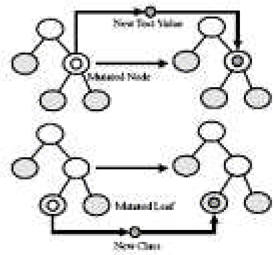
GAtree acts according to classical genetic algorithm (Fig. 2.) only modifying the chromosome representation from binary to tree. Decision tree fitness is evaluated by applying it to the test data.



```
generate initial population;
        while (termination condition = false) do
                parent selection;
                crossover operation;
                mutation operation;
                new population;
        end while;
end
```

Fig. 2. Genetic algorithm pseudo-code

The mutation and crossover operations are performed with tree structures as shown on Fig. 3 and Fig. 4.



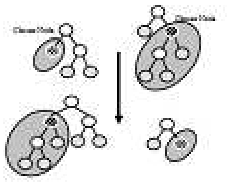
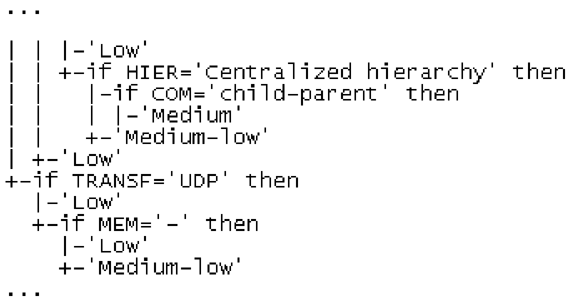


Fig. 3. Mutation [18]

Fig. 4. Crossover [18]

Since it is not possible to present the whole generated decision tree due to its size (43) only a fragment is shown on Fig. 5. Best genome score was equal to 0.571384, average genome score was equal to 0.516793.



```
...
| | |-'Low'
| | +-if HIER='Centralized hierarchy' then
| |    |-if COM='child-parent' then
| |    | |-'Medium'
| |    +-'Medium-low'
| +-'Low'
+-if TRANSF='UDP' then
  |-'Low'
  +-if MEM='-' then
    |-'Low'
    +-'Medium-low'
...
```

Fig. 5. Decision tree fragment

The decision tree efficiency was tested against 5 worms with known propagation values that were not included in the data file. All the test examples were classified correctly and proved the model efficiency.

Conclusions

Analysis of current research in sphere of malware propagation modeling was performed. The model, including virus strategy representation format and genetic algorithm based decision tree generation, for estimating the propagation rates of known and perspective Internet worms after their propagation reaches the satiation phase was presented. The proposed estimation algorithm, presented as a decision tree, was based on the known worms' propagation strategies with correlated propagation rates analysis. The model performance was tested on a worms with a known propagation rate after the satiation and proved its efficiency. The proposed model enables us to estimate the worm's propagation rates after satiation. The proposed model can be used as a framework for other malware types propagation rate analysis.

References

1. **Pricewaterhouse Coopers.** Information security breaches survey 2006 // Technical report. – UK Department of Trade and Industry, 2006.
2. **Szor P.** The Art of Computer Virus Research and Defense. – Addison Wesley Professional, 2005.
3. **LR Ryšių reguliavimo tarnyba.** 2007-ųjų metų tinklų ir informacijos saugumo būklės Lietuvoje tyrimas, įmonių apklausa // Technical report. – LR Ryšių reguliavimo tarnyba, 2007.
4. **Gostev A.** Kaspersky Security Bulletin 2006: Malware Evolution // Technical report. – Kaspersky Lab, 2007.
5. **Corrons L.** PandaLabs' Annual Report // Technical report. – PandaLabs, 2007.
6. **Schmugar C.** Malware estimation for 2007 // McAfee News. – McAfee Avert Labs, 2007.
7. **Nazario J.** Defense and Detection Strategies against Internet Worms. – Artech House, Inc., 2004.
8. **Kephart O. J., White S. R.** Directed-Graph Epidemiological Models of Computer Viruses // Proceedings of the 1991 IEEE Computer Society Symposium on Research in Security and Privacy. – Oakland, California, USA. – 1991. – P. 343–359.
9. **Cliff C. Zou, Don Towsley, Weibo Gong.** Email Virus Propagation Modeling and Analysis // Technical report TR-CSE-03-04. – University of Massachussets, Amherst, 2004.
10. **Ramachandran K., Sikdar B.** Modeling malware propagation in Gnutella type peer-to-peer networks // Parallel and Distributed Processing Symposium, IPDPS. – 2006. – Vol. 20.
11. **David Dagon, Cliff Zou, Wenke Lee.** Modeling Botnet Propagation Using Time Zones // 13th Network and Distributed System Security Symposium NDSS. – 2006.
12. **Staniford S., Paxson V., Weaver N.** How to 0wn the Internet in Your Spare Time // Proceedings of the 11th USENIX Security Symposium. – 2002.
13. **Zesheng Chen, Lixin Gao, Kevin Kwiat.** Modeling the Spread of Active Worms // Proceedings of IEEE INFOCOM 2003. – IEEE, 2003.
14. **Serazzi G. Zanero S.** Computer Virus Propagation Models // Lecture Notes in Computer Science. – Springer-Verlag, 2004. – P. 26–50.
15. **Cliff Changchun Zou, Weibo Gong, Don Towsley.** Code Red Worm Propagation Modeling and Analysis // Proceedings of CCS'02. – Washington DC, USA. – 2002.
16. **Cliff Changchun Zou, Weibo Gong, Don Towsley.** Worm Propagation Modeling and Analysis under Dynamic Quarantine Defense // Proceedings of WORM'03. – Washington DC, USA. – 2003.
17. **Cliff Changchun Zou, Weibo Gong, Don Towsley.** On the performance of Internet worm scanning strategies // Performance Evaluation. – Elsevier, 2005. – Vol. 63. – P. 700–723.
18. **Papagelis A., Kalles D.** Breeding Decision Trees Using Evolutionary Techniques // Proceedings ICML'01. – Williamstown, 2001.

Submitted for publication 2008 02 15

N. Goranin, A. Čenys. Malware Propagation Modelling by the Means of Genetic Algorithms // Electronics and Electrical Engineering. – Kaunas: Technologija, 2008. – No. 6(86). – P. 23–26.

Existing malware propagation models mainly concentrate to forecasting the number of infected computers in the initial propagation phase. In this article we propose a genetic algorithm based model for estimating the propagation rates of known and perspective Internet worms after their propagation reaches the satiation phase. Estimation algorithm is based on the known worms' propagation strategies with correlated propagation rates analysis and is presented as a decision tree, generated by GAtree v.2 application. Genetic algorithm approach for decision tree generation is selected taking into consideration the efficiency of this method while solving optimization and modeling tasks with large solution space. The performed tests have shown that the proposed model is efficient and can be used as a framework for modeling propagation rates after the satiation phase of different malware types. Ill. 5, bibl. 18 (in English; summaries in English, Russian and Lithuanian).

Н. Горанин, А. Чянис. Моделирование распространения вредоносного программного кода при помощи генетического алгоритма // Электроника и электротехника. – Каунас: Технология, 2008. – № 6(86). – С. 23–26.

Основной целью существующих на сегодняшний день моделей распространения вредоносного программного кода является прогноз числа заражённых компьютеров в начальной стадии его распространения. В этой статье предлагается модель, целью которой является оценка скорости распространения червей в сети Интернет после достижения фазы насыщения. Алгоритм оценки основан на анализе стратегий распространения известных червей и соответствующих скоростей распространения после фазы насыщения. Представление алгоритма организовано в виде дерева решений, сгенерированного при помощи программного обеспечения GAtree v.2. Генетический подход к генерированию дерева решений обоснован эффективностью данного метода при решении сложных задач моделирования и оптимизации. Проведённые тесты показали эффективность модели и возможность её использования при оценке скорости распространения других типов вредоносного программного кода. Ил. 5, библ. 18 (на английском языке; рефераты на английском, русском и литовском яз.).

N. Goranin, A. Čenys. Kenksmingo programinio kodo plitimo modeliavimas genetiniais algoritmais // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2008. – Nr. 6(86). – P. 23–26.

Esami kenksmingo programinio kodo plitimo modeliai yra orientuoti į užkrėstų kompiuterių skaičiaus prognozavimą pradiniame plitimo etape. Šiame straipsnyje siūlomas genetiniu algoritmu pagristas modelis, skirtas žinomų ir perspektyvių interneto kirminų plitimo greičiams nustatyti po prisisotinimo fazės. Algoritmas pagrįstas žinomų kirminų plitimo strategijų ir susijusių plitimo greičių analize ir pateikiamas kaip sprendimų medis, sugeneruotas GAtree v.2 programine įranga. Genetinis algoritmas pasirinktas sprendimų medžiui generuoti atsižvelgiant į metodo efektyvumą sprendžiant optimizavimo ir modeliavimo uždavinius, turinčius didelę sprendimų erdvę. Atlikti algoritmo testai parodė, kad pasiūlytas modelis yra efektyvus ir gali būti naudojamas kaip pagrindas vertinant kito kenksmingo programinio kodo plitimo greičius po prisisotinimo fazės. Il. 5, bibl. 18 (anglų kalba; santraukos anglų, rusų ir lietuvių k.).