Defending Polymorphic Worms in Computer network using Honeynet

aUpendra Kumar, bBimal Kumar Mishra, cG. Sahoo
a,cDepartment of Information Technology, Birla Institute of Technology, Mesra, Ranchi, India – 835 215
bDepartment of Applied Mathematics, Birla Institute of Technology, Mesra, Ranchi, India -835 215

Abstract
We propose a defense mechanism in computer network using gate-translator, double honeynet, sticky honeypot and antivirus engine of CloudAV, which attracts polymorphic worms. An algorithm is proposed to detect and remove the polymorphic worms and innocuous traffic related packets. Future antivirus is implemented on logically detached unused system.

Keywords - Polymorphic worm, Honeynet, Honeypot, Sticky honeypot, Cloud computing.

1. Introduction
Incoming polymorphic worms’ detection and removal are measure challenges of computer network security. A polymorphic worm is a worm that changes its appearance with every instance. Polymorphic codes are itself developing codes that mutate each copy by keeping original code unchanged but it changes its pattern each time and sends a copy to another system [1-5]. Latest worms often share same invariant content due to the fact that they exploit the same vulnerability. Further we can say that single contagious byte string signature cannot match polymorphic worms robustly, so to match exploit’s structure of worms, a family of signatures of polymorphic worms is used[ 6-10 ].

Polymorphic worms are detected by honeypot. Honeypot is an exciting new technology with enormous potentiality in the security community. The concept were first introduced by several icons in computer security specifically by Cliff stall in the book “cuckoo’s Eggs”, and Bill Cheswick’s paper,” An Evening with Berferd”, since then honeypot continued to evolve developing into the power security tools[11-13]. There are several categories of honeypot but mainly we have used high interaction honeypot and low interaction honeypot. High interaction honeypot works on real operating system (O.S) to attract and analyze incoming threats through internet .It constantly try to provide safer network from any type of attack. It has been specially defined for research honeypot. The example of high interaction honeypot is honeynets or honeynet. The honeynet is complex to design, implement and maintain, but able to handle huge risk. Low interaction honeypot minimizes the risk associated with network but never provide real O.S to attract or interact with hacker or malicious code in computer network, even if it is easy to deploy and maintain. Honeyed, sticky honeypot are examples of low interaction honeypot. Both work on the concept of monitoring unused IP address. Anytime it can see a connection attempt to an unused IP, pretending to the victim. It detach and logs any connection to any UDP or TCP port and also we can configure emulated service to monitor specific port such as an emulated FTP server to monitor TCP ports .It detects and logs activity to capture attackers interaction with the emulated server[14-17].

Gate-translator is used as an edge router between local network and Internet. It detects suspicious and innocuous traffic and redirect them towards honeynet-I. Honeynet-I interacts with honeynet-II to generate maximum instance of polymorphic worms and redirecting old worms and innocuous traffics towards router. The new signatures of worms are generated by generating new signature .Generated new signature of worms is filtered by several engines of CloudAV antivirus as well as auto integrated blocks of future defined antivirus on the basis of behavioral detection engine [18-20].

A CloudAV: N-Version antivirus identifies malicious and unwanted software by multiple, heterogeneous engine in parallel to provide N-version protection. Cloud AV includes a light weight, cross platform host agent, with ten antivirus engine and two behavioral detection engine [21]

In this paper, we develop a frame work of computer network having two honeynet, a sticky honeypot, gate translator and unused IP address system. New worms are detected by generating new signature .Generated new signature of worms is filtered by several engines of CloudAV antivirus as well as auto integrated blocks of future defined antivirus on the basis of behavioral detection engine .If unused IP address system is not filtered then node is automatically quarantine and process of filtration is continued to removal as well as. Experts intervention may be implemented to remove polymorphic worms , if necessary.
2. System architecture

As depicted in figure 1, gate translator access innocuous and suspicious traffics and transmission is redirected towards honeynet-I. Honeynet-I distinguish old worm and innocuous packets from new polymorphic worm with the help of honeynet-II, having some predefined worm’s packets of database. Old worm and innocuous packets are redirected towards router for filtration. The filtered packets are accessible in computer network. Honeynet – II includes database of worms where newly detected worms forms clusters. Each cluster in the RDBMS passes to the clustering components to perform new clusters and help to produce new accurate signatures. The new clusters produce by clustering components are passed to the signature generator component, which generates multiple signatures for each new cluster, using particular algorithm [22-26]. The new worm signature is redirected towards sticky honeypot to minimize instance of worm and again redirect towards unused IP address system .In this frame work, we have deployed and evaluated production quality in cloud antivirus, which has ten antivirus engines and two behavioral detection engines to execute on unused IP address system. If CloudAV unable to remove newly detected worms then unused IP address system is automatically quarantined [27-28]. Since honeynet—II has set of blocks of antivirus to remove future polymorphic worms, which are developed with the help behavioral detection engine which is deployed on unused system continuously till the removal of polymorphic worms.

3. Algorithm

a. Gate-translator captures Innocuous and Suspicious traffic and redirect them towards honeynet-I.
b. Honeynet-I distinguish old worm and innocuous traffic from new polymorphic worms, with the help of predefined signature of worms.
References