Detecting Mobile Spam Botnets Using Artificial immune Systems

Ickin Vural, Hein Venter

Chapter 14

DETECTING MOBILE SPAM BOTNETS USING ARTIFICIAL IMMUNE SYSTEMS

Ickin Vural and Hein Venter

Abstract Malicious software infects large numbers of computers around the world. Once compromised, the computers become part of a botnet and take part in many forms of criminal activity, including the sending of unsolicited commercial email or spam. As mobile devices become tightly integrated with the Internet, associated threats such as botnets have begun to migrate onto the devices. This paper describes a technique based on artificial immune systems to detect botnet spamming programs on Android phones. Experimental results demonstrate that the botnet detection technique accurately identifies spam. The implementation of this technique could reduce the attractiveness of mobile phones as a platform for spammers.

Keywords: Botnets, mobile devices, malware, artificial immune systems

1. Introduction

Unsolicited bulk email or spam is predominantly sent by criminal entities who use compromised computers running botnet software [7]. A botnet leverages the computational resources and bandwidth of a large number of computers to send massive amounts of spam.

Until recently, mobile devices were limited in their resources and functionality. However, because of their rapid increases in computational power, features and connectivity, coupled with their largely unexplored code bases and frequently discovered security flaws, mobile devices are becoming ideal candidates for recruitment in botnets.

Mobile devices access the Internet using High Speed Downlink Packet Access (HSDPA) and General Packet Radio Service (GPRS) [9]. The connections between the Internet and mobile devices act as gateways for malware to move from the Internet to mobile networks. As more

transactions are conducted using mobile devices, vendors must provide mobile applications that ensure security and ease of use [5]. An implementation that enables users to identify botnets on their mobile devices would slow the emergence of Short Message Service (SMS) spam, and reduce the attractiveness of mobile devices to spammers.

This paper describes a technique based on artificial immune systems to detect botnet spamming programs on Android phones. Experimental results demonstrate that the botnet detection technique accurately identifies spam on an infected phone.

2. Background

Malware has begun to appear on mobile devices [13]. SymbOS Mobispy [13], the first well-known mobile device spyware, remotely activated infected phones and turned them into eavesdropping devices, secretly sending copies of text messages to malicious entities. RedBrowser [13] for J2ME is a Trojan that pretends to access Wireless Application Protocol (WAP) web pages via SMS messages; in reality, however, it sends SMS messages to premium rate numbers that charge users high fees. The targeting of mobile devices by malware underscores the possibility that these devices will soon be recruited into botnets.

2.1 Botnets

A bot network or botnet is a set of machines that have been compromised by a spammer using malicious software sent over the Internet. A bot is one infected machine in the network of infected machines that constitutes a botnet. The bot software hides itself on a host machine and periodically checks for instructions from the botnet administrator. Botnet administrators typically control their botnets using Internet Relay Chat (IRC) [1].

The owner of a compromised computer usually has no idea that it has been compromised until the ISP severs its Internet connection for sending spam. Because most ISPs block bulk email that they suspect to be spam, spammers typically use botnets to send low volumes of email from numerous infected computers. Thus, spam is not sent from just one suspicious computer, and the spam is traceable only to innocent individuals, not the botnet operator.

While the number of botnets appears to be increasing, the number of bots in each botnet is dropping [1]. In the past, botnets with more than 80,000 bots were common. However, currently active botnets typically consist of a few hundred to a few thousand bots. One reason is that smaller botnets are more difficult to detect [14].

3. Artificial Immune Systems

There is a growing interest in developing biologically-inspired solutions to computational problems. An example is the use of artificial immune systems (AISs) that mimic the adaptive response mechanisms of biological immune systems to detect anomalous events.

The primary component of a biological immune system is the lymphocyte, which recognizes specific "non-self" antigens that are found on the surface of a pathogen such as a virus [2]. Once exposed to a pathogen, the immune system creates lymphocytes that recognize cells with the abnormal antigens on their surface. The lymphocytes have one or more receptors that bind to cells carrying the abnormal antigens. The binding process prevents the abnormal antigens from binding to healthy cells and spreading the virus.

An important feature of a biological immune system is its ability to maintain diversity and generality. A biological immune system uses several mechanisms to detect a vast number of antigens (foreign non-self cells) using a small number of antibodies [10]. One mechanism is the development of antibodies through random gene selection. However, this mechanism introduces a critical problem – the new antibody can bind not only to harmful antigens but also to essential self cells. To help prevent serious damage to self cells, the biological immune system employs negative selection, which eliminates immature antibodies that bind to self cells. Only antibodies that do not bind to any self cell are propagated [4]. Negative selection algorithms have proven to be very good at differentiating between self (normal) and non-self (abnormal), and have, therefore, been used to to address several anomaly detection problems [3, 6].

A typical negative selection algorithm [2] begins by randomly generating a set of pattern detectors. If the pattern matches self samples, it is rejected. If the pattern does not match self samples, it is included in the set of new detectors. This process continues until enough detectors are created. The created detectors are then used to distinguish between self and non-self samples in new data.

4. Botnet Detection in Mobile Devices

A unique characteristic of artificial immune systems is that training only requires positive examples [6]. This is ideal in situations where a profile of non-self and the requisite training examples are not available. This makes artificial immune systems an ideal candidate for tackling the SMS spam classification problem for which only one class of pattern is available for training.

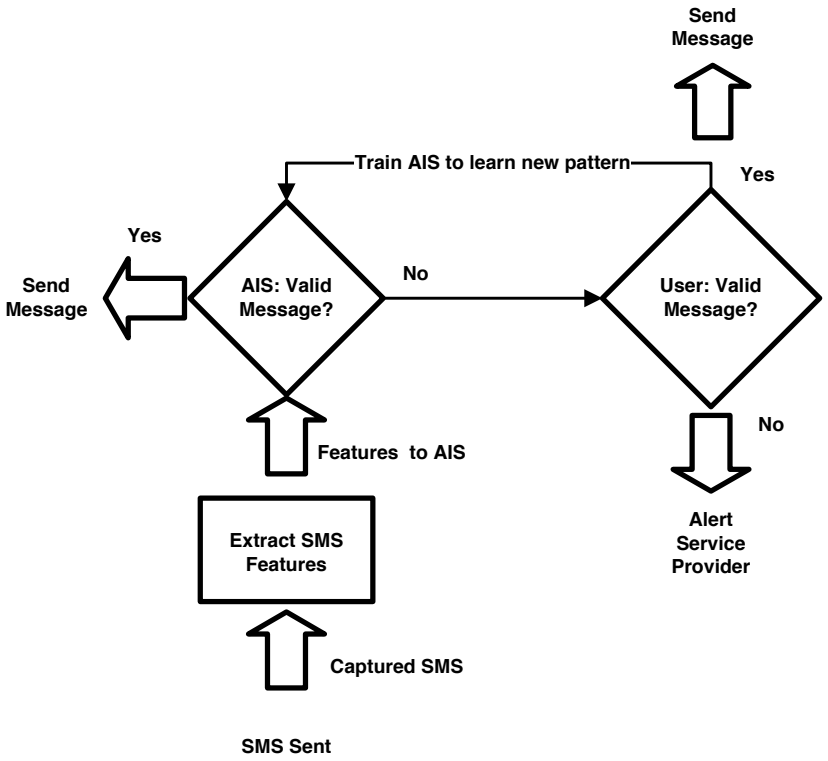

Figure 1. AIS-based botnet detection.

We have implemented a botnet detector based on artificial immune systems. The implementation currently executes on an Android mobile device emulator. Once ported to a mobile device, the botnet detector should be able to capture and analyze all outgoing SMSs.

The botnet detector learns to classify valid SMSs (self) from invalid SMSs (non-self). When the botnet detector encounters an SMS that it suspects to be invalid (non-self), it alerts the user and asks for confirmation that the message is valid. If the user indicates that the message is valid, then the detector learns to recognize the new pattern as a valid SMS. If the user indicates that the message is invalid, the detector sends an alert to the service provider and a digital forensic investigation can be initiated if necessary.

Figure 1 presents an overview of the artificial-immune-system-based botnet detection methodology. The mobile device user enters a text message and sends it to a recipient. The message is intercepted and specific message features are analyzed by the botnet detector. The detector uses the message features to determine whether the message is valid or

not. If the message is determined to be valid, it is sent onwards. If the message is determined to be spam, the user is asked to confirm that the message is valid. If the user confirms that the message is valid, the message is sent onwards and the detector learns to recognize the type of message as valid. If the user indicates that the message is invalid, an alert is sent to the network provider. This would typically occur only after several invalid responses are produced as a result of false alarms from the detector and/or user error.

4.1 SMS Message Patterns

Behavior models of individual mobile device users and groups of users can be constructed using statistical or social network analysis techniques [11]. The behavior models are used to establish the normal or expected behavior of mobile users. User behavior is then monitored and compared with current or recent usage data to detect abnormal behavior.

The botnet detector creates a signature (pattern) for each SMS message sent by the device. The pattern incorporates the following features:

- Total number of characters including white spaces.
- Total number of characters excluding white spaces.
- Number of capital letters.
- Number of white spaces.
- Number of punctuation characters.
- Number of digits.

The selected features correspond to a generic SMS signature because they permit the creation of a profile of the user's messaging behavior. Punctuation, capitalization and message length reveal valuable information about a user's SMS sending behavior. Additional characteristics will be added in the future to increase the accuracy of botnet detection.

Each pattern consists of real values assigned to each feature. A binding (detection) between a pattern and an SMS message occurs when the sum of the Euclidean distances between each corresponding feature value of the detector pattern and SMS is less than an affinity threshold.

4.2 Artificial Immune System Algorithms

This section describes two negative selection artificial immune system (AIS) algorithms that have been implemented. The two algorithms use the same pattern features and binding measure, but differ in how the affinity threshold is determined.

Algorithm 1 : Negative selection AIS.

Given A a set of valid SMS signatures
    n the user-defined number of antibodies
    e the user-defined affinity threshold
Initialize the set of patterns B to empty
while $|B| < n$ **do**
  Randomly generate pattern D
  for each message $a \in A$ **do**
    if dist(a,D) $\leq e$ **then**
      break while
    end if
  end for
  Add D to B
end while

Algorithm 1 builds a set of patterns B that do not match any pattern in the set A of valid SMS sample patterns. The number of elements n in B is a parameter that is set by the user. A match occurs when the Euclidean distance between the sample and the pattern is less than or equal to the user-defined affinity threshold e.

For each new message sent, the pattern corresponding to the message is measured against all the patterns in B. If the affinity of a pattern in B is less than or equal to e, the message is identified as not valid (non-self). If the user indicates that the message is valid, then the new signature is added to A and all the patterns in B that match the new signature are removed. The process is repeated until B contains n patterns.

Algorithm 2 : Negative selection AIS with minimum affinity.

Given A a set of valid SMS signatures
    n the user-defined number of antibodies
Initialize the set of patterns B to empty
while $|B| < n$ **do**
  Randomly generate pattern D
  $D_e = \min_{a \in A}$ dist(a,D)
  Add D to B
end while

Algorithm 2 eliminates the need for the user to set the affinity threshold e. Instead, for each new pattern D, the minimum of its Euclidean distances to the patterns in A is used as the affinity for the new pattern [6]. This means that the closest signature in A to the pattern D deter-

mines its affinity threshold D_e. Thus, the result is a set of patterns, each with its own affinity threshold. This removes the indefinite process of generating random patterns until a signature-tolerant pattern is located. Also, it reduces the number of parameters that have to be specified by the user.

Note that the affinity is calculated between the new message and each pattern in B. Therefore, if a message is incorrectly labeled as spam, then, instead of generating new patterns, the affinity thresholds are updated to the new minimum distances.

The size n of the antibody list is set to be 1.5 times the number of training messages. This value was identified based on trial-and-error experimentation to meet two goals, reduced database storage requirements on the mobile device and no overfitting of data during the training phase. This provides a tradeoff between spam pattern storage and detection.

5. Experimental Results

Experimental tests of the two artificial-immune-system-based detection algorithms used an Android smart phone emulator. The Android operating system is based on a modified version of the Linux kernel. There are currently more than 70,000 applications available for Android phones, which makes it the second-most popular mobile development environment after Apple iOS, which has more than 250,000 applications [8]. Developers write managed code in the Java language, controlling the device via Google-developed Java libraries [12].

The botnet detector captures all outgoing SMS messages, extracts the message features from the message body and saves them in an SQLite3 database. The detector then processes the data to determine if the message is valid or not. Figures 2 and 3 show responses to valid and invalid SMSs during the evaluation phase.

The two algorithms were tested on the same data and valid/invalid outputs. The training data consisted of 60 randomly-selected (valid) SMS messages that were sent by one of the authors of this paper over a period of one month. A second set of fourteen randomly-selected (valid) SMSs were used to test the accuracy of the botnet detector. The tests also used six spam (invalid) SMS messages that were selected from unsolicited messages received by the authors during the one-month period.

The results in Table 1 demonstrate that both the algorithms detect invalid SMS messages. Note that Algorithm 1 (first row) uses the user-defined affinity threshold while Algorithm 2 (second row) uses the affinity threshold based on the Euclidean distance.

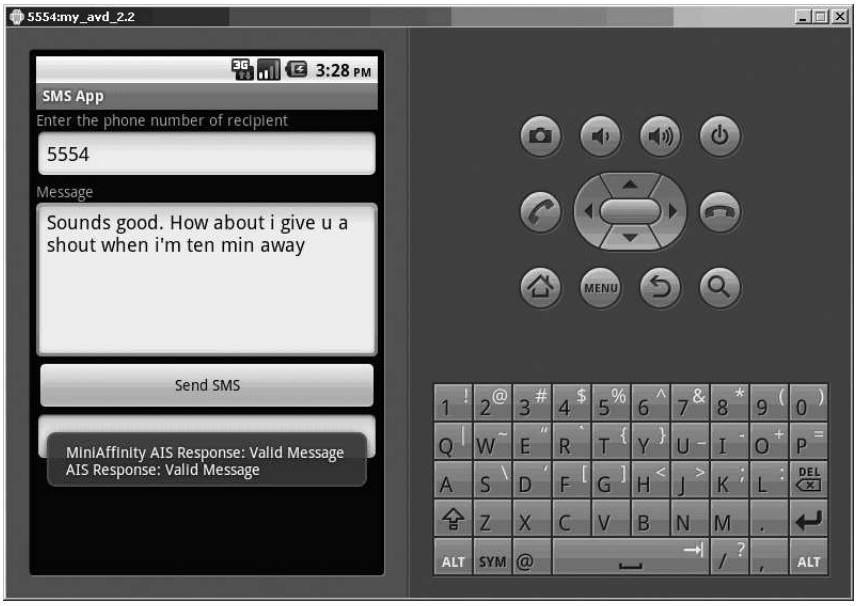

Figure 2. Response to a valid message.

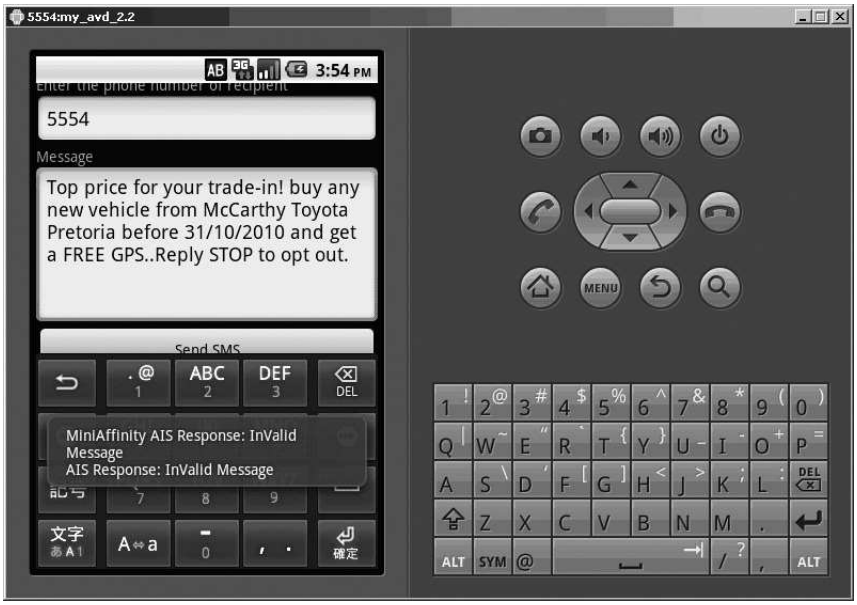

Figure 3. Response to an invalid message.

Table 1. Botnet detection results.

Algorithm	**Valid Message (Self)**	**Invalid Message (Non-Self)**	**Total Error**
AIS with user-defined threshold	86%	67%	19%
AIS with affinity	93%	83%	10%

Algorithm 1 identifies 86% of the valid messages and 67% of the invalid messages (spam) with a total error (incorrectly identified messages) of 19%. Algorithm 2, which uses the minimum Euclidean distance affinity, identifies 93% of the valid messages and 83% of invalid messages for a total error of 10%. The minimum Euclidean distance affinity yields better results than a user-defined threshold. Note that additional metrics could be incorporated to increase the accuracy of antigen binding. Also, the accuracy could be improved by increasing the size of the antibody list used to match non-self messages.

6. Conclusions

The artificial-immune-system-based botnet detector can be used to combat threats to mobile devices. In particular, the detector can help prevent mobile devices from sending SMS spam messages. Also, the implementation can function as a network forensic application that alerts service providers to botnet compromises.

The advantage of using an artificial immune system is that training only requires positive examples, which are readily available prior to an exploit. Future research will focus on improving detection accuracy by implementing a positive selection artificial immune system as well as a fuzzy logic detector. Additional features extracted from SMS messages (e.g., time of day and number of recipients) will also be used during the training phase to improve detection accuracy.

Acknowledgements

This research was supported by the National Research Foundation of the Republic of South Africa under Grant No. 2054024.

References

[1] E. Cooke, F. Jahanian and D. McPherson, The zombie roundup: Understanding, detecting and disrupting botnets, presented at the *Steps to Reducing Unwanted Traffic on the Internet Workshop*, 2005.

[2] D. Dasgupta (Ed.), *Artificial Immune Systems and Their Applications*, Springer-Verlag, Berlin, Germany, 1999.

[3] S. Forrest, S. Hofmeyer and A. Somayaji, Computer immunology, *Communications of the ACM*, vol. 40(10), pp. 88–96, 1997.

[4] S. Forrest, A. Perelson, L. Allen and R. Cherukuri, Self-nonself discrimination in a computer, *Proceedings of the IEEE Symposium on Research in Security and Privacy*, pp. 202–212, 1994.

[5] Georgia Tech Information Security Center, Emerging Cyber Threats Report for 2009, Georgia Institute of Technology, Atlanta, Georgia (hdl.handle.net/1853/26301), 2008.

[6] A. Graaff and A. Engelbrecht, Optimized coverage of non-self with evolved lymphocytes in an artificial immune system, *International Journal of Computational Intelligence Research*, vol. 2(2), pp. 127–150, 2006.

[7] Internet Service Providers' Association, Spam, Parklands, South Africa (www.ispa.org.za/spam), 2009.

[8] S. Jobs, Keynote address, presented at the *Apple Worldwide Developers Conference* (www.apple.com/apple-events/wwdc-2010), 2010.

[9] S. Kasera and N. Narang, *3G Mobile Networks: Architecture, Protocols and Procedures*, McGraw-Hill, New York, 2005.

[10] J. Kim and P. Bentley, An evaluation of negative selection in an artificial immune system for network intrusion detection, *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 1330–1337, 2001.

[11] M. Negnevitsky, M. Lim, J. Hartnett and L. Reznik, Email communications analysis: How to use computational intelligence methods and tools, *Proceedings of the IEEE International Conference on Computational Intelligence for Homeland Security and Personal Safety*, pp. 16–23, 2005.

[12] A. Perelson and G. Weisbuch, Immunology for physicists, *Reviews of Modern Physics*, vol. 69(4), pp. 1219–1268, 1997.

[13] J. Shah, Online crime migrates to mobile phones, *Sage*, vol. 1(2), pp. 22–23, 2007.

[14] T. Wilson, Botnets come roaring back in new year, *Information Week*, January 29, 2011.