Botnet Economics: Uncertainty Matters

Zhen Li
Department of Economics and Management
Albion College
Email:*zli*@albion.edu

Qi Liao, Aaron Striegel
Department of Computer Science and Engineering
University of Notre Dame
Email:{*qliao, striegel*}@nd.edu

Abstract

Botnets have become an increasing security concern in today's Internet. Thus far the mitigation to botnet attacks is a never ending arms race focusing on technical approaches. In this paper, we model botnet-related cybercrimes as a result of profit-maximizing decision-making from the perspectives of both botnet masters and renters/attackers. From this economic model, we can understand the effective rental size and the optimal botnet size that can maximize the profits of botnet masters and attackers. We propose the idea of using virtual bots (honeypots running on virtual machines) to create *uncertainty* in the level of botnet attacks. The uncertainty introduced by virtual bots has a deep impact on the profit gains on the botnet market. With decreasing profitability, botnet related attacks such as DDoS are reduced if not eliminated from the root cause, i.e. economic incentives.

I. Introduction

A hot topic nowadays in the Internet security community is botnets - referring to collections of compromised computers, or bots controlled by botnet masters. It is widely accepted that botnets impose one of the most serious threats to the Internet since they are predominantly used for illegal activities. For example, Rajab et al. find that a major contributor of unwanted Internet traffic - 27% of all malicious connection attempts - can be directly attributed to botnet-related spreading activity [1].

The attackers or hackers on the Internet were generally thought to be less financially driven in the past, i.e. motivated by self-fulfilment, fun, and proof of skills. Recently however, cybercriminals have been moving toward business models that involve building, exploiting and maintaining botnets. These cybercriminals collect, use, rent and trade botnets to make economic gains. Botnets can be exploited for various purposes, the most dominant uses including distributed denial-of-service attacks (DDoS), SMTP mail relays for spam (Spambot), ad click fraud, the theft of application serial numbers, login IDs, and financial information such as credit card numbers and bank accounts, etc. Almost all these tasks can be used to make money or have the potential to make money.

Researchers and Internet Service Providers (ISPs) have largely explored sophisticated technical only solutions with limited success. Recent trends note that the problems themselves are only growing, not abating. Existing technical approaches aim at either to prevent infected machines from reaching the target, or to redirect the visit of infected computers to a different site [2], [3]. Such defenses tend to be passive and inefficient mainly because current Internet architecture makes it extremely hard if not possible to differentiate a "pretend-to-be-legitimate" request from a "true legitimate" visit. Especially as botnets evolve quickly to become a significant part of the Internet, they are also increasingly hidden. New directions of thinking and effective alternatives are imminently required to deal with the problems at the root cause.

Today's botnet masters and attackers are seeking money, driven by profits, and motivated more by a desire to gain financially than to create havoc. Taking away the financial incentives that lead them to join malicious Internet activities in the first place is hence a promising new line of thinking in fighting the battle against botnet attacks. This study explores the worth and benefits by learning from economics and applys economic theories in the analysis of botnet-based attacks and activities.

Rational people think at the margin, one of the essential economic principles, suggests that when making economic decisions, people compare costs and benefits, and will only do things if the benefit of doing it exceeds the costs. The *cost-benefit analysis* would guarantee the maximum profit to an

economic agent. Applying the principle to for-pay attacks or other illegal activities, both botnet masters and attackers (who rent bots from previous) are by nature economic agents who participate in the botnet market seeking for economic returns. Similar to other rational behaviors like consumers or firms, botnet masters/attackers make economic decisions in order to reach the highest level of satisfaction, i.e., profit-driven botnet masters and attackers make their decisions regarding the optimal size of botnets, the effective size of bot rental, etc. to reap the maximum level of profit. Based upon the above, the contribution of this study is the systematic modeling of the botnet operation and utilization as a result of *profit-maximizing decision-making* from the perspectives of both botnet masters and attackers. The economic model developed in this study can help understanding the interaction between botnet masters, attackers, and defenders, the effective rental size and the optimal botnet size, cost and benefit, and many other aspects.

Another key contribution of this paper is to propose an interesting economic solution to the botnet problem. By introducing *virtual bots* (honeypots running on virtual machines that are to be compromised by the botnet masters), we create *uncertainties* and *interference* in the botnet market. As shown in this paper, these uncertainties have a tremendous impact on the effective botnet size and therefore the profitability of botnet operators and attackers. Botnet masters and attackers, being profit-driven rational economic behaviors, make decisions to seek the maximized profit, whose level depends on factors such as costs of operating botnets, payoff received for successfully disabling victim web sites, market rental price of botnets, etc. Given rational profit-driven botnet masters and attackers, both the size of rental and the size of botnets determined on a honeypot-free Internet black market are economically efficient. At any point in time, the capacity of the server limits the number of compromised machines supported, further limiting the number of bots rent and used to attack victims [4]. Therefore, having virtual bots in botnets reduces the probability of launching a successful attack and thus reduces the profitability of botnet market. The profit margin of the market is reduced not only through lowering revenue levels of market participants, but also through increasing costs of operating botnets. With falling profit margins, botnets and the associated attacks will eventually decrease if not outright disappear.

The remainder of the paper is organized as follows. Section II discusses technical background on botnet style DDoS attacks and defense mechanism, our threat model and the related work. Section III develops the assumptions, the variables, and profit levels of botnet masters and attackers in the benchmark model where virtual bots are not around. The profit maximization problem is formalized for both botnet masters and attackers. The fact of modeling the botnet masters' and the attackers' decision-making as a profit maximization problem allows us to find the optimal sizes of botnets, honeypots, and rentals used for attacks. Section IV extends the benchmark model to accommodate the existence of honeypots. We first assume the probability for a rental machine to be virtual is fixed, and then relax the assumption to analyze a more informative case in which the probability of fake bots is unknown to botnet masters and attackers. It also describes how this method can be used to understand and undermine botnet attacks from the root cause, i.e. economic incentives. The impacts on botnet masters, attackers, and defenders introduced by this uncertainty are analyzed in detail. Section V discusses technical deployment feasibility and a few challenges. We walk through examples with concrete numeric values coupled with graphical illustration. Finally, we conclude and propose future work in Section VI.

II. Background and Related work

In a botnet-style distributed denial of service (DDoS) attack, the attacker chooses a subset of botnets to either flood or consume end servers resources. Since those requests are not spoofed, they appear all *legitimate*, but much more intensely than normal use and causes the system to become busy, rendering the site unavailable to other legitimate users. Regardless of the type of DDoS attack, bandwidth depletion or resource depletion schemes, the goal of a DDoS attack is to impair the target's

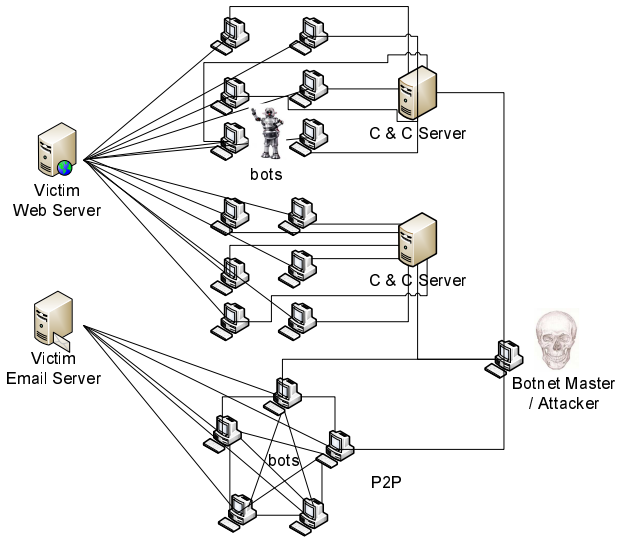

Fig. 1. A scenario of botnet attacks launched by robot computers (bots) controlled by the botnet master and attacker.

functioning, effectively shutting down the victim by forcing it to spend resources handling the attacker's traffic. An example of the botnet DDoS attack is illustrated in Figure 1.

Defending against botnet DDoS attacks is an extremely challenging problem. Traditionally, defenses against those attacks have focused only on technical solutions. Approaches include rate limiting/filtering the offense hosts [2], [3], tracing back [5]–[7], or host-based anomaly filtering [8]–[10]. These methods require either accurately identifying the source as "bad" or "good", constant updating signatures, or support from network architecture. This results in a never ending arms race between attackers and defenders, which is an undesirable position for a content provider.

We note that as researchers become more aware of the economic nature of Internet security problems, recent research has been seeking help from economic principles. To stem the flow of stolen credit cards and identity thefts, Franklin and Perrig [11] propose two technical approaches to reduce the number of successful market transactions, aiming at undercutting the cybercriminals verification or reputation system. The approach by Xu and Lee [8] uses game theory to model the attackers and defenders. Although their approach is by nature a technical DDoS defense, it is interesting to notice that they use a game-theoretical framework to analyze the performance of their proposed defense system and to guide the design and performance turning of the system.

The closest study to ours is the study by Ford and Gordon [12], which targets at malicious-code generated revenue streams. We both aim at designing botnet-disabling mechanisms from an economic perspective that are in the direct control of defenders. Nevertheless, there are noticeable differences between the two studies. In contrary to the focus on online advertising fraud, our model covers more general botnet attacks with a threat model focusing more on botnet DDoS attacks. Our contribution is that we model botnet masters' and attackers' decision-making as solving a profit maximization problem. Notably, we also incorporate the *diurnal pattern* and live population when modeling the botnet behavior. Depending on the optimal strategies botnet masters and attackers adopt, we illustrate in details how honeypots can be deployed to change economic motivations of illegal Internet practitioners. In this sense, we are in line with these researchers by claiming that botnet-related crimes will dramatically decrease if botnet masters give up on it - that is, when maintaining botnets becomes more troublesome than worthwhile.

We also propose a fresh new method of using virtual bots to introduce the *uncertainties* to the optimizing problem through analysis of those virtual bots' impact on the botnet market. Although the idea of honeypots is not new [13], honeypots have primarily been used for data collecting to understand the botnet or mapping the infected machines to track the control channel rather than

undermining botnets by removing the financial incentives of running and employing the botnet. By extending the functioning of honeypots in the direction of interfering with the money-driven Internet malicious activities, the value of honeypots is fundamentally improved, especially when taking into account the potential effectiveness of our proposed method.

III. The Benchmark Model

In this section, we consider a benchmark model in which virtual machines are not present to interfere with the botnet. We present the assumptions of the model, the variables and constant parameters, and the profit levels of both botnet masters and attackers as a result of their profit maximization decision-making.

A. Profit-driven Cybercriminals

Internet-based crimes have been shifting from reputation economy to cash economy. Today, large fraction of Internet-based crimes is profit driven and can be modeled roughly as rational behavior. The Internet underground market creates a large fortune. The exponential growth of botnet with millions of infected computers bought and traded on an underground market has evolved into billion-dollar "shadow industry" [14]. Being such a lucrative business, Internet illegal activities have been popular and hard to kill. Any effective approach aiming at eliminating such activities must remove the financial incentives out of them. Economic theories would help.

Botnet economics is by nature similar to other economics whereby rational individuals driven by profits make economic decisions to maximize their well-being. Applying the cost-benefit principle from economics to Internet crimes, a botnet master will keep botnets if the benefit of doing so is larger than the costs. Similarly, attackers will be better off if they commit an action whose benefits are larger than costs.

Evidence has been found that compromised machines are actually rent on underground markets [11]. It is realistic to model Internet market as the trading place where bots are rent to attackers for launching DDoS attacks. We choose to model botnet-based DDoS attacks first because of its straightforwardness. Moreover, (botnet-related) DDoS is still the primary concern for network security operations [15]. In the rest of the section, we build a theoretical model to illustrate how the two parties - botnet masters and attackers - make economic decisions in order to reap maximum profit.

B. Assumptions

The key assumption is the rationality of botnet masters and attackers. For any market, there must be a long-run equilibrium in which all market forces have been balanced. Suppose the Internet black market is in long-run equilibrium, We note the following assumptive parameters.

1) n^e is the minimum number of machines required to achieve a task (e.g. disable a website[1]). We assume that technical capability determines the size of n^e, which both botnet masters and attackers take as given. We refer to n^e as the effective number of rentals (and as we will see later, since it costs money to rent botnets, in the steady state, attackers' profit-maximizing size of rental is equal to n^e).
2) An attacker is only paid if the attack successfully disables the target site. The payment received by the attacker is denoted as M.
3) The rental price per bot (denoted as P) is determined on Internet black markets, which both botnet masters and attackers take as given.

[1]Alternatively, we can view n^e as the minimum number of accesses required to disable a website, and further define the number of accesses per machine to figure out the size of rental. We do not see it necessary to go into such details and believe our conclusions are not affected.

4) Botnet masters who manage bots use Command and Control (C&C) channel[2] to communicate with zombie computers in botnets. A typical C&C channel can host q machines simultaneously, which is also the live population on the C&C channel at any point in time.[3] The unit cost of maintaining a C&C channel is given at m.
5) A real bot machine operates on average t hours per day, and d days per week due to owner's diurnal patterns and physical constraints. Of all the live population, botnet masters randomly select bots to lease out.

In summary, the exogenous/given variables are the effective size of rentals (n^e), the number of machines a C&C channel can support at a point in time (q), the average cost of maintaining a C&C channel (m), the unit rental price of compromised machines (P), the payment for a successful attack (M), and how often a real machine operates (t and d).

C. Model Without Virtual Machines

In the benchmark model, we set up the profit maximization problems for a representative botnet master and a representative attacker where virtual machines are not present to interfere with the botnet. Profit is the difference between revenue and costs, both can be monetary and psychological. Since it is hard to measure or quantify psychological benefits and costs, we just focus on the monetary aspect of the analysis.

The profit maximization problems for a representative botnet master and a representative attacker are as follows.

For the attacker:

$$\begin{aligned} \max_{n}(Profit) &= M - P \times n \\ s.t. &\quad n \geq n^e \end{aligned} \tag{1}$$

where the subject condition requires that the attacker must rent at least the effective number of machines to launch a successful attack.

For the botnet master:

$$\begin{aligned} \max_{k,N}(Profit) &= P \times n - m \times k - a(N) \\ s.t. &\quad k \geq \frac{n}{q} \\ &\quad N \geq \frac{n}{\frac{t}{24} \times \frac{d}{7}} \end{aligned} \tag{2}$$

where N is the size of a typical botnet, which is simply the number of machines in a botnet. N is called the footprint of the botnet. $a(N)$ is the penalty function for the botnet master, measuring the economic losses suffered from being detected and arrested. Since the chance of being identified and arrested is higher as the size of the botnet increases, the penalty function is increasing in the size of the botnet ($a'(N) > 0$). The second restriction for the botnet master implies that the active members in the botnet ($N \times \frac{t}{24} \times \frac{d}{7}$) must be no smaller than the live population (n) because the botnet master can only rent out active machines. The first restriction for the botnet master suggests that the total number of C&C channels must be enough to support the n machines being leased.

The control variable for the attacker is the size of rental (n). The control variables for the botnet master are the number of C&C channels (k) and the size of the botnet (N) to maintain.

Given the consideration of both the attacker and the botnet master, the order of the decision making and the first-best model solutions are as the following.

[2]Although we are considering Internet Relay Chat (IRC), which is dominant C&C channel in today's botnet, the parameter for botnet maintenance costs can be defined accordingly based on the underlying technique adopted to control bots, whether through IRC or other decentralized systems such as P2P.

[3]Similar to the determination of n^e, how many bots, q, a C&C channel can host is determined by technological progresses and limited by the capacity of the channel. Given technology, q is fixed.

1) The attacker rents n machines to launch a successful attack; After the victim is taken down, the attacker receives M payment. Since it costs money to rent machines, at given M, the attacker's profit is maximized at $n = n^e$. In other words, in the steady state, the equilibrium number of rental is equal to the effective size of rental.
2) After observing the number of machines the attacker is willing to rent, the botnet master chooses the size of the botnet to maintain that will satisfy the rental needs of the attacker. Without uncertainty, since a typical machine runs t hours a day and d days a week, the steady-state size of the botnet is $N = \frac{n^e}{\frac{t}{24} \times \frac{d}{7}}$. Meanwhile, the botnet master needs to maintain enough C&C channels to host the n^e rental machines. Given the total revenue $P \times n^e$, maximizing profit is equivalent to minimizing costs, which is further equivalent to maintaining the minimum number of C&C channels $k = \frac{n^e}{q}$.

From above, when the botnet master and the attacker do not have to worry about virtual machines, efficient market results are achieved by realizing effective levels of rentals, number of C&C channels, and size of botnets. Without uncertainty, the botnet master's and the attacker's benchmark profits are deterministic. Let π_b stand for the profit earned by the botnet master and π_a be the profit for the attacker, their profit levels can be represented as follows, respectively.

$$\pi_b = P \times n^e - m \times \frac{n^e}{q} - a(\frac{n^e}{\frac{t}{24} \times \frac{d}{7}}) \tag{3}$$

$$\pi_a = M - P \times n^e \tag{4}$$

Examining the expressions of steady-state profits for the botnet master and the attacker, it can be seen that for the existence of the business, both profits must be non-negative. Combining the botnet master (seller of the botnet) and the attacker (buyer of the botnet), the market is profitable as long as both sides of the market are profitable,

$$M \geq P \times n^e \geq (m \times \frac{n^e}{q} + a(\frac{n^e}{\frac{t}{24} \times \frac{t}{d}})) \tag{5}$$

Adding (3) and (4), the size of the gains on the market is

$$\pi_a + \pi_b = M - m \times \frac{n^e}{q} - a(\frac{n^e}{\frac{t}{24} \times \frac{d}{7}}) \tag{6}$$

On current Internet black markets, the chance for a botnet master to be arrested is small. The widespread (and increasing) illegal botnet practices suggest that the profitability of the business may be quite significant, and hence participating in the market is attractive and rewarding.

One thing we do not take into account is the idle time of botnets - the time periods when botnets are not leased. The attacks do not happen all the time. The botnet master cannot rent the botnets as often as he/she would like. When the botnet is at idle, it receives no revenue and occurs only costs. The calculation of profits in the benchmark model is per successful attack. We can accommodate the concern of idle time straightforwardly by specifying the profit as the profit reaped in a period of time. The setup and solutions of the model are unchanged.

IV. Optimization Model with Virtual Machines

In the benchmark model, botnet masters and attackers earn profits and thus will remain in the market. To push them away from the market, we ought to reduce their profit level and make the business less attractive. Economic theory suggests that uncertainty is costly. When market situation becomes less clear for some reason, market participants would be reluctant to do the business and ask for higher compensation for the increased risks resulting from ambiguity. The idea provides a new approach to interfering with Internet underground market - to make it less efficient and less deterministic. We propose that creating honeypots for botnet masters to compromise will do the job.

In this section, we extend the benchmark model to allow the existence of honeypots in botnet. We first assume that the probability for a rental machine to be virtual is fixed, and later relax the assumption to analyze a more realistic and informative case in which market participants have no idea about the number of honeypots having been created.

A. Fixed Probability For A Rental Bot Being Virtual

The introduction of virtual machines creates uncertainty to the botnet in large. Virtual bots will not attack the victim as ordered. If still $n = n^e$ machines were rent, a number of inactive machines would make the attack unsuccessful. The actual size of rental (n) can no longer be equal to the effective size of rental (n^e). With some of n being virtual machines, renting n^e is not enough, implying that the new equilibrium size of rental must be larger than n^e.

We model the profit maximization problems for the botnet master and the attacker to show what happens with the introduction of virtual machines. For the time being, we assume that the probability for a rental machine to be virtual is fixed.

Let p_v denote the probability for a rental machine to be virtual, and p_v is fixed. The profit maximization problem for a typical attacker now looks as follows.

$$\begin{aligned} \max_n(Profit) &= M - P \times n \\ s.t. &\quad n \times (1 - p_v) \geq n^e \end{aligned} \tag{7}$$

For the botnet master, the profit maximization problem is the same as in the benchmark model since his/her decision-making is based upon the size of rental chosen by the attacker.

Solving the problems results in two conclusions:

1) To launch a successful attack, the attacker now has to rent $n = \frac{n^e}{1-p_v}$ machines, larger than in the benchmark model.
2) To accommodate the $n = \frac{n^e}{1-p_v}$ machines leased, the botnet master has to maintain $k = \frac{n^e}{(1-p_v)\times q}$ C&C channels. In the meantime, the new equilibrium size of botnet increases to

$$N = \frac{n^e}{(1 - p_v) \times \frac{t}{24} \times \frac{d}{7}} \tag{8}$$

If everything else remains unchanged, the profit for both the botnet master and the attacker are different from the benchmark model. For the botnet master, the profit may either go up or go down. On one hand, the botnet master's revenue increases due to more machines rent; on the other hand, the botnet master has to acquire more C&C channels to support the increased rental, and he also suffers a higher chance of being arrested. The botnet master's profit margin is now:

$$\pi_b^{v1} = P \times \frac{n^e}{1 - p_v} - m \times \frac{n^e}{(1 - p_v) \times q} - a(\frac{n^e}{(1 - p_v) \times \frac{t}{24} \times \frac{d}{7}}) \tag{9}$$

where π_b^{v1} represents the profit margin for the botnet master when the probability for a rental machine to be virtual is fixed at p_v.

The attacker's profit must decline. With the same payment for successfully taking down the victim, the attacker incurs larger costs of renting machines. The new profit level for the attacker is therefore

$$\pi_a^{v1} = M - P \times \frac{n^e}{1 - p_v} \tag{10}$$

where π_a^{v1} stands for the profit margin for the attacker when the probability for a rental machine to be virtual is fixed at p_v.

Adding (9) and (10), the size of the total gains on the market shrinks to

$$\pi_a^{v1} + \pi_b^{v1} = M - m \times \frac{n^e}{(1 - p_v) \times q} - a(\frac{n^e}{(1 - p_v) \times \frac{t}{24} \times \frac{d}{7}}) \tag{11}$$

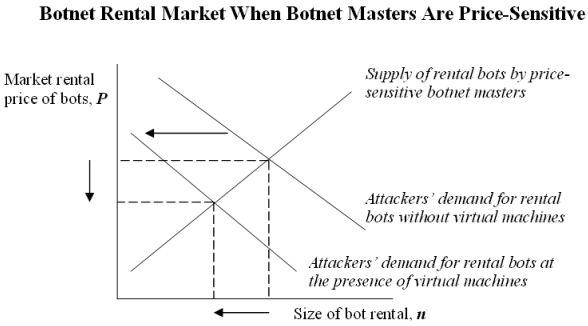

Fig. 2. In the underground market for botnets where botnet masters are price-sensitive, a supply and demand model suggests the decreased price and bot rental after introducing virtual machines.

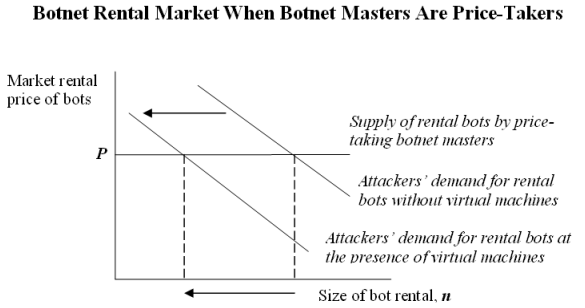

Fig. 3. In the botnet underground market where botnet masters are price-takers, a decreased bot rental is suggested at the presence of virtual bots.

Obviously, the existence of virtual machines lowers the incentives for attackers to rent machines. For the botnet master, the profit level depends on the rental price of machines P. The profit level decreases as the rental price P falls. If relaxing the assumption of a given rental price (that is, if P is allowed to adjust to market situations), the attacker's decreased demand for botnets will push down the rental price of machines (that is, P will fall). Market price P is further decreasing in p_v, thus a higher p_v will lower the botnet master's profit through two channels: lowered revenue due to lower price and higher costs of maintaining more C&C channels (Figure 2). The supply curve illustrates the botnet masters' willingness to sell at any given market rental price of bots. As the price rises, price-sensitive botnet masters' are willing to rent out more bots, hence the supply curve for price-sensitive botnet masters is upward-sloping. The supply and demand model suggests the decreased price and bot rental after introducing virtual machines.

Alternatively, Figure 3 illustrates the botnet rental market where botnet masters are price-takers. The demand curve summarizes attackers' willingness to pay for each rental bot at any given level of bot rental. The declined quality of bots due to the existence of virtual machines reduces attackers' willingness to pay. Accordingly, the new demand curve at the presence of virtual machines is lower than the demand curve without virtual machines. The supply and demand model in this case suggests the decreased bot rental after introducing virtual machines.

In the following analysis, we will hold market price as given. Price changes are not essential to our analysis because the rental price received by the botnet master is just the price paid by the attacker. Price fluctuations cause income redistribution between botnet masters and attackers rather than affecting the combined benefits of the market.

The analysis in this subsection shows how the introduction of virtual machines may alter economic benefits to interest parties. By creating virtual bots to disturb botnets, we've seen the possibility of reducing profitability of participating in Internet black markets, and hence reducing the incidence of black market activities. By reducing the potential profit levels of both botnet masters and attackers, creating virtual machines has a large potential to reduce unfavorable Internet practices.

B. Uncertainty For A Rental Bot Being Virtual

In previous subsection we demonstrate that creating honeypots reduces the attractiveness of participating in the black market for botnets. In this section we relax the assumption of a fixed p_v and introduce *uncertainty* to the market. In other words, this time p_v becomes unknown to black market participants (botnet masters, attackers, etc.). The following analysis shows that uncertain proportion of virtual machines will make the situation even harsh for botnet masters and attackers.

To that end, the model needs to be modified. We continue denoting the probability for a rental bot to be virtual as p_v, but it is unknown to the market this time. We denote the probability for a botnet

style attack to be successful as p_s, which depends on p_v and the total number of machines rent,

$$p_s = f(p_v, n^u) \tag{12}$$

where n^u is the size of rental in the uncertain environment. p_s is decreasing in p_v and increasing in n^u. (12) has a discrete format. $p_s = 1$ if $n^u \times (1 - p_v) \geq n^e$; $p_s = 0$ if $n^u \times (1 - p_v) < n^e$.

The first step of the game is still for the attacker to determine the number of machines to rent (n^u), which is the optimal solution to the attacker's profit maximization problem. The chance of launching a successful attack depends on how likely for a bot to be virtual. For DDoS attacks, payment is more likely predicated upon the target sites actually being disabled. Therefore, we can model the attacker's profit maximization problem as follows.

$$\begin{aligned} \max_{n^u}(Profit) &= E - P \times n^u \\ &= M \times p_s - P \times n^u \\ &= M \times f(p_v, n^u) - P \times n^u \\ s.t. \quad & n^u \times (1 - p_v) \geq n^e \end{aligned} \tag{13}$$

where we replace the probability of launching a successful attack p_s with its determinants p_v and n^u. E stands for the *expected* revenue of the attacker. To make the attack successful, the attacker has to rent at least $n^u = \frac{n^e}{1-p_v}$ machines. As $p_v \to 1$, $n^u \to \infty$.

Taking the first order derivative of the objective function with respect to n^u, we get the first order condition for the maximizing problem, $M \times f'(p_v, n^u) - P = 0$, or $f'(p_v, n^u) = \frac{P}{M}$, which implies that by observing market price of renting machines and the payment to be received after launching a successful attack, the attacker rents n^u such that the first order condition holds true.

If p_v were known to the attacker, the minimum size of rental would be $\frac{n^e}{1-p_v}$. The unknown probability p_v makes it impossible for the attacker to pin down the size of rental. If he rents too many, he will incur unnecessary costs; if he rents too few, the attack fails. He receives no payment and only pays rental costs. Thus, there is a trade off between rental costs and the odds of a successful attack.

We note that *the probability for a rental bot to be virtual (p_v) is different from the number of virtual machines as a percentage of the botnet size*. Let p_b represent the percentage of virtual machines in the botnet and V be the number of virtual machines in the botnet. Recall that N stands for the footprint of the botnet, then $p_b = \frac{V}{N}$. The number of "real" machines (machines that are not virtual) is thus $(N - V)$.

The machines that the botnet master can rent to the attacker must be live machines. When the botnet master needs to choose n^u machines from the botnet, he/she has to choose live machines. A real machine may have idle time as well as live time, while a virtual machine can run $24/7$. The chance for a virtual machine to be chosen is likely to be higher than that of a real machine. If the botnet master selects machines randomly from the live population, the chance for a virtual machine to be picked c_r and the chance for a real machine to be picked c_v have the following relationship: $c_v = \frac{24}{t} \times \frac{7}{d} \times c_r \geq c_r$.

Without virtual machines, the attacker rents $n = n^e$ machines and the botnet master keeps the size of the botnet at $N = \frac{n^e}{1-p_v}$. The chance for a real machine to be picked is $c_r = \frac{t}{24} \times \frac{d}{7}$. With the existence of virtual machines, the effective size of the botnet is $N^u = B + V$, where N^u is the size of botnet under uncertainty and B is the number of real machines. To satisfy the need of renting n^u machines, V, N^u and n^u have the relationship of

$$V + (\frac{t}{24} \times \frac{d}{7}) \times (N^u - V) = n^u \tag{14}$$

From (14) we can derive the probability for a machine in the n^u rental machines to be virtual,

$$p_v = \frac{V}{n^u} = \frac{V}{V + (\frac{t}{24} \times \frac{d}{7})(N^u - V)} \tag{15}$$

The profit-maximizing size of rental is equal to the minimum number of live machines. Since all virtual machines are active around the clock, V virtual machines are all selected for renting to the attacker. The uncertainty of p_v comes from the uncertainty of the number of virtual machines in a botnet. The uncertainty of p_v further leads to the ambiguity in the rental market for botnets, which reduces market efficiency.

The uncertainty also affects botnet masters' decision-making. A representative botnet master's profit maximization problem can be written as follows.

$$\begin{aligned} \max_{k,N}(Profit) &= P \times n^u - m \times k - a(N) \\ s.t. \qquad & k \geq \frac{n^u}{q} \\ & N \geq \frac{n^u}{\frac{t}{24} \times \frac{d}{7}} \end{aligned} \tag{16}$$

The constraint conditions illustrate that at any time, the botnet master must have enough C&C control channels and machines to stay in business. The solutions to the problem still take the following format: $k = \frac{n^u}{q}$ and $N = \frac{n^u}{\frac{t}{24} \times \frac{d}{7}}$. The uncertainty of n^u due to the unknown p_v leads to the uncertainty of k and N, both are increasing in n^u. With uncertain number of virtual machines V (and hence p_v) and size of rental, there is no way to determine the appropriate/effective size of the botnet.

The profit margins for the botnet master and the attacker are calculated as

$$\pi_a^{v2} = M \times f(p_v, n^u) - P \times n^u \tag{17}$$

$$\pi_b^{v2} = P \times n^u - m \times \frac{n^u}{q} - a(\frac{n^u}{\frac{t}{24} \times \frac{d}{7}}) \tag{18}$$

Adding (17) and (18), the size of the gains on the whole market is now

$$\pi^u = M \times f(p_v, n^u) - m \times \frac{n^u}{q} - a(\frac{n^u}{\frac{t}{24} \times \frac{d}{7}}) \tag{19}$$

Since $f(p_v, n^u) \leq 1$, $n^u > n$, and $a(\cdot)$ is increasing in n^u, the market profitability shrinks, meaning that the total benefit available for the two parties is smaller. Indeed, both parties are only left with a smaller profit margin than in the previous two cases.

It is important to go over the motivation and preferences of each interest party, and see the effects of an uncertain p_v.

- The attacker.
  The attacker decides the minimum/effective size of rental that guarantees a successful attack n^u, which is determined according to $f'(p_v, n^u) = \frac{P}{M}$. Given market prices of rental and attack, n^u is increasing in p_v. The attacker's profit is decreasing in p_v.
- The botnet master.
  By observing the number of machines the attacker is willing to rent, the botnet master decides the minimum/effective number of C&C channels and the size of botnet to maintain that allow at least n^u machines are alive ensuring there are always enough machines for renting. An uncertain p_v increases the botnet master's operation costs and may eventually reduce his/her profit if the market rental price of "low-quality" botnet drops and he/she further suffers reputational losses and an increased chance of being arrested. Note for both the attacker and the botnet master, undesirable costs incur.

- The defenders[4].
  The strategy is simply to create virtual slices/images on their computers to interfere with the botnet market. Both the botnet master's and the attacker's costs are directly and positively related to the probability for a bot to be virtual among the n^u rental machines. That is, p_v is the essential factor that is, if not fully, at least partially controlled by the defenders. Higher p_v will effectively reduce the profits earned by both the botnet master and the attacker. If p_v is high enough, renting botnets to launch attacks or other illegal activities may no longer be profitable. Even some profits remain, the reduced profit margin will certainly make the business not as attractive as before[5].

Although we have modeled the profit maximization decision-making for the attacker and the botnet master separately, the model conclusions will be the same if the two parties are combined to model the optimal results on the whole market. Therefore, if botnets are not rent to attackers but are used by botnet masters themselves to launch attacks, the model predictions work equally well.

V. Further Discussion and Case Study

First, a few counter-virtual measurements that might be adopted by the botnet master are discussed in this section, for example, what if the botnet master selects machines according to lifetime of being a botnet member rather than selecting machines randomly (or, what if the botnet master adopts a"first-in-first-out" strategy). What about insurance, would that help? Second, we walk through examples as case study coupled with graphical analysis of the model. Last, some technical deployment feasibility is discussed.

A. Counter-Virtual Strategies

First, let us look at "first-in-first-out" strategy. First-in-first-out means that the botnet master selects machines according to the length of being compromised. "Older" member bots are more likely to be chosen. This strategy may seem advantageous than random selection at the first sight, but it will not nullify our method. The first-in-first-out strategy simply imposes more challenges for researchers to develop approaches for preventing a virtual machine from being detected by the botnet master. Meanwhile, since virtual machines are not subject to the life cycle of a real machine, they tend to have longer lifetime, which can even increase the probability for a virtual machine to be selected.

If the botnet market becomes aware of the problems created by virtual machines, the botnet master may consider offering warranty or insurance to attackers and promises to replace inactive machines. This seems like a good idea but it would be very difficult for the botnet master to implement it because

1) All the warranty depends on the capability for the attacker/botnet master to find out which machine is inactive, which takes time;
2) Even the previous is possible, having virtual machines distributed widely among botnets and the fact that a virtual machine is more likely to be picked further complicate the situation;
3) Some type of attacks (such as DDoS) may be time-restricting. Once the first wave of attack fails, the target site may have been aware of the attack and initiated counter-attacking.

To counter the uncertainty created by unknown p_v, the attacker may rent $n^u = \frac{n}{1-p_v^g}$ machines at an estimated level of $p_v = p_v^g$. If $n^u = \frac{n}{1-p_v^g}$ turns out to be insufficient, the attacker then increases the intensity of attacks per (real) machine (upon detecting virtual machines). There are again two major difficulties with this counter-virtual strategy. The first is about the timing, i.e. how likely and quickly is it for the attacker and the botnet master to detect virtual machines? The second issue is the increased chance of being blocked if each real bot has to send more access requests. That is, it will

[4]Defenders refer to whoever has the incentive to run/maintain honeypots such as researchers and government agencies. While these organizations by law have desire to fight against cybercriminals, private parties may also be motivated to create honeypots if they are financially compensated. For example, a honeypot server may collect data on the botnet to sell to customers for development of infrastructure protection techniques.

[5]Furthermore, the increased likelihood for an attack to fail also increases the psychological costs of launching such an attack, which makes the practice even less interesting.

be harder for the attacker to mimic a human visitor. In other words, the heavier each machine attacks, the more likely will it be detected and filtered. Therefore, it is concluded that the strategy of creating virtual machines to blur Internet black markets is robust to above various possible counter-strategies that cybercriminals may adopt.

Indeed the most obvious and challenging counter-virtual strategy the botnet master may explore is to improve the detection of fake bots. For example, the botnet master may monitor whether bots participate in the attack or respond to other malicious commands as instructed. Section V-C discusses issues related to such countermeasure in more details.

B. Examples and Illustration

We now look at a case study with numerical examples and graphical illustration. From above, the essential component of our strategy is the uncertainty of p_v, or the ambiguous number of virtual machines that have been created (V). An interesting question is how large should V be to completely wipe off the profits reaped from participating the market. Since modeling botmastes and attackers respectively is equivalent to modeling the entire market, we focus on analyzing how the total size of the market profit is affected by changing the number of virtual machines, and figuring out the cutoff value of it.

Substituting (8) into (15), we express the number of virtual machines V as a function of the probability for a rental machine to be virtual p_v.

$$V = \frac{n^e \times p_v}{(1-p_v)(1-p_v \times (1-\frac{t}{24} \times \frac{d}{7}))} \tag{20}$$

The uncertainty of p_v makes it impossible to solve for the botnet size N^u and the size of market profit π^u. We assign some values to the parameters and show how the two variables (N^u and π^u) change with p_v.

For simplicity, suppose $n^u \geq \frac{n^e}{1-p_v}$ is satisfied, hence $f(p_v, n^u) = 1$. We also drop the penalty function from the market profit function[6]. The market profit (19) is simplified as

$$\pi^u = M - m \times \frac{n^e}{(1-p_v) \times q} \tag{21}$$

Given the parameters (M, m, n^e and q), we can solve for the cutoff p_v that reduces the market profit to break even (and if p_v exceeds the cutoff value, the market profit becomes negative). The formula of the cutoff p_v is

$$p_{v,cutoff} = 1 - \frac{m \times n^e}{M \times q} \tag{22}$$

Based upon the relationship between p_v and V as shown in (20), we can derive the critical number of virtual machines required.

For example, if the parameters take the following values: $M = 1,000$, $m = 40$, $n^e = 1,000$, and $q = 50$.[7] The corresponding cutoff value is $p_{v,cutoff} = 0.2$.

Suppose the average hours during which a real machine is alive is $t = 8$. The average days for a real machine to be at work is $d = 5$. To reach the cutoff $p_{v,cutoff}$, the number of virtual machines that the researcher needs to create is $V_{cutoff} = 295$. The size of the botnet is accordingly $N = 5,250$.

The numerical example suggests that given the parameters, the market profit will be lowered down to zero if the chance for a rental bot to be virtual is 0.2. For a technically-determined effective size of rental $n^e = 1,000$, 295 virtual machines are required. Without virtual machines, the botnet master

[6]In reality, the chance for a botnet master to be detected and arrested is small. Dropping the penalty component of the costs does not damage the model conclusions. Effects of non-zero legal punishment and how legal enforcement can be combined with honeypots to fight botnets, especially when botnets are used to launch attacks with linearly increasing payoffs such as spams are studied in a related work.

[7]The actual values of the parameters can be estimated from empirical studies. The numbers assigned here are for illustrative purposes.

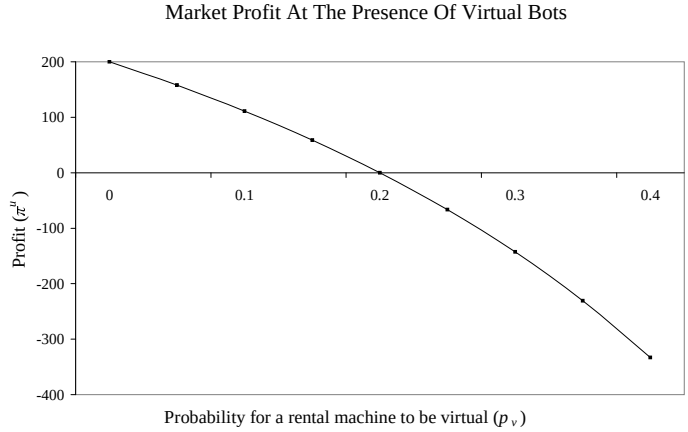

Fig. 4. Botnet market profit decreases with increasing chance of fake rental bots.

only needs to maintain the botnet size at $N = \frac{n^e}{\frac{t}{24} \times \frac{d}{7}} = 4,200$. The interference by virtual machines enlarges the botnet size by the rate of $\frac{1}{1-p_v}$. At the cutoff $p_v = 0.2$, the botnet size is enlarged by 1.25 times.

Note the previous numerical example is based upon the assigned parameter values. If they change, the cutoff probability and the number of virtual machines also change. m and n^e affect p_v negatively, and M and q affect p_v positively. From the perspective of researchers, a negative impact on p_v is favorable since a lower p_v requires fewer virtual machines to be in place. Increasing cost of maintaining channels (higher m)[8] and larger number of machines required to disable the target site (larger n^e) raise the operation burden of the botnet master. By contrast, more payoff for disabling the victim (larger M) and more machines a C&C channel can support (larger q) enhance the motivation for attacks and reduce the operation costs for the botnet master.

We now illustrate graphically how the key variables are related using the same parameter values specified.

First of all, the market profit margin depends on the probability for a rental machine to be virtual p_v. It is interesting to know how this profit margin changes with p_v. Figure 4 illustrates the mathematical relationship

$$\pi^u = 1,000 - 40 \times \frac{1,000}{(1-p_v) \times 50}$$

Secondly, the number of virtual machines (V) varies with the probability for a rental bot to be virtual (p_v). The relationship between V and p_v is

$$V = \frac{1,000 \times p_v}{(1-p_v) \times (1 - p_v \times (1 - \frac{8}{24} \times \frac{5}{7}))}$$

Recall the relationship between p_v and the botnet size N^u, we get the following formula linking the two at the given parameter values:

$$N^u = \frac{1,000}{(1-p_v) \times \frac{8}{24} \times \frac{5}{7}} = \frac{4,200}{1-p_v}$$

The graphical illustration of how V and N^u are related to p_v is given in Figure 5.

The above numerical and graphical illustration show that uncertainty matters given the cutoff probability of fake rental bots $p_{v,cutoff}$. The availability of virtual machines largely reduces economic

[8]Botnet masters may seek for innovation in response to the increased use of honeypots. For example, they may develop cheaper means of C&C (i.e., lower m). According to (21) and (22), profit may increase and the cutoff p_v has to be larger. Cheaper means of C&C is unfavorable innovation concerning fighting attacks. Nevertheless, it does not affect the nature of model conclusions.

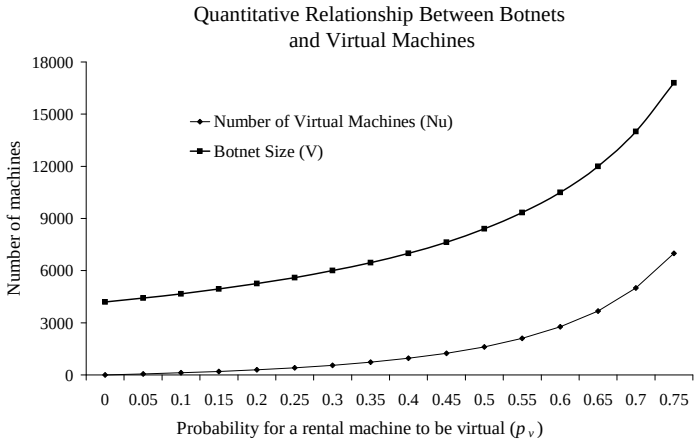

Fig. 5. Optimal botnet size and rental size increase as the chance for a rental bot to be virtual increases.

payoffs for participating in the Internet black market, which reduces the attractiveness of the practice. Making p_v a random number will make the situation even more challenging for botnet masters and attackers.

More likely, the rough ranges of the parameter values are common knowledge. Botnet masters and attackers could also figure out the cutoff value of p_v. By increasing the size of the botnet, they may be able to convert a loss into a profit. To counter react, researchers may have to increase the number of virtual machines, which may further force the botnet masters to expand botnets. Consequently, having p_v fixed may result in an unpleasant situation similar to arms race.

Our proposed strategy becomes much more effective by making p_v uncertain. Without researchers' and defenders' commitment to creating just the "right" number of virtual machines to reach the cutoff $p_{v,cutoff}$, it is difficult if ever possible for the illegal practitioners to guess the actual number of virtual machines. Optimal decisions are therefore no way to make. Since the attacker receives no money if the attack fails, one safe bet may be just to rent as many as possible. The botnet master has to expand the size of botnets as well. The increased costs for both parties reduce the profit margins. If the costs increase by too much, all the profit margins may be disappearing. Note that, at the same time of discouraging botnet masters and attackers from entering the market, the uncertainty helps reduce the operation costs of defenders. They may reduce the number of virtual machines without being aware of. The uncertainty (or randomness of creating virtual machines in some sense) facilitates the implementation of the proposed methodology.

C. Technical Challenges

We further discuss a few feasibility issues such as the magnitude of virtual machines and counter-detection techniques. First of all, the number of virtual machines does not have to be big. According to previous studies, the botnet size ranges from roughly a few hundreds to hundreds of thousands. For example, Dagon et al. establish that botnet sizes may reach 350,000 members [16]. Rajab et al. indicate that the effective sizes[9] of botnets rarely exceed a few thousand bots [1]. A recent study by Rajab et al. revisits the question of botnet size and draws the distinction between footprint (the overall size of the infected population at any point in the lifetime of a botnet) and live population (the number of live bots simultaneously present in the command and control channel). They show that while the footprints of the botnets can grow to several tens of thousands of bots, their effective sizes usually are limited to a few thousands at any given point in their lifetime. For example, botnet footprint sizes can exceed 100,000 infections, their live populations are normally in the range of a few

[9]The effective size of a botnet is the number of bots connected to the IRC channel at a specific time. While the effective size has less impact on long term activities such as executing commands posted as channel topics, it significantly affects the number of minions available to execute timely commands such as DDoS attacks.

thousand bots [4]. The relatively limited size of botnets suggests that it may not be easy to enlarge botnets dramatically and rapidly due to some practical or technological barriers. If the probability for a machine to be virtual in the rental botnet is at a decent level, botnets will be significantly affected. For example, suppose $p_v = 0.1$, then the botnet size has to be 11 percent[10] larger compared with the situation in which virtual machines are not around. The attacker has to rent 11 percent more machines and suffers a 11 percent increase in costs. There is also a 11 percent increase in the costs for the botnet master to maintain more C&C channels and more machines, which can by significant. The contrast between the relative easiness to build virtual machines and the difficulty in enlarging botnets implies the opportunities for our plan to work.

The functioning of honeypots is pivoting on camouflaging fake bots. Indeed, botnets are not equally complicated. They diversify in terms of technological complexity. Botnets can be roughly categorized into three groups, depending on the botnet master's technological proficiency:

- Case I. Low: It should not be a big problem for defenders to make virtual machines to join a botnet.
- Case II. Medium: Botnet masters only check compromised machines at the entry of a bot. If a virtual machine passes this entry test, it will not be evaluated again.
- Case III. Advanced: The most challenging situation is when a sophisticated botnet master sends commands to test machines not only at the entry, but also from time to time. In this case, what requires is some anti-detecting technique or strategy. For example, allowing virtual machines to fulfill some trivial tasks would make virtual machines trustworthy to the botnet master. To follow this "I-fool-you, catch-me-if-you-can" strategy, it is crucial to find ways for virtual machines to judge which orders are innocuous to follow. What technical tools/progresses are necessary to disguise honeypots from being detected is also a promising further research topic.

The dynamic features of botnets also facilitate our method. According to Karasaridis et al. [17], the botnets are very dynamic in nature. Based on long-term monitoring of validated malicious botnets, they estimate that the average bot stays about two to three days on the same botnet controller, switching controller addresses and domains very frequently. A duration of a couple of days makes it harder and less productive to conduct test orders frequently. More likely, botnet masters may only command a newly compromised machine to do a simple task at entry. Botnet masters also steal each others' machines. Honeypots may function equally well if being lost from one botnet to another. Furthermore, newer bots can automatically scan their environment and propagate themselves using vulnerabilities and weak passwords. Generally, the more vulnerabilities a bot can scan and propagate through, the more valuable it becomes to a botnet controller community. Therefore, a virtual machine-created pseudo-bots can propagate by including more virtual machines into a botnet, and enhance the higher weights and the "importance" of the virtual machines to botnet masters.

The botnet controller community features a constant and continuous struggle over who has the most botnets, and the largest amount of "high-quality" infected machines, like university and corporate machines. It may be economically reasonable for a botnet master to create larger botnets. For example, advertising a larger botnet may send a positive signal to potential buyers on Internet underground markets indicating the botnet master is experienced and ought to have good reputation. Operating a larger botnet may also facilitate certain tasks that botnets are for. For example, a larger botnet may be more effective to disable a target by overwhelming it or more spam emails can be sent in a short period of time by having more machines do the job. Since botnet masters have to keep recruiting new machines even they are fully aware of the existence of honeypots, the virtual bots' entry to botnets can never be shut down.

Meanwhile, the size of a botnet is subject to an upper bound, sometime specified by the width of the C&C channel. Therefore, there is a tradeoff between hacking more machines and increasing C&C channels. The more machines hacked, the larger the size of the botnet, and more buffer can be obtained, but more machines require more C&C channels, which increases operating costs of the

[10]The size of the botnet is $\frac{1}{1-0.1} = 1.11$ times of the size in the benchmark case. The increase in size is 11 percent.

botnet and the chance of being detected. The existence of honeypots makes maintaining a botnet more costly and risky since the botnet master may have to increase the size of the botnet to compensate for the uncertain inactive honeypots. One thing to note, instead of increasing the size of the botnet, the botnet master may rather reduce the size of the botnet, and only keep those "safe" and active machines. It is certainly a strategy botnet masters may use, the risk of that is a continuously declining botnet due to the life cycle of a comprised machine. Figuring out the optimal size of botnet given the complicated scenario then becomes mission impossible.

VI. Conclusion and Future Work

Profit-driven botnet attacks impose serious threats to the modern Internet. Given that money is perhaps the single determining force driving the growth in botnet attacks, we propose an interesting economic approach to take away the financial incentives. By introducing the uncertainty level, we make the optimal botnet size infeasible for the botnet operators. As the chance of uncertainty increases, both botnet masters' and attackers' profits can fall dramatically.

The proposed scheme is advantageous versus existing schemes in that it strikes at the root motivation for the botnets themselves, i.e., the profit motivation. Regardless of the type of command and control structure, the sophistication of compromising new hosts, or the creation of new avenues to market botnet services, we believe this paper nicely demonstrates how the application of economic principles can offer significant benefit to combatting botnets.

The paper is the stepping stone of a series of analyses. In a related work, we include non-zero legal punishment into the profit maximization problem and discuss how the coordination of legal engagement and honeypots works to reduce financial incentives of non-DDoS botnet-related cybercrimes whose payoffs are linearly increasing in the use of botnet. Moreover, with varying qualities of botnets and diversified reputation of botnet masters, Internet botnet markets may be more monopolistic competitive or price discriminated. The assumptions of price-taking market participants and a single rental price of bots may be relaxed to study price discrimination, and such modification of the problem setup may result in some interesting results. Legalizing Internet black markets is another attractive and challenging idea. Besides economic factors, technical, social, ethical and legal considerations all play certain roles. A wealth of research can be carried out along this line of thinking.

References

[1] M. A. Rajab, J. Zarfoss, F. Monrose, and A. Terzin, "A multifaceted approach to understanding the botnet phenomenon," in *6th ACM SIGCOMM conference on Internet Measurment, SESSION: Security and Privacy*, 2006, pp. 41–52.

[2] R. Mahajan, S. Bellovin, S. Floyd, J. Ioannidis, V. Paxon, and S. Shenker, "Controlling high bandwidth aggregates in the network," *ACM SIGCOMM Computer Communication Review*, vol. 32, no. 3, pp. 62–73, July 2002.

[3] D. K. Y. Yau, J. C. S. Lui, F. Liang, and Y. Yam, "Defending against distributed denial-of-service attacks with max-min fair server-centric router throttles," *IEEE/ACM Transactions on Networking*, vol. 13, no. 1, pp. 29–42, 2005.

[4] M. A. Rajab, J. Zarfoss, F. Monrose, and A. Terzis, "My botnet is bigger than yours (maybe, better than yours): Why size estimates remain challenging," in *Proceedings of the first conference on First Workshop on Hot Topics in Understanding Botnets*, Cambridge, MA, 2007, p. 5.

[5] K. Park and H. Lee, "On the Effectiveness of Probabilistic Packet Marking for IP Traceback under Denial of Service Attack," in *Proc. of INFOCOM 2001*, 2001, pp. 338–347.

[6] A. Snoeren, C. Partridge, L. Sanchez, C. Jones, F. Tchakountio, S. Kent, and W. Strayer, "Hash-Based IP Traceback," in *Proc. of SIGCOMM*, 2001, pp. 3–14.

[7] S. Savage, D. Wetherall, A. P. Karlin, and T. Anderson, "Practical Network Support for (IP) Traceback," in *Proc. of SIGCOMM*, 2000, pp. 295–306.

[8] J. Xu and W. Lee, "Sustaining availability of web services under distributed denial of service attacks," *Transactions on Computers*, vol. 52, no. 2, pp. 195–208, Feb 2003.

[9] C. Jin, H. Wang, and K. Shin, "Hop-Count Filtering: An Effective Defense Against Spoofed DoS Traffic," in *Proc. of the 10th ACM Conference on Computer and Communications Security*, 2003, pp. 30–41.

[10] S. Jin and D. Yeung, "A Covariance Analysis Model for DDoS Attack Detection," in *Proc. of the IEEE International Conference on Communications (ICC)*, vol. 4, June 2004, pp. 1882–1886.

[11] J. Franklin and A. Perrig, “An inquiry into the nature and causes of the wealth of internet miscreants,” in *Proceedings of the 14th ACM conference on Computer and Communications Security, SESSION: Internet Security*, Alexandria, Virginia, 2007, pp. 375–388.
[12] R. Ford and S. Gordon, “Cent, five cent, ten cent, dollar: Hitting botnets where it really hurts,” in *New Security Paradigms Workshop*, 2006, pp. 3–10.
[13] P. Bcher, T. Holz, M. Ktter, and G. Wicherski, “Know your enenmy: Tracking botnets.” *The Honeynet Project & Research Alliance*, March 2005.
[14] “Computer scientist fights threat of botnets.” *ScienceDaily*, Nov. 10 2007. [Online]. Available: http://www.sciencedaily.com/releases/2007/11/071108141303.htm
[15] “Worldwide infrastructure security report vol.ii (2006),” *ARBOR NETWORK*. [Online]. Available: http://www.arbornetworks.com/report
[16] D. Dagon, C. Zou, and W. Lee, “Modeling botnet propagation using time zones,” in *Proceedings of the 13th Annual Network and Distributed System Security Symposium (NDSS’06)*, Feb. 2006.
[17] A. Karasaridis, B. Rexroad, and D. Hoeflin, “Wide-scale botnet detection and charaterization,” in *USENIX Workshop on Hot Topics in Understanding Botnets (HotBots’07)*, 2007.