Evaluation of the Severity of DoS Attacks on Computer Networks

Amar Aissani
Department of Computer Sciences
USTHB
Algiers, Algeria
aaissani@usthb.dz

Maroua Yaiche Achour
Department of Computer Sciences
USTHB
Algiers, Algeria
yaichemaroua@gmail.com

***Abstract*—In this paper, we derive stochastic recursive equations describing the evolution of a computer network under Denial-Of-Service (DoS) attacks (flooding attacks). The queue has several input processes, (i) the regular one (control packet flow and background or non control applications), which describes the input under network normal status; and (ii) the attack packet flow. We concentrate on some particular security measures, namely, the load or the loss probability. The load is strongly connected with the stability, which is understood as the convergence of the underlying stochastic process to a unique stationary ergodic regime. Loss of packets can occur although the system is stable. There are no specific requirements regarding statistical assumptions for establishing such equations. However, in order to derive stability condition and stationary performance measures we will need assumptions like stationarity and ergodicity (the independence is not required). Finally, we provide some numerical illustrations showing the effect of parameters on security measures and thus the severity of such attacks.**

***Keywords- Security; Queueing; Reliability; DoS Attack*.**

I. INTRODUCTION

Most of information systems are exposed to Denial-Of-Service (DoS) attacks, which is a one of the various forms of security threat of computer networks [1, 3, 9, 11]. The aim of DoS attacks is to exhaust a resource in the target system, that can be anything related to network computing and service performance (link bandwith, TCP connection buffers, application/service buffer, CPU times). Such attacks can also exploit a specific vulnerability in order to reduce or completely subvert the availability of the service provided. A common strategy used by an intruder to cause a DoS attack on a given target is to flood it with a continuous stream of packets that exhausts its connectivity. DoS attacks that use this kind of strategy are called brute-force attacks. Distributed Denial-of-service (DDoS) attacks are simply DoS attacks performed by multiple agents simultaneously. Many efforts have been made, in parallel with the evolution of DoS attacks, in the field of prevention and detection in networking security. Several countermeasures have been proposed, and can be roughly categorized as host-based systems and network based systems. Host-based systems are deployed on end-hosts and typically use firewall, intrusion detection systems (IDS) and/or balance the load among servers. This technique can help to protect the server, but not the legitimate access to the server because high-volume traffic may congest the incoming link to the server. Network-based systems are deployed inside networks (on routers) and fall into two categories: (1) Detection/identification mechanisms using signal processing and or statistical techniques; (2) Defense techniques using traffic control mechanisms such as egress or ingress filtering, route-based packet filtering disabling unusued services, and honeypots (see [1, 3, 9, 11]).

The Internet traffic is a complex stochastic process and there are several studies trying to describe a mathematical framework to model the behavior of these kinds of attacks. The interest of such models is to investigate how the attacks and other security anomalies affect the performance of the Network. It is difficult for legitimate users to launch real DoS attacks against the prototype of network to measure performance, since the attacks are themselves classified as cybercrime against the law.

We consider the model of NBCS (Network-Based Control System) described in [9] in which packets moving from one site to another have to access shared resources (communication links and network equipment). For each router in the path between a plant and a controller, the mechanism governing packet transmission can be abstracted by a queue with FIFO (First-in-First-Out) discipline of service. Packets arrive randomly at a router and can be modeled by some stochastic processes. If a packet finds the router CPU idle, it will be immediately served for a random amount of time. If the router CPU is busy, the packet will be in the queue to wait. When a queue with a finite size is full, the newly arrived packet enters “orbit” (a sort of queue) and repeat his attempt until he finds a place in the queue. Note that in the original version of [9] it is assumed that such a packet is dropped.

The routers in the path handle not only the NBCS packets flow, but also other traffic (non-control applications and flows of other NBCS systems). So, the model assumes several input processes: the first one is the regular (or

ISBN: 978-1-61208-198-4

legitimate) NBCS packet flow, the second one is the regular background traffic and the other is the DoS attack flow.

So, we consider that the server (i.e., the router CPU) has several inputs. The regular input describes the packet flow under normal network status. We separate the basic flow of control applications, which is assumed to be a Poisson process with rate λ_1 packets/sec. and the background flow (non control application/other packet flows), a Poisson process with rate ψ packets/sec. The attack traffic is modeled by a Poisson process with rate ϕ packets/sec. The Poisson assumption is conforming to some experimental studies in traffic studies [10] and also to some statistics about Denial of Service activities [11].

We denote by $\alpha = \{\sigma_n^1\}$, $\beta = \{\sigma_n^2\}$, $\delta = \{\sigma_n^3\}$ the sequences of service times for the control flow, the background flow and the attack flow respectively. We assume that these sequences form stationary ergodic sequences (or which is equivalent metrically transitive) without the usual independence assumption. The stationarity is understood here in the strict sense. The inter-retrial times are independent identically distributed random variables with common exponential distribution function with parameter $\nu > 0$.

In order to take into account the implemented defense mechanisms (firewall for example), we introduce a filtering parameter p, $0 < p < 1$. So, when a regular packet finds the service blocked, it is dropped with probability p. With probability $1 - p$ it joins the service area (if it is not full) or a retrial group (also called an orbit). From the orbit it repeats his attempts at rate $\nu > 0$ until it gets service.

When a packet of the attack traffic finds the service blocked, it is dropped with probability q. It joins service area or orbit with probability $1 - q$. In orbit, the attack packet evolves as a regular one: it adjusts its strategy and merges into regular packets. So, q can be seen as the filtering probability of an attack packet. For coherence, one can assume that $1 > q > p > 0$.

The paper is organized as follows. In Section II, we model the behavior of the network with a stochastic recursive sequence (SRS) (a generalization of the embedded Markov chain).

This representation gives an algorithm (Section III) for the simulation of sample paths of the underlying SRS and the statistical estimation of several security measures. We derive also the stability condition, which insures the existence of a unique stationary regime. There are no specific requirements regarding statistical assumptions for establishing such equations. However, in order to derive stability condition and stationary performance measures we will need assumptions like stationarity and ergodicity (the independence is not required).

In Section IV, we adjust the model by considering some other types of attacks which conduct to the interruptions of service.

Finally, we provide (Section V) some numerical illustrations showing the effect of parameters on security measures and thus the severity of such attacks.

II. A STOCHASTIC EQUATION FOR SIMULATING NETWORK SAMPLE PATHS

In a first part, we assume that the buffer $K = 0$. Let $\{N(t), t \geq 0\}$ be the number of packets in the orbiting queue at time t. It represents a stochastic process on the discrete space of natural integers. Let $C(t)$ be another 3-valued random process describing the server status: $C(t) = 0$ if the server is free at time t; $C(t) = i$ if the server is busy by service of a certain packet of type i at time t, $i = 1,2,3$.

We consider the process $\{N_n\}$ embedded immediately after service times γ_n (i.e., $N_n = N(\gamma_n + 0)$. Denote by $X_n = (C_n, N_n), n \geq 1$ the sequence of successive states of the system at these epochs where $C_n = C(\gamma_n + 0)$. Observe that if the sequence α, β, δ are independent and identically distributed, then the sequence $\{X_n, n \geq 1\}$ forms a Markov chain (in the usual sense) defined on the state space $S = \{1,2\} \otimes IN$ and the ergodicity condition can be derived using the Foster-Moustafa-Tweedie criterion [5].

We next show that the process $\{N_n\}$ is a stochastic Recursive Sequence in the sense of Borovkov [4, 5]. Recall that a process $\{N_n\}$ is called a SRS with driver $\{\{\xi_n\}, f\}$, if for some function f it satisfies the equation $N_{n+1} = f(N_n, \xi_n), \forall n \geq 0$ where the driving sequence ξ_n is a stationary ergodic stochastic process.

It is well known that for the classical FIFO queue, idle (respectively, busy) server period coincides with the system idle (respectively, busy) period. It is not the case for retrial queues where in the system busy period the systems evolves as an alternating sequence of idle periods and busy periods of the server.

The situation is slightly different in the case of finite buffer. Let τ_n be the n th idle server period, i.e., the time between the end of the $n-1$th service till the beginning of the nth service. The distribution of τ_n is determined by the competition between inter arrival times and inter retrial times, which event occurs first. This idle period ends when either there is an external arrival (regular or attacker) or when a call from orbit tries to retry. Under our assumptions, τ_n is exponentially distributed with parameter $\lambda+\nu$ where $\lambda=(\lambda_1+\psi)(1-p)+\phi(1-q)$ (constant retrial policy) and $\lambda+\nu N_n$ (linear retrial policy). The conditional probability, given $\Im(\gamma_n)$ (the sigma algebra generated by events describing state of the system up to time γ_n), that the first event to occur after the $n-1$th service ends (and after the served packet has left the system) is an external arrival (regular or attacker), equals $\frac{\lambda}{\lambda+\nu}$ (respectively, $\frac{\lambda}{\lambda+\nu N_n}$). The conditional probability, given $\Im(\gamma_n)$ that the first event to occur after the $n-1$th service ends (and after the served packet has left the system) is a retrial, equals $\frac{\nu}{\lambda+\nu N_n}$ (respectively, $\frac{\nu N_n}{\lambda+\nu N_n}$).

Consider the following two Pseudo Random Generators, $u_n^1=\{u_n^1, n=0,1,...\}$ and $u_n^2=\{u_n^2, n=0,1,...\}$ They are described in fact by two sequences of random variables distributed uniformly on $[0,1]$ mutually independent, and independent of the sequences α,β,δ. Let $\chi(A)$ be the characteristic function of the event A : $\chi(A)=1$, if A has occurred, $\chi(A)=0$ otherwise. We will need also a mean to generate the input Poisson random processes.

For the formal description below we introduce an application $\Pi : IR^+\times[0,1]\rightarrow IN$ defined by

$$\Pi(t,x)=\inf\left\{n\in IN:\sum_{k=0}^{n}\frac{t^k e^{-t}}{k!}\geq x\right\}.$$

Thus, $\Pi(t,u_n^1)$ implements a Random Poisson Generator for the sequences the instant of primary arrival packets (regular or attacker) or secondary (retrials) [2, 12]. The second sequence $u_n^2=\{u_n^2, n=0,1,...\}$ will be used to generate which event has occurred. Formally, we can consider the following events G_n, H_n, S_n, R_n such that

$$G_n=\left\{0\leq u_n^2\leq\frac{\lambda_1(1-p)}{\lambda+\nu N_n}\right\},$$

$$H_n=\left\{\frac{\lambda_1(1-p)}{\lambda+\nu N_n}\leq u_n^2\leq\frac{(\lambda_1+\psi)(1-p)}{\lambda+\nu N_n}\right\},$$

$$S_n=\left\{\frac{(\lambda_1+\psi)(1-p)}{\lambda+\nu N_n}\leq u_n^2\leq\frac{(\lambda_1+\psi)(1-p)+\phi(1-q)}{\lambda+\nu N_n}\right\},$$

$$R_n=\left\{\frac{(\lambda_1+\psi)(1-p)+\phi(1-q)}{\lambda+\nu N_n}\leq u_n^2\leq 1\right\}.$$

According to the relations above, we have the following stochastic equation

$$N_{n+1}=\max(0,N_n+\xi_n)=(N_n+\xi_n)^+ \qquad (2.1)$$

where $\xi_n=h(N_n,\sigma_n^1,\sigma_n^2,\sigma_n^3,u_n^1,u_n^2)$ is given by

$$\xi_n=\chi(G_n)\Pi(\lambda\sigma_n^1,u_n^1)+\chi(H_n)\Pi(\lambda\sigma_n^2,u_n^1)+\chi(S_n)$$
$$\Pi(\lambda\sigma_n^3,u_n^1)+\chi(R_n)(\Pi(\lambda\sigma,u_n^1)-1) \qquad (2.2)$$

for linear retrials. In the case of constant retrial rate, the equations (1)-(2) remains valid except that the term νN_n is replaced by ν in the definition of the events G_n, H_n, S_n, R_n.

Formula (2.1) is just an arithmetical count of the number of customers in the system at a given time. The number of customers N_{n+1} in orbit after the n+1 service equal the number of customers N_n at the previous nth service time plus the variable ξ_n. This variable counts the difference between the number of arrivals and departures during the period $[\gamma_n,\gamma_{n+1}]$ (interval between the two successive departures nth and n+1th. The operator max stay here, since the variable ξ_n cannot be negative.

The first term in formula (2.2) counts the number of packets of the basic flow (control applications), which have been accepted by the filter (with probability 1-p), i.e., when the event G_n occurs; the second term counts the number of packets of the background flow, which have been accepted by the filter(also with probability 1-p) (when H_n occurs); the third term counts the number of packets of the attack flow, which have been accepted by the filter (with probability 1-q), i.e., when the event S_n occurs); finally, the forth term counts the number of packets which has been served and exit the systems (when the event R_n occurs).

In both cases, the process

$$\xi_n = \left(\sigma_n^1, \sigma_n^2, \sigma_n^3, u_k^1, u_k^2, k \le n\right)$$

is the driving sequence for the SRS taking values in $\Theta = IR^+ \otimes IR^+ \otimes IR^+ \otimes [0,1] \otimes [0,1]$ and is assumed stationary ergodic.

In (2.1)-(2.2) it is assumed that the buffer K=0, while the original model [9] assume a finite buffer of capacity $K \ge 1$. In this case, a retrial occurs if a packet finds the buffer full. So, the stochastic equation (1) needs to be refined.

Let M_n be the number of packets in the buffer at time γ_n , and then the basic process is now described by the two-dimensional process $Y_n = (M_n, N_n)$. In this case, the SRS has the following form

If $M_n < K, \xi_n \le K - M_n$, then

$$\left(M_{n+1}, N_{n+1}\right) = \left(M_n + \xi_n, N_n\right) \qquad (3.1)$$

If $M_n = K, \xi_n > K - M_n$, then

$$\left(M_{n+1}, N_{n+1}\right) = \left(K, N_n + \xi_n - (K - M_n)\right). \qquad (3.2)$$

The process Y_n describe the behavior of the network when $K \ge 1$, but finite. The above SRS shows how to compute Y_n. We distinguish two cases. Formula (3.1) corresponds to the case when the buffer is not full and formula (3.2) to the case when the buffer is full, i.e., $M_n = K$.

III. SIMULATION ALGORITHM AND IT'S PERORMANCE

The representation under the form of SRS is particularly adapted to a discrete-event simulation of the network under DoS attacks.

```
Set $N_0 = 0$ (initialization)
Repeat
$u^1 \leftarrow$ Random; {generation of $u^1$ and arrival event}
$u^2 \leftarrow$ Random; {generation of $u^2$ and the type of the arrival packet}
$u^3 \leftarrow$ Random; {generation of $u^3$ and the service time random variable according to the given probability distribution}
Poisson variables are generated using any algorithm for Poisson process.
For all n,
Computation of $\xi_n = f(u^1, u^2, \sigma^i), i = 1,2,3$ by formula (2.2) or (2.3).
Computation of $N_{n+1} = \max(0, N_n + \xi_n) = (N_n + \xi_n)^+$.
Computation of the state at time $n+1$ given $N_n$ and $\xi_n$.
End for
Until $n < T$ ($T =$ end of simulation).
```

Based on the SRS formulation,, the above algorithm gives directly a sample of the steady state distribution provided the network is stable (see Fig. 1 in Section V). Next, we can compute statistical estimate of any security metric directly from sample paths (Delay, Loss probability, Load…

In fact, the algorithm simulate the physical operation of the system, arriving customers (regulars or attackers), retrial requests, filtering actions and service of customers. It handle these different actions by the next-event incrementing procedure, which differs from the fixed-time incrementing in that the master clock is incremented by a variable amount rather than by a fixed amount of time.

Conceptually, the next-event incrementing procedure is to keep the simulated system running without interruption until an event occurs, at which point the algorithm pauses momentarily to record the change in the system. To implement this idea, the algorithm actually proceeds by keeping track of when the next few simulated events are scheduled to occur, jumping in simulated time to the first of these events, and updating the system. The cycle ends at time T and it is repeated as many time as desired, say N times.

We can see that the running time is $N \times T$ unit of times.

It is important to estimate the quality of the estimation of mean performance measures. The precision is $\frac{1}{\sqrt{N}}$ by the law of large numbers [2, 12].

We can prove [3] that the network is stable if $\rho < 1$, ,where

$$\rho = \frac{\lambda + \nu}{\nu} \times$$
$$\times \left[\lambda_1^2 (1-p) E(\sigma^1) + \psi^2 (1-p) E(\sigma^2) + \phi^2 (1-q) E(\sigma^3)\right]$$

in the case of constant retrials and

$$\rho = \lambda \times \left[\lambda_1^2 (1-p)E(\sigma^1) + \psi^2 (1-p)E(\sigma^2) + \phi^2 (1-q)E(\sigma^3)\right]$$

in the case of linear retrials.

Here, the stability is understood as the strong coupling convergence [3-6, 8]) to a unique stationary regime. This condition is also a condition of convergence of the algorithm of Section III. The formula for ρ depends on the retrial policy (constant or linear). This quantity represents the traffic intensity and also the load, which will serve here as a security measures for detecting the status of the network: normal or under attack. The network is under attack if the value of ρ crosses a given threshold.

Another security metric, which is not considered here is the Loss probability, when $K \geq 1$. In this case, we can detect a DoS attack if the loss probability (depending on K) is large. So, the security status is defined by a threshold $\varepsilon = \varepsilon(K) > 0$ small enough. The network is under DoS attack if the loss probability is $\geq 1 - \varepsilon$. An application of such security measure can be found in the work [1] with a different model.

IV. Attacks on the availability

We have up now considered DoS attacks, more precisely flooding attacks which aim to saturate the system by sending many requests of service. But, there is another type of attacks which exploit a specific vulnerability in order to reduce or completely subvert the availability of the service provided (interruption of service). In this section we take into account such attacks in the previous model by introducing a new parameter θ, the rate of such attacks. So, we assume that the service becomes unavailable for a random restoration period of time. Such attacks occur according to a Poisson process with rate θ. We denote by $r^{(n)} = \left\{r_i^{(n)}, i = 1,2,...\right\}$ the sequence of "renewal" (restoration to the as-good-as new state) times, which is assumed again stationary ergodic and independent of the other sequences of parametric random variables. In this case, we have again the representation of the basic process under the form of SRS (2.1). We need only to take into account delay due to renewal times and the incrementation of DoS attacks during such periods :

$$\sum_{i=1}^{\Pi(\theta\sigma, u_n)} \Pi\left(\omega r_i^{(n)}, u_i^{(n)}\right), \qquad (4.1)$$

where $\omega = \lambda_1(1-p) + \phi(1-p)$ or $\psi(1-q)$ according to the case which occurs.

Formula (4.1) indicates that the full service of a given customer (if it is not lost) is the pure service plus the cumulated duration of all interruptions occurring during this service.

The model can also take into account other types of interruptions, for example due to software or hardware failures.

V. Numerical illustrations

In this section, we show the effect of DoS attack on some security measures. First, Fig. 1 shows some sample paths of the stochastic process $\{N_n\}$ and the simulation algorithm of Section III.

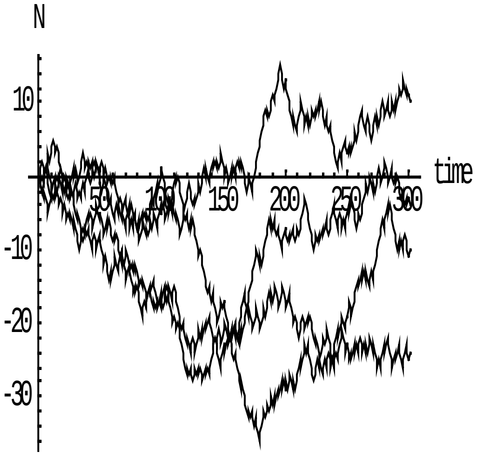

Figure. 1. Sample paths of the process $\{N_n\}$.

From these sample paths, we can compute the sample average of any security measure (for example, loss probability, etc.), which is an estimation of the true security measure. This estimation is unbiased, consistent and efficient (in the statistical sense) [2 , 12].

Fig. 2 compares the evolution of the load ρ as a function of the attack parameter ϕ. We neglect the background flow ($\psi = 0$) and fix some parameters. We assume the mean service times are identical for all types of requests and set $\lambda_1 = 10/\sec$. We observe that the load increases with the severity of the attack (when the attack rate increases) for a fixed value of the retrial rate. The load decreases with increasing of the retrial rate.

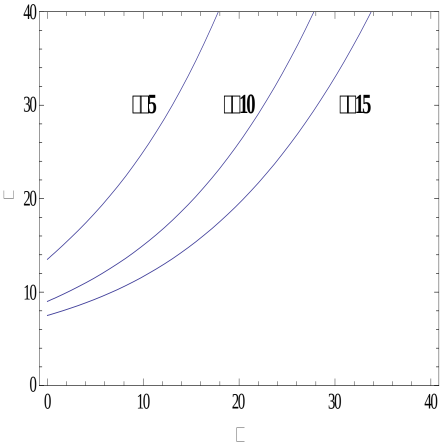

Figure 2. Effect of DoS attack rate ϕ on the load ρ for different values of the retrial rate ν.

Fig. 3 is another view of this observation. It shows the effect of the retrial rate ν on the load for different values of the attack parameter ϕ. We consider three cases $\phi = 0$ (under normal network status), $\phi = 20$ or $\phi = 40$ (under DoS attack).

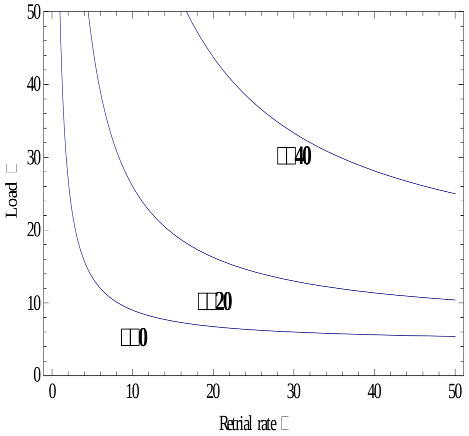

Figure 3. Effect of retrial rate ν on the Load ρ for Different Values of ϕ.

We observe that the load decreases with increasing of the retrial rate ν. The load increases with the severity of attack.

VI. CONCLUSION

In this paper, we have provided an extension of the model of [1], which takes into account the possibility of retrials of packets and also the existence of defense mechanisms (firewalls). The evolution of the system is described by a stochastic recursive sequence, which provides a practical mean to simulate sample paths of the underlying process and estimate several security measures. Such a security measure serves as an indicator of intrusion. The stability condition is obtained under quite general assumptions about service process (stationarity in the strict sense and ergoditicity). The model can be refined by taking into account some other phenomena and also the comparison with real data. Although it is a practice to assume Poisson arrivals in a first study, it will be interesting to consider the case of non Poisson arrivals as reported in some experimental studies.

ACKNOWLEDGMENT

We are very grateful to the reviewers and the editor for their helpful comments and suggestions. This research was supported by the Ministry of High Education and Scientific Research.

REFERENCES

[1] A. Aissani, "Queueing Analysis for Network Under DoS Attacks," In Lecture Notes in Theoretical Computer Science, O. Gervasi and al., Ed. Springer Heidelberg, Berlin, vol. 5073, Part II, pp. 500-513, 2008.

[2] A. Aissani, Modeling and Simulation. 2^{nd} Edition, Office of University Publications (OPU), Algiers, 2010, (in French).

[3] A. Aissani, "Stochastic Analysis of a Network under DoS Attacks," unpublished.

[4] E. Altman, "On the Stability of Retrial Queues," Queueing Sys. vol. 26, no 3-4, pp. 343-363, 1997.

[5] A. Borovkov, Ergodicity and Stability of Stochastic Processes, Wiley, New York, 1998.

[6] A. Borovkov and S.G. Foss, "Stochastic Recursive Sequences and their Generalizations," Siberian Advances in Mathematics. vol. 2, pp. 16-81, 1992.

[7] G. Falin and J.G.C. Templeton, Retrial Queues, Chapman and Hill, New Jersey, 1997.

[8] T. Kernane and A. Aissani, "Stability of Retrial Queues with Versatile Policy," Appl. Math. & Stoch. Analysis. Article ID 54359, pp. 1-16, 2006.

[9] M. Long, J. Chawan-Hwa and J. Hung, "Denial of Service Attacks on Network-Based Control Systems: Impact and Mitigation," IEEE. Transactions on Industrial Informatics, vol. 1, no 2, pp. 85-96, 2005.

[10] R. Martin , Basic Traffic Analysis, Prentice-Hall, New Jersey, 1993.

[11] D. Volker, G., Savage, "Inferring Internet Denial-of-Service Activity," Proc. UNESIX Security Symposium, pp. 9-22, IEEE Press, New York, 2001.

[12] P. Tavel, Modeling and Simulation Design. AK Peters Ltd., Natick, MA , 2007.