A Principle of a Data Synthesizer for Performance Test of Anti-DDOS Flood Attacks

Ming Li [1] and Wei Zhao [2]

***Abstract*—** Distributed denial-of-service (DDOS) flood attacks remain a big issue in network security. Real events of DDOS flood attacks show that an attacked site (e.g., server) usually may not be overwhelmed immediately at the moment attack packets arrive at that site but sometime late. Therefore, a site has a performance to resist DDOS flood attacks. To test such a performance, data synthesizer is desired. This paper introduces a principle to synthesize packet series according to a given value of the Hurst parameter for performance test of anti-DDOS flood attacks.

***Keywords*—** Long-range dependent traffic, testing, distributed denial-of-service flood attacks, synthesizing traffic, intrusion tolerance.

I. Introduction

ATTACKS may take the advantages of the principle of the Internet (such as openness, resource sharing, assessability, and so forth) to launch DDOS flood attacks, see e.g. [1-6]. Though there are systems and approaches for detecting DDOS flood attacks, see e.g. [7-14], things relating to performance test of anti-DDOS flood attacks are rarely reported.

Briefly speaking, a DDOS attacker sends flood packets on a victim such that the attacked site denies services it normally offers or its performance significantly degrades [12-18]. The analysis of real attack events shows that an attacked site generally may not be overwhelmed immediately at the moment of t_0, where t_0 is the start time at which the site is attacked under DDOS flood attacks [17]. Let the attacked site be overwhelmed at time t_1. Then, we call the time interval $(t_0, t_1) \triangleq T_a$ transition process time of attack [31]. The parameter T_a is a measure to reflect the ability of a site to resist DDOS flood attacks. In other words, T_a characterizes a performance of anti-DDOS flood attacks. The larger the T_a the stronger the ability of a site to resist DDOS flood attacks. Thus, it is worth studying performance test of anti-DDOS flood attacks because such a test may provide useful information to evaluate that performance of the protected site under current infrastructure of network and technologies used in that site. Fig. 1 illustrates a performance test scheme of anti-DDOS flood attacks, where test data generator provides simulated DDOS flood packet series while T_a recorder records the performance T_a.

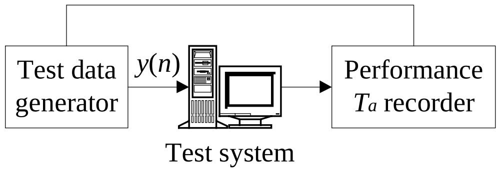

Fig. 1. Test diagram.

From a view of instrumentations, test data generator is a key part in the testing scheme indicated in Fig. 1. This paper aims at introducing a principle regarding test data generator. To this end, Section 2 explains the research background. Section 3 discusses simulation of packet series for a given value of the Hurst parameter (H for short). A case study is given in Section 4. Conclusion and discussion are in Section 5.

II. Research Background

A DDOS flood attacker may coordinate a vast number of Internet hosts distributed all over the world to launch attack packets upon a target site as indicated in Fig. 2, where $x(t)$ stands for aggregated normal traffic, $a(t)$ for aggregated attack traffic and $y(t)$ aggregated abnormal traffic, which is the actual traffic at the target system during attack transition process.

Though modeling abnormal traffic under DDOS flood attacks is rarely reported, we know that abnormal traffic is at unusually high rate [17]. In this regard, our previous work [16] exhibits the change trend of H of traffic time series under DDOS flood attacks. Hence, this paper introduces a flexible data generation for synthesizing abnormal traffic according to a given H value.

Manuscript received March 17, 2007; Revised version received Sept. 29, 2007. This work was supported in part by the National Natural Science Foundation of China under the project grant number 60573125. Wei Zhao's work was also partially supported by the NSF (USA) under Contracts 0808419, 0324988, 0721571, and 0329181. Any opinions, findings, conclusions, and/or recommendations in this paper, either expressed or implied, are those of the authors and do not necessarily reflect the views of the agencies listed above.

[1] Ming Li (corresponding author) is with the School of Information Science & Technology, East China Normal University, No. 500, Dong-Chuan Road, Shanghai 200241, PR. China. (Tel.: +86-21-5434 5193; fax: +86-21-5434 5119; e-mails: ming_lihk@yahoo.com, mli@ee.ecnu.edu.cn).

[2] Wei Zhao is with Rensselaer Polytechnic Institute, 110 Eighth Street, 1C05 Science, Troy, NY 12180-3590, USA. (e-mail: zhaow3@rpi.edu).

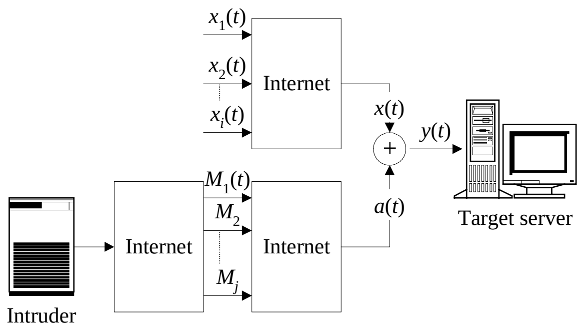

Fig. 2. DDOS flood attack scheme.

It is noted that performance test of anti-DDOS flood attacks is different from normal workload test. For workload test, one needs simulation of normal traffic according to a certain traffic model which obviously differs from abnormal traffic required by the performance test of anti-DDOS flood attacks.

III. Packet Series Synthesizer

A purely random data series for a given value of H, say $G(n)$, cannot be directly used for testing because testing needs packet series. Thus, the discussed test data generator consists of a generator of random data and a packet shaper as shown in Fig. 3. The functionality of random data generator is to synthesize a data series according to a given H value while packet shaper to transform a pure data series G to a packet series y such that it can be accepted by a system being tested. In this way, the output y of packet shaper has a given value of H.

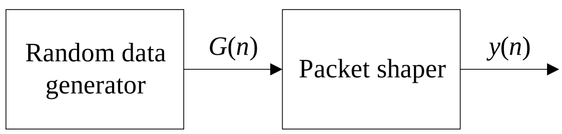

Fig. 3. Diagram of test data generator.

For random data generation, we adopt the method discussed in [19], where white noise is a building block.

As known, ideal white noise is rooted at Brownian motion. Thus, by coloring white noise, one can easily synthesize a series the H value of which is pre-required [19].

Let $w_p(t)$, $W_p(\omega)$ and $S_p(\omega)$ be white noise function, its spectrum and power spectrum, respectively. Then,

$$W_p(\omega) = \mathrm{F}[w_p(t)] = \int_{-\infty}^{\infty} w_p(t)e^{-j\omega t}dt, \tag{1}$$

$$S_p(\omega) = W_p W_p^* = \text{Constant}, -\infty < \omega < \infty, \tag{2}$$

where W_p^* represents the complex conjugation of W_p and F the operator of Fourier transform.

Equation (2) shows that $w_p(t)$ has infinite bandwidth. Practically, however, the bandwidth of a data series is finite as the bandwidth of a system is limited. Therefore, we consider band-limited white noise in our work.

Let $w(t)$, $W(\omega)$ and $S(\omega)$ be the band-limited white noise function, its spectrum and power spectrum, respectively. Then,

$$W(\omega) = \mathrm{F}[w(t)] = \int_{-\infty}^{\infty} w(t)e^{-j\omega t}dt, \tag{3}$$

$$S(\omega) = \begin{cases} W(\omega)W^*(\omega) = \text{Constant}, & |\omega| \le B_c \\ 0, \text{ otherwise} \end{cases}, \tag{4}$$

where B_c is the bandwidth of $w(t)$.

Let ϕ be a real random function with arbitrary distribution. Let

$$W(\omega) = \begin{cases} e^{j\phi(\omega)}, & |\omega| \le B_c \\ 0, \text{ otherwise} \end{cases} \tag{5}$$

be a Fourier transform of band-limited unit white noise. Then,

$$S(\omega) = \begin{cases} W(\omega)W^*(\omega) = 1, & |\omega| \le B_c \\ 0, \text{ otherwise} \end{cases}. \tag{6}$$

Therefore, band-limited unit white noise in time is given by

$$w(t) = \mathrm{F}^{-1}[W(\omega)], \tag{7}$$

where F^{-1} is the inverse of F. In practice, Fourier transform and its inverse are done by a fast Fourier transform (FFT) algorithm [20]. With FFT, (7) becomes

$$w(n) = \mathrm{IFFT}[W(\omega)], \tag{8}$$

where IFFT represents the inverse of FFT. The following demonstrates a synthesis of band-limited white noise.

Suppose the synthesized $w(n)$ is of 2^{12} length and $\varphi(\omega)$ is given by

$$\varphi = [1 - rnd(0,\ 1)]^{-2\pi}, \tag{9}$$

where $rnd(0, 1)$ is uniformly distributed number within (0, 1). The random function φ in (9) is heavy-tailed [21,22]. Fig. 4 indicates φ and its histogram. Fig. 5 shows the generated band-limited white noise $w(n)$ and its power spectrum.

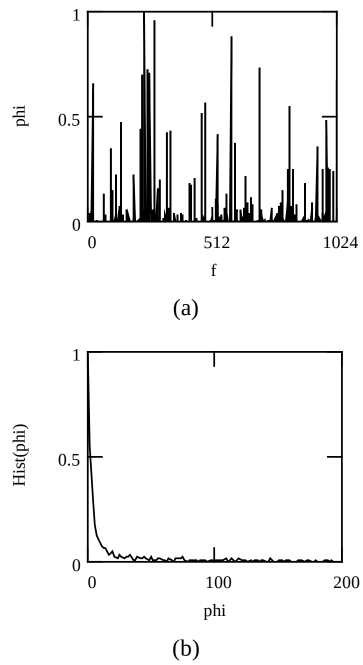

Fig. 4. Generated random phase $\varphi(n)$. (a). Generated random phase. (b) Histogram of random phase.

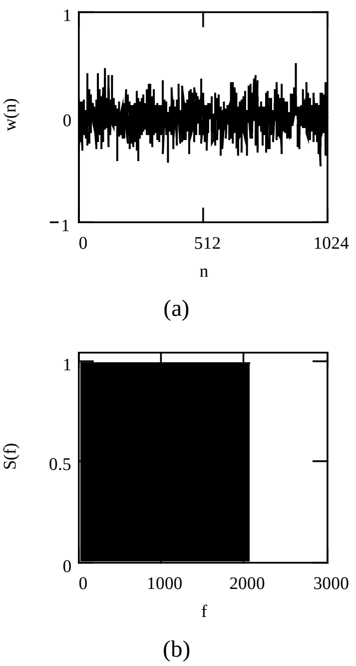

Fig. 5. Illustration of white noise generation. (a) Band-limited white noise $w(n)$. (b) Power spectrum of $w(n)$.

All figures are plotted in the normalized case. In Fig. 4 (a) and Fig. 5 (a), only 1024 points of data are used to illustrate the figures clearly.

Technically, packets described by $y(n)$ should be consistent with the protocol stack used in a system being tested. Since the discussed test scheme focuses on H value assumption, packet shaper just transforms G to a packet series that can be accepted by link layer protocol. Suppose the link layer of a system being tested is with Ethernet protocol [23]. Then, the transformation from $G(n)$ to $y(n)$ can be described as follows:

$$y(n) = <dst, src, len, kG(n)>$$

where *dst* is the destination address of packet $y(n)$ (i.e., the MAC address of system being tested), *src* is the source address that can be any six octets, *len* is the length of $kG(n)$, and k is the calibration coefficient such that $kG(n) \le$ the Maximum of Transmission Unit (MTU) of the tested link.

IV. A Case Study

A random data series can be synthesized according to a given autocorrelation function (ACF) such that the ACF of the synthesized series equals to the given ACF [19,27].

In the sense of approximation, ACF of a traffic series can be described by the ACF of fractional Gaussian noise (FGN) [24-26,28-30]. Let $g(k)$ be normalized ACF of FGN in the discrete case. Then,

$$g(k) \triangleq g(k; H) = 0.5[(|k| + 1)^{2H} - 2|k|^{2H} + (|k| - 1)^{2H}], \quad (10)$$

where $H \in (0.5, 1)$ is the Hurst parameter, which is a measure to characterize the long-range dependence and self-similarity for standard FGN [16,26,28,30]. For a given value of H, therefore, one may synthesize its corresponding series $G(n)$ by the following expression.

$$G(n) = w * \mathrm{IFFT}\{[\mathrm{FFT}(g(k; H))]^{0.5}\} \quad (11)$$

where $*$ stands for convolution (see Appendix). Fig. 7 shows a case with $H = 0.83$. With

$$y(n) = <dst, src, len, kG(n)>,$$

therefore, we have a packet series for a given $H = 0.83$.

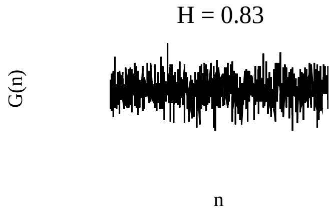

Fig. 7. Synthesized series according to a given value of the Hurst parameter.

V. Conclusion and Discussion

From a view of instrumentations in techniques, a key issue for performance test of anti-DDOS flood attacks is to synthesize attack packet series according to a given value of the Hurst parameter. We note that the issue of traffic model, such as ACF, under DDOS flood attacks remain challenge [16], though there are advances in modeling attack-free traffic, see e.g. [28-30]. Therefore, the goal of this paper is to suggest a principle of packet series synthesizer for the performance test of anti-DDOS flood attacks. The synthesizer consists two parts. One is simulation of random data series according to a given value of the Hurst parameter and the other packet shaper that transforms a pure data series to a packet series. The synthesizer has been explained and a case study demonstrated.

Appendix

The detailed discussions about (11) are given in [19,27]. However, the following brief explanation appears enough to show that (11) is a formula to synthesize a data series according to a given ACF by using white noise as a building block. Doing the Fourier transform on both sides of (11) yields

$$\mathrm{FFT}[G(n)] = \mathrm{FFT}[w]\ [\mathrm{FFT}(g(k; H))]^{0.5}. \quad (A.1)$$

As the power spectrum of w is 1, we have

$$|\mathrm{FFT}[G(n)]|^2 = \mathrm{FFT}[g(k; H)]. \quad (A.2)$$

Considering the Wiener-Khinchine relation, (A.2) implies that (11) holds.

References

[1] http://staff.washington.edu/dittrich/misc/ddos/.
[2] D. Dittrich, The DoS Project's ‘Trinoo’ Distributed Denial of Service Attack Tool, http://staff.washington.edu/dittrich/misc/trinoo.analysis.
[3] D. Dittrich, The ‘Tribe Flood Network’ Distributed Denial of Service Attack Tool, http://staff.washington.edu/dittrich/misc/tfn.analysis.txt.
[4] D. Dittrich, The ‘Stacheldraht’ Distributed Denial of Service Attack Tool, http://staff.washington.edu/dittrich/misc/stacheldraht.analysis.txt.
[5] D. Dittrich, The ‘Mstream’ Distributed Denial of Service Attack Tool, http://staff.washington.edu/dittrich/misc/mstream.analysis.txt.

[6] S. Dietrich, N. Long, and D. Dittrich, An Analysis of the 'Shaft' Distributed Denial of Service Tool, http://www.adelphi.edu/~spock/shaft_analysis.txt.

[7] V. Paxson, "An analysis of using reflectors for distributed denial-of-service attacks," *Computer Communication Review* 31 (3), 2001.

[8] http://www.intrusion-detection-system-group.co.uk/index.htm, *An Introduction to Intrusion Detection Systems and the Dragon IDS Suite*, 2001.

[9] P. Innella and O. McMillan, An Introduction to Intrusion Detection Systems, Tetrad Digital Integrity, LLC, http://www.securityfocus.com/infocus/1520, Dec. 2001.

[10] http://www.anml.iu.edu/ddos/links.html, Advanced Networking Management Lab (ANML), Distributed Denial of Service Attacks (DDoS) Resources.

[11] http://www.cisco.com/warp/public/cc/pd/sqsw/sqidsz/prodlit/ids4f_ds.htm.

[12] M. Li and W. Zhao, "A statistical model for detecting abnormality in static-priority scheduling networks with differentiated services," *2005 Int. Conf., Computational Intelligence and Security (CIS'05)*, Springer LNAI 3802, 2005, 267-272.

[13] M. Li, "An approach to reliably identifying signs of DDOS flood attacks based on LRD traffic pattern recognition," *Computer & Security*, 23 (7), 2004, 549-558.

[14] R. Bettati, W. Zhao, and D. Teodor, "Real-time intrusion detection and suppression in atm networks," *Proc., the 1st USENIX Workshop on Intrusion Detection and Network Monitoring*, April 1999.

[15] B. Todd, Distributed Denial of Service Attacks, http://www.opensourcefirewall.com/ddos_whitepaper_copy.html.

[16] M. Li, "Change trend of averaged Hurst parameter of traffic under DDOS flood attacks," *Computers & Security*, 25 (3), 2006, 213-220.

[17] L. Garber, "Denial-of-service attacks rip the Internet," *Computer*, 33 (4), 2000, 12-17.

[18] R. A. Kemmerer and G. Vigna, "Intrusion detection: a brief history and overview," *Supplement to Computer (IEEE Security & Privacy)*, 35 (4), 2002, 27-30.

[19] M. Li and C.-H. Chi, "A correlation based computational model for synthesizing long-range dependent data," *Journal of the Franklin Institute*, 340 (6-7), 2003, 503-514.

[20] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*, 2nd Edition, Cambridge University Press 1992.

[21] V. Paxson and S. Floyd, "Wide area traffic: the failure of Poisson modeling," *IEEE/ACM T. Networking*, 3 (3), 1995, 226-244.

[22] R. Saucier, Computer generation of statistical distributions, Technical Report, ARL-TR-2168, Army Research Laboratory, March 2000.

[23] R. M. Metcalfe and D. R. Boggs, "ETHERNET: distributed packet switching for local computer networks," *Communications of the ACM*, 19 (7), 1976, 395-403.

[24] V. Paxson, Fast, "Approximate synthesis of fractional Gaussian noise for generating self-similar network traffic," *Computer Communications Review*, 27 (5), 1997, 5-18.

[25] M. Li, "Modeling autocorrelation functions of long-range dependent teletraffic series based on optimal approximation in Hilbert space-a further study," *Applied Mathematical Modelling*, 31 (3) Mar. 2007, 625-631.

[26] W. Willinger and V. Paxson, "Where mathematics meets the Internet," *Notices of the American Mathematical Society*, 45 (8), 1998, 961-970.

[27] M. Li, W. Jia, and W. Zhao, "Simulation of long-range dependent traffic and a tcp traffic simulator," *Journal of Interconnection Networks*, 2 (3), 2001, 305-315.

[28] M. Li, "Modeling autocorrelation functions of long-range dependent teletraffic series based on optimal approximation in Hilbert space-a further study," *Applied Mathematical Modelling*, 31 (3), 2006, 625-631.

[29] M. Li and SC Lim, "Modeling network traffic using Cauchy correlation model with long-range dependence," *Modern Physics Letters B*, 19 (17), 2005, 829-840.

[30] S. C. Lim and M. Li, "Generalized Cauchy process and its application to relaxation phenomena," *Journal of Physics A: Mathematical and General*, 39 (12), 2006, 2935-2951.

[31] M. Li, Jun Li, and Wei Zhao, "Simulation study of flood attacking of ddos," *the IEEE 3rd International Conference on Internet Computing in Science and Engineering, the IEEE 3rd International Conference on Internet Computing in Science and Engineering, ICICSE08*, IEEE CS Press, 28-29 Jan. 2008, Harbin, China.

Ming Li was born in 1955 in Wuxi, China. He completed his undergraduate program in electronics engineering at Tsinghua University. He received the M.S. degree in ship structural mechanics from China Ship Scientific Research Center (CSSRC) and Ph.D. degree in computer science from City University of Hong Kong, respectively. From 1990 to 1995, he was a researcher in CSSRC. From 1995-1999, he was with the Automation Department, Wuxi University of Light Industry. From 2002 to 2004, he was with the School of Computing, National University of Singapore.

In 2004, he joined East China Normal University (ECNU) as a professor in both electronics engineering and computer science. He is currently a Division Head for Communications & Information Systems at ECNU. His research areas relate to applied statistics, computer science, measurement & control. He has published over 70 papers in international journals and international conferences in those areas. Li is a member of IEEE.

Wei Zhao is a professor of computer science and the dean for the School of Science at Rensselaer Polytechnic Institute. His research interests include distributed computing, real-time systems, computer networks, and cyberspace security. Zhao received a PhD in computer and information sciences from the University of Massachusetts, Amherst. He is a Fellow of the IEEE. He has published over 280 papers in international journals, international conferences, and book chapters.