A Novel Approach to Camouflaging the Transmission of a Digital File through the Internet based on RSF model

Rashiq R. Marie[1], Jonathan M. Blackledge[2]
Zarka Private University, Jordan[1]
Dublin Institute of Technology, Ireland[2]

Abstract

There is a number of previous studies have demonstarted the self-similarity(self-affinity) property of Internet network data traffic. Self-affinity and self-similarity are properties we associate with fractals. In this paper, firstly, we adopt the Random Scaling Fractal (RSF) model to capture the fractal behavior of internet traffic and to simulate their selff-affine characteristics. The relevance and validation of the proposed model have demonstrated by application of studies for measured Internet traffic. Secondly, we utilize the fractal nature of internet packets traffic for the application to securing data transmission through the Internet and instead of attaching a complete file (encrypted or otherwise) at a given time, we first encrypt the file, then split a binary representation of the file into a sequnce of binary files . The sizes of binary files are determined by the fractal characteristics of the Internet through which the data is to be transmitted using our proposed model. The data is then applied to the Internet as e-mail attachments and submitted according to a genrated sequence of inter-submission times computed that is computed using the same model. The recipient of the data recovers the information by concatenating the files sequence.

1. Introduction

The explosive growth in the Internet data traffic has made the study of the nature and statistical characteristics of this type of traffic increasingly important. Sesveral measurments and statistical studies have observed the fractal nature of Internet traffic [1], [2] and [3], (see Figure1). These studies have all convincingly demonstrated the presence of *self-affine* characteristics. Self-affinity and self-similarity are properties we associate with *fractals* - objects that appears the same regardless of the scale at which they are viewed. With Internet data, this it manifested in the absence of a natural length of a 'burst'; at every time scale, ranging from a few milliseconds to minutes and hours, bursts occur that consist of sub-periods separated by less bursty sub-periods. The commonly assumed models for network traffic data (e.g. the Poisson model) do not 'fit' the statistical characteristics of Internet traffic data, sinceily these models were not able to capture the fractal behavior of Internet traffic. Figure 1, taken from [4], illustrates the failure of the Poisson model in capturing internet traffic burstiness. Howevre, many models in the literature have considered the main characteristics of Internet traffic either by processing the real measurements of traffic in the time- or frequency-domain.

In this study, firstly, we adopt the *Random Scaling Fractal (RSF)* model , in the frequency domain, to capture the fractal behavior of internet traffic and to simulate their self-affine characteristics. The relevance and validation of the proposed model have demonstrated by application of studies for measured Internet traffic was obtained at the campus network of Loughborough University captured using the *tcpdump* utility. The measurements were made over the course of about one hour, and recorded all packets arriving at or originating from the host site [5].

The used measurements represent the number of packets(bytes) that arrived over the server at the Computing Services Department and the corresponding timestamps for their arriving. The data in its original formate was at the scale of 1μsc. We aggregated this data to resolve at different time scales 1 sec, 0.1 sec, 0.01 sec, and 0.001 sec for computational convenience. The resulting time series array with the number bytes(packets) that arrived in each time interval is considered as the set of observations in this work. Among all the possible traffic characteristics, our work focuses on two packet properties: *packet size* in bytes and *packet inter-arrival time*.

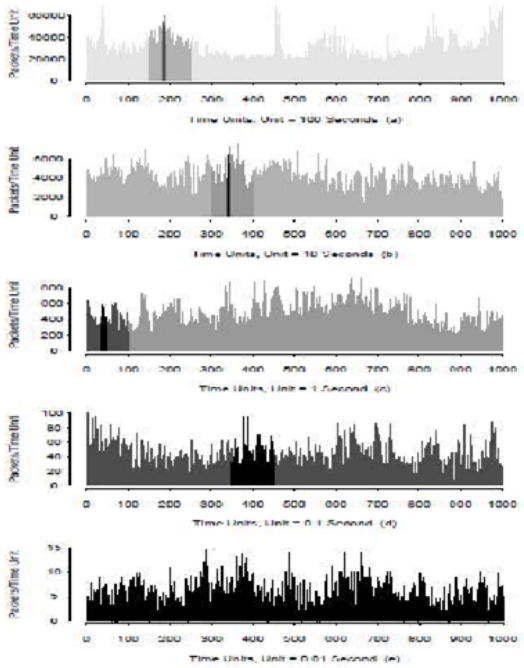

Figure 1. Pictorial "proof" of self-similarity: LAN traffic(packets per time unit) on five different time units (from [2]).

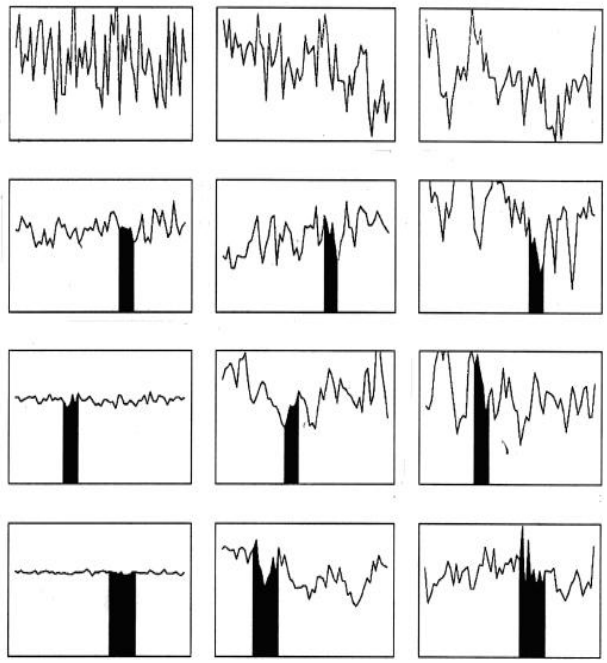

Figure 2. Simulated Poisson based time-series model (left), real Internet traffic time-series (centre) and simulated fractal time-series model taken over four different time scales (from top to bottom respectively). The shaded areas highlight the data displayed in each plots above respectively.

Secondly, we utilize the fractal nature of internet packets traffic for the application to securing data transmission through the Internet and instead of attaching a complete file (encrypted or otherwise) at a given time, we first encrypt the file (using Crypstic$^{\mathrm{TM}}$), then split a binary representation of the file into a sequnce of binary files . The sizes of binary files are determined by the fractal characteristics of the Internet through which the data is to be transmitted using our proposed model. The data is then applied to the Internet as e-mail attachments and submitted according to a genrated sequence of inter-submission times computed that is computed using the same model. The recipient of the data recovers the information by concatenating the files sequence.

To our knowledge, this approach is the first of its kind to use the self-affine nature of Internet traffic time series in order to disguise the transmission of a digital file. The rest of this paper is organized as follows. Section 2 provides the characterization of the Internet traffic using fractal model. In Section 3 we present the estimation of fractal parameter for Internet properties. In Section 4 we present the covert transfer of data through the Internet. Finally, in Section 5 we conclude with the main results.

2. Self-similar and fractal characteristics of Internet traffic data

The fractal concept was first introduced by Mandelbrot in the conext of turbulence in the early 70's. Since then fractal processes have been widdely used in averiety of reserach fildes like geophysics, image rocessing , stock market modeling and recently network traffic characterization. Fractals are applicable when the underlying process being mathematically modeled have a similar appearance regardless of the time or observation scale. It turns out that much of the traffic riding the Internet can be modeled using fractals [6]. It is indicated in [4] that we define a process to posses fractal (mono-fractal) characteristics, if there exist a relationship of the form

$$Q(\tau) \propto \tau^{D}$$

where Q is a certain quantity of the underlying process that depend on τ and D. Herein, τ denotes a resolution in $time$ or $space$ of observation variables at which Q is evaluated, and D is a $fractal \;\; dimension.$

To declare $fractality$, the above relationship is supposed to hold for a range of different τ-values, with a value of D that is less than the embedded dimension (i.e. $D < 2$). Due to the extreme variability the Internet traffic data exhibiting such fractal-like structures over almost all scales of resolution, the fractal characteristics can exist both in *temporal* and *spatial* scales. Several methods are commonly used for measuring the self-similarity dependence on a real network traffic trace, either in time or frequency doamin analysis. Shortly, before we introduce the adopted method in frequency domin, here we mention two examples of time domain-based methods, the *Resecaled Adjusted Range* (R/S/) plot and the *Variance Time* plot.

2.1 The R/S Method

The R/S method is one of the oldest and best known methods for estimating H. Let X_k, $k = 1, 2, 3, ..., N$ be a set of N observations for the number of bytes or packets in each interval (bin) and let $\overline{X}(N)$ and $S(N)$ be the mean and the standard deviation of these observations.

The R/S-statistic or *rescaled adjusted range* , is defined by the ratio:

$$\frac{R(N)}{S(N)} = \frac{U - L}{S(N)}$$

where ,
$U = max(0, w_1, w_2, w_3, ..., w_N)$,
$L = min(0, w_1, w_2, w_3, ..., w_N)$
$w_k = \sum_{i=1}^{k} X_i - k.\overline{X}(N), \qquad k = 1, 2, 3, ..., N$

Husrt found empirically, that for many time series observed in nature, they are well represented by the relation

$$\frac{R(N)}{S(N)} \approx C.N^H, \qquad N \to \infty$$

where C is a finite positive constant. By taking logs we obtain

$$log(\frac{R(N)}{S(N)}) \approx log(C) + H.log(N).$$

Therefore, the slope of a plot of $log(R/S)$ against $log(N)$ provides the Hurst parameter [7].

2.2 Variance-Time Method

This method relies on the slowly decaying variance of a self-similar series. Let X_k be a series of observations for the number of bytes(packets) in each interval (bin) $k = 1, 2, 3, ..., N$. If we take a sample of m points then the variance-time plot is obtained by plotting $log[Var(X^{(m)}]$ against $log(m)$ and by fitting simple lines through the resulting points in the plane. An estimate of the Hurst parameter is given by $H = 1 - \beta/2$ where β is slope of the plot [7].

2.3 Experimental Proof of the Fractal Nature of Internet Traffic

A simple way of understanding fractality (self-similarity) is in terms of scale-invariance. Basically, this means that whatever be the time-scale over which the traffic is plotted, the plots will appear (intuitively) very 'similar' to one another. Figure 3 shows four time series plots of size 1024 fot Internet traffic induced by a reference trace[5]. The horizontal axis represents the time scale; the vertical axis represents traffic load in bytes per unit time. The plots are produced by aggregating the (bytes) traffic into discrete time units (bins) of 1000 ms (1 s), 100 ms, 10 ms and 1 ms. The plots of the Internet packets traffic for the same trace is given in Figure 4, with the same considerations as those in the plots of bytes traffic whereas the vertical axis represents the number of packets per unit time.

We observe, from the plots given in each figure, that all the plots are 'similar' to each other, i.e. the bytes (or packets) traffic appears to look the same over the whole spectrum of time scales. Starting from a time scale 1000 ms, all subsequent plots are obtained from the previous one by decreasing the time resolution by a factor of 10. It is apparent that all the plots appear almost similar and exhibit self-similarity, indicating that different values of magnification give similar plots. In other words, all these plots show clearly the burstiness of the Internet traffic across many time scales (1 ms, 10 ms, 100 ms or 1000 ms). Increasing the time scale (time bin) of observation, say from 1 ms to 1000 ms, does not cause the traffic to 'smooth out' as would normally be expected. Instead, the traffic of all traces continues to exhibit burstiness.

Figure 5 shows the plot of sample of size 1024 points of inter-arrival packet time sequences corresponding to the given trace. The horizontal axis represents the sample point, the vertical axis represents the inter-arrival time in seconds. If we look at the plots of the time series of the Internet bytes or packets traffic or the plots of the packets inter-arrival time sequence, we see a similar appearance to the plots in each figure, regardless of time scale; this is the characteristic signature of the fractal behavior of such time series. Notice that, in each plot, the absence of a fixed length of a 'burst'; we observe different burst lengths on different time scales. This experimental observation is consistent with the prior studies.

Next we introduce the method in frequency domain analysis by which we try to capture the fractal behavior of Internet traffic in which we adopt a Random Scaling Fractal, RSF(q), model to characterize the self-affine characteristic of the Internet traffic.

2.4 Random Scaling Fractal Signal

Many signals observed in nature are random fractals. Random Scaling Fractal (RSF) are signals whose the probability distribution function of the RSF signals has the same 'shape' irrespective of the scale over which they are observed. So that, the RSF signals are *statistically self-similar* or *self-affine*; 'they look the same' in stochastic sense at different scale. We can define this property as follows: Suppose $s(t)$ is a signal, with $\Pr[s(t)]$ its PDF and λ is a scaling parameter, then the signal $s(t)$ exhibits statistical self-affinty if

$$\Pr[s(\lambda t)] = \lambda^r \Pr[s(t)], \qquad r > 0.$$

RSF signal are characterized by power spectra whose frequency distribution is proportional to $1/\omega^q$ where ω is the frequency and $q > 0$ is the 'Fourier Dimension', a value that is simply relate to the Fractal Dimension, D and Hurst (Dimension) parameter H. The relationship between the q, H and D is given by

$$q = H + 1/2 = (5 - 2D)/2$$

This power law describes the conventional RSF models which based on stationary processes in which the 'statistics' of the RSF signals are invariant of time and the value of q is constant. It is mention the Hurst parameter (Dimension) H measures the features of self-affinity of time series in real-time domain. Herein, we present the description of these features through processing the time series in the frequency domain in which we assume that the power spectrum of this signal is dominated by a RSF model $P(\omega) = c/\omega^q$, where $c > 0$.

Assume $X(t)$, in time domain, be a time series of the Internet bytes traffic or packets inter-arrival times which to be assume a self-affine signal, (see Figure 3, Figure 4 and Figure 5). The power spectrum of such signal can be written as $P(\omega) = |X(\omega)|^2$, where $X(\omega)$ is Fast Fourier Transform (FFT) of the time series in frequency domain (i.e. $X(\omega) = fft(X(t))$). For such time series the power spectrum, $P(\omega)$ obeys the RSF model

$$P(\omega) = \frac{c}{\omega^q}$$

The Figure 6 show example of different plots of the measured power spectrum of Internet bytes traffic over four different time units. The Figure 7 shows the plot of measured power spectrum of packet inter-arrival times. These figures give the evident that the power spectrum of the time series signal of Internet traffic obeys the RSF model. However, we can characterize the behavior of Internet traffic time sereis through estimating the parameter q in the proposed model where the estimated values of this parameter will reflect the degree of self-similarity (fractality) in real Internet

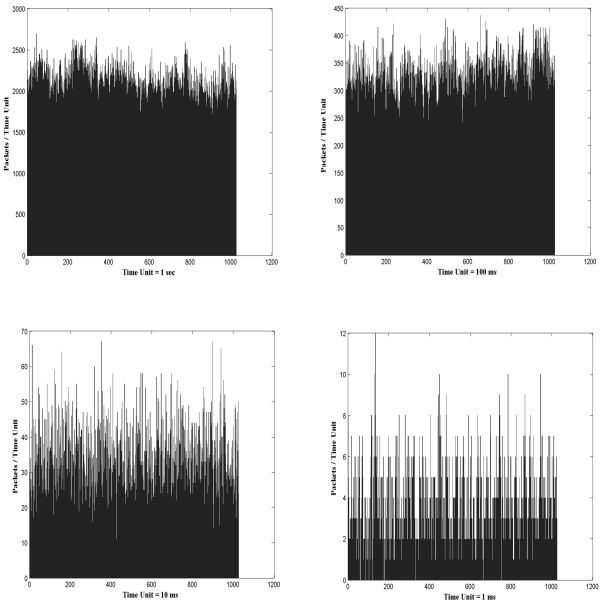

Figure 3. Number of packets/s for four time scales; from upper to lower and from left to right: 1, 0.1,0.01 and 0.001 second

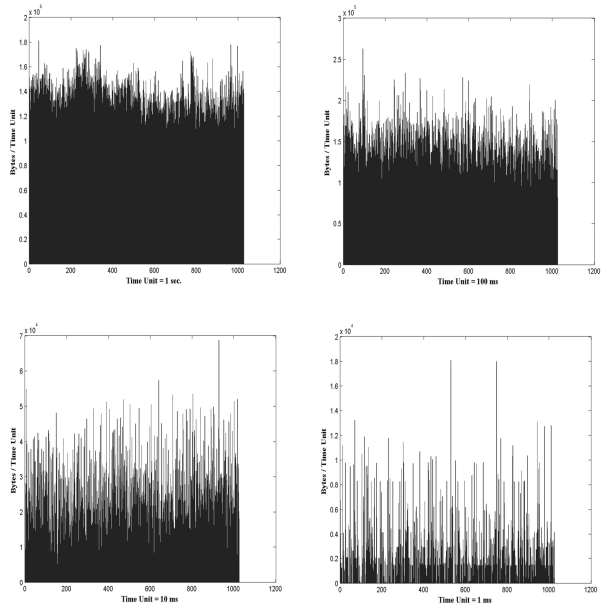

Figure 4. Number of bytes/s for four time scales; from upper to lower and from left to right: 1, 0.1,0.01 and 0.001 second

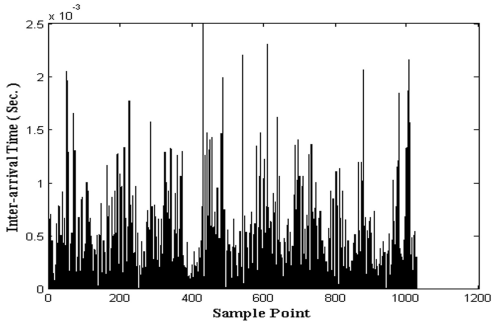

Figure 5. Inter-arrival times of packets.

traffic. To do this, and in the following section, we apply the Least Square Method on the measurements of Internet traffic.

2.5 Power Spectrum Analysis (PSA)

Let $X_i, \quad i = 1, 2, 3, ..., N$ (N being a power of 2) be an aggregated time series that represent the number of bytes (packets) that occurred in a predefined unit time, this unit time may 1 sec, 100 msec, 10 msec, or 1 msec. Consider the case in which the digital power spectrum $P_i \equiv P(\omega_i)$ is given by applying a FFT to this time series. This data can be approximated by

$$F(\omega_i) = \frac{c}{\mid \omega_i \mid^{\frac{q}{2}}}$$

or

$$\mid F(\omega_i) \mid^2 = \hat{P}(\omega_i) = \frac{c}{\mid \omega_i \mid^q}$$

If we consider the error function

$$E(q, c) = \sum_{i=1}^{N} [\ln P(\omega_i) - \ln \hat{P}(\omega_i)]^2$$

$$= \sum_{i=1}^{N} [\ln P(\omega_i) - C + q \ln \omega_i]^2$$

where $C = \ln c$, and it is assumed that the spatial frequency $\omega_i > 0$ and the measured power spectrum $P(\omega_i) > 0$, $\forall i$ then the solutions of equations (least square method) $\frac{\partial E}{\partial q} = 0$ and $\frac{\partial E}{\partial C} = 0$ gives

$$q = \frac{N \sum_{i=1}^{N} (\ln \omega_i) \ln P(\omega_i) - (\sum_{i=1}^{N} \ln \omega_i)(\sum_{i=1}^{N} \ln P(\omega_i)}{(\sum_{i=1}^{N} \ln \omega_i)^2 - N \sum_{i=1}^{N} (\ln \omega_i)^2}$$

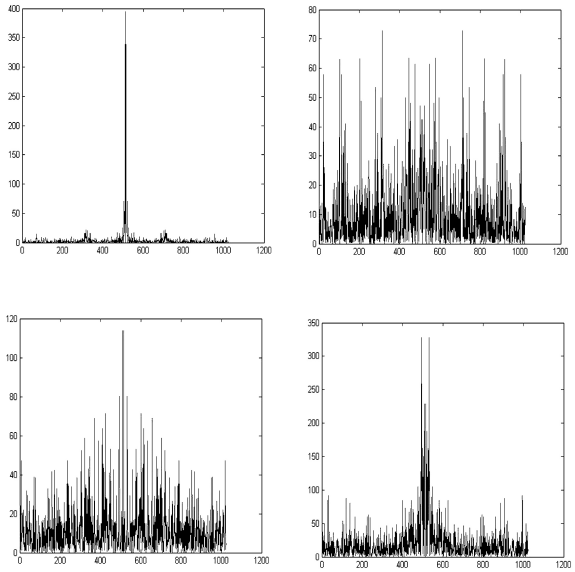

Figure 6. Measured power spectrum of bytes traffic, from top to bottom and from left to right: 1sec (q=0.45), 100ms (q=0.28), 10ms (q=0.10) and 1ms (q=0.15).

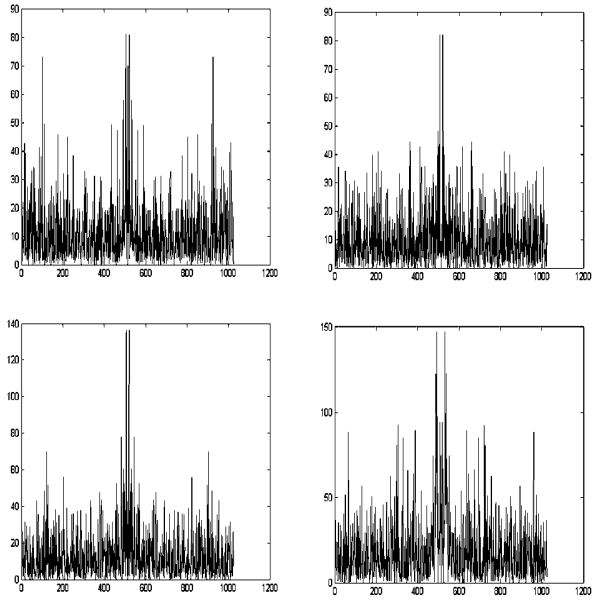

Figure 7. Measured power spectrum of packets inter-arrival times, from top to bottom and from left to right: (q=0.08), (q=0.15), (q=0.19) and (q=0.11).

and

$$C = \frac{1}{N}\sum_{i=1}^{N} \ln P(\omega_i) + \frac{q}{N}\sum_{i=1}^{N} \ln \omega_i$$

Since the power spectrum of real signals of size N is symmetric about the DC level, where the DC level is taken to the mid point $\frac{N}{2} + 1$ of the array, so in practice we consider only $\frac{N}{2}$ of data that lie to the right of DC [8].

3. Estimating Fourier Dimension

The two main properties of network packets are their *sizes* in bytes and their *inter-arrival* times (timestamps differences between consecutive packets) and , as shown above, both of them are obey to the Random Scaling Fractal, $RSF(q)$, model. To estimate the fractal parameter in these series we convert them to frequency domain in which we assumed that the empirical power spectrum of each series has an envelop Power Spectrum Density Function (PSDF) which given as the RSF model $P(\omega) = |\omega|^{-q}$.

By using Moving Window technique, we choose a window of size $N = 1024$ to move over the points of the time series according to the given time unit. From each window segment we apply the PSA to estimate the Fourier Dimension q, after transforming the given segment to the spectral domain. The following algorithm summarizes the steps of the estimation process:

1. Use a window of size N=1024 over the points of a given time sereis to extract a segment array, say X_i, normalize the segment to get Y_i, $i = 1, 2, 3, ..., N$.

2. Compute DFT of Y_i using a Fast Fouruier Transform (with spectral shifting) to yield Z_i

3. Compute the empirical power spectrum, $P = |Z_i|^2$, and extract the right havle of P.

4. Compute the Fourier Dimension q using the computaional formula of the PSM given above.

5. Iterate step 1 through to step 4 until the end of the time series.

Table 1 and Table 2 gives the results of estimation of q form the packet sizes for over different time scales and packet inter-arrival times, respectively. From these results we note as the time unit increases from 1ms to 1000 ms then as well the estimated value of q increases. In the meanwhile, the higher the load on the Internet the higher the estimated Fourier dimension i.e., the degree of self-similarity in the arriavl rate process (in termes of packets or bytes). Visual comparisons between the different traces also suggest that large q, the "burstier" the corresponding trace appears.

Table 1. The estimated values of q, on different time scales.

Wind. No.	1 ms **Byte**	10 ms **Byte**	100 ms **Byte**	1 s **Byte**
1	0.07	0.02	0.25	0.77
2	0.10	0.17	0.26	0.58
3	0.10	0.10	0.18	0.74

Table 2. The estimated values of q for packet Inter-arrival series.

Wind. No.	1	2	3	4	5
est(q)	0.14	0.26	0.10	0.11	0.17

4. Covert Transfer of Data through the Internet

In this section we introduce novel method of using the fractal characteristics of Internet traffic to camouflage the transfer of a digital file through the Internet. For applications to securing data transmission through the Internet, instead of attaching a complete file (encrypted or otherwise) at a given time, we split a binary representation of the file into a number of blocks depending on a generated fractal trace.

Each block is then saved as a sequence (a trace) of binary files in which the statistical characteristics of such traces fit with the characteristics of the actual trace of Internet packets. The sizes of the split files are determined by the fractal characteristics of the Internet through which the data is transmitted using the Random Scaling Fractal model. At the same time, a sequence of inter-submission times are generated using the same model. This sequence is used to formulate the required timestamps. The data is then applied to the Internet as e-mail attachments and submitted according to the sequence of timestamps computed(see Figure 8). The recipient of the data recovers the information by concatenating the files sequence.

4.1 Generation of Fractal Trace

The method of generation a synthetic traffic trace for the purpose of securely transferring a set of files as e-mail attachments is considered. Generally, this equates to the problem of letteing the behavior of e-mail arrivals match the behavior of packets observed in actual computer networks. In what follows, we introduce the steps of the algorithm associated with method to generate a synthetic trace or sample pathe which dispalys the same statistical properties as the actual data traffic.

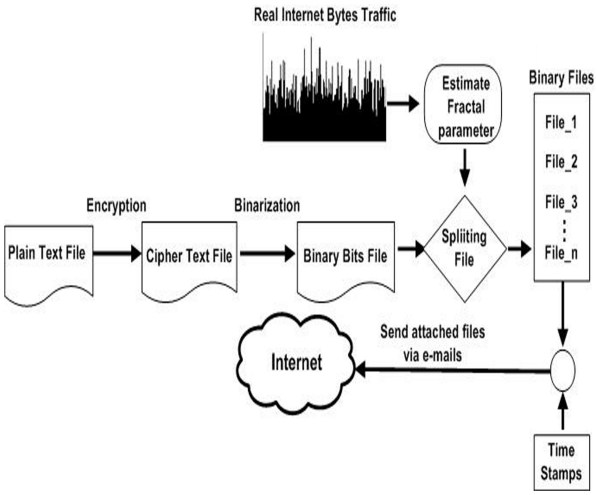

Figure 8. The mechanism of sending a digital file over the Internet.

In this way, we generate synthetic trace or sample path that exhibit the same statistical properties as the real measured packets traffic. The points of the sample path represent the sizes in bits of the split files. In general, this method of generating a synthetic trace with fractal characteristics depends on generating white Gaussian noise that is filtered using the Random Scaling Fractal model with a suitable value of the parameter q.

To ensure that the synthetic trace is representative of a trace that is likely to be encountered in the 'real world', the synthetic trace is used the estimated values of q from the captured traces of actual Internet traffic, see Table 1. The inputs to the method are q, the desired Fourier dimension, and N, the desired number of points in the synthesized sample, (where N is a power on 2) . In what follows, we give the description of the method.

1. Generate a stream of white Gaussian noise, sayX_i, where $i = 1, 2, 3, ..., N$.
2. Compute the Discrete Fourier Transform (DFT) of X_i using a Fast Fourier Transformation (FFT) to give a new series Y_i.
3. Apply the fractal filter on Y_i using the RSF model to obtain Z_i.
4. Compute the inverse DFT of Z_i using an FFT to give W_i.
5. Consider the real part of W_i giving fractal signal.
6. Compute the Hilbert transform of the fractal signal
7. Compute the modulus of H_i and normalize the result h_i.
8. Rescaling the sequence h_i, and then round down the results to the nearest integer towered zero, we obtained the sequence R_i. The resultant sequence of interger values constitute the $sizes$ series of the of the actual files (packets) sizes as described above.

   Figure 9 shows four time series plots of synthetic sample paths. Each path is of length 1024. The horizontal axis represents the sample number; the vertical axis represents traffic load in bytes per time unit. The plots are produced according to the algorithm given above and with four different estimated values of the Fourier dimension $q = 0.59, 0.2, 0.12,$ and 0.2. From this figure we see clearly that the plots of the generated series using the algorithm described above appear to be somewhat like to the plots of the actual series of the Internet traffic (e.g. see Figure and Figure 4).

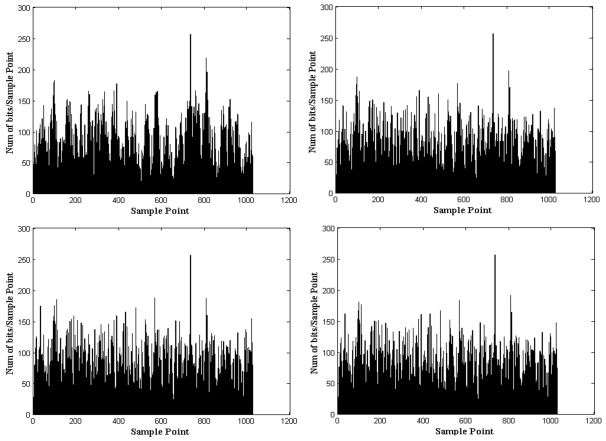

Figure 9. Synthetic Internet Bits Traffic Bursts over Four Orders of Magnitude: 1 sec, q=0.6 (top-left); 100 ms, q=0.3 (top-right); 10 ms, q=0.13 (bottom left) and 1 ms, q=0.2 (bottom-right).

4.2 Generation of the timestamps sequence

Similarly, the synthetic sequence of inter-submission times is generated which used to produce the timestamps in military form. A timestamps sequence, say $T_1, T_2, T_3, ..., T_N$ is required to send each file in the trace of files at time intervals compatible with the fractal characteristic of packets arrival times, i.e. the fractal time signature. The principal steps to generate such sequence are the same as the steps used in generating frcatl trace of files sizes, but here the value of fractal parameter is taken form Table 2.

After that, write the sequence of timestamps in 'military form', i.e, hh:mm:ss (hours, minutes and seconds respectively). However, to do this, we first consider an initial and then add the inter-submission values obtained to the initial time in an accumulative way until the last timestamp is reached. Table 3 shows the first 10 points of a sequence of timestamps that is obtained by applying the above algorithm according to the RSF model with $q = 0.10$ and the initial time is '22:10:45'. Figure 10 shows the plot of 1024 sample points of the sequence of inter-submission with $q = 0.10$. Note that the plot of the inter-submission time sequence is similar to the plot of the actual packet inter-arrival times sequence (see Figure 5) where the estimated value of q from this sequence is 0.10.

Table 3. The first 10 points of synthetic Inter-submission times with the corresponding Timestamps.

Inter-submission Time (Second)	Timestamp
53	22:11:38
8	22:11:46
7	22:11:53
7	22:12:00
4	22:12:04
43	22:12:47
28	22:13:15
21	22:13:36
12	22:13:48
5	22:13:53
.	.
.	.
.	.

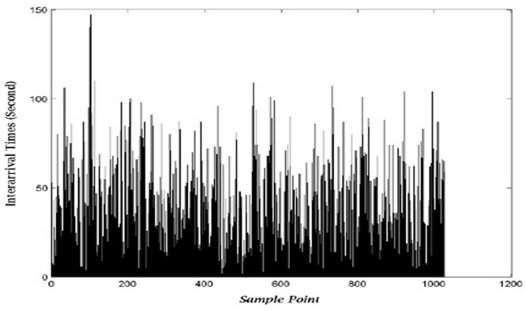

Figure 10. Plot of the synthetic sequence of inter-submission times for $q = 0.10$

4.3 Transmission of files between a sender-receiver pair on a network

In recent years the Internet has become the most popular media for information transfer in the world. Terms like 'e-mail' and 'e-commerce' are well known and accessible to everyone. Here, the resulting series R_i where $i = 1, 2, 3, ..., N$ represents the synthetic fractal trace, and N is the length of this trace. Each element in the generated trace is considered to be the size of each file of bits, and the length of such trace is the number of the split files. We describe the mechanism of sending and receiving the files below. In light of the above, if we consider the set of split files to be $file_1, file_2, file_3, ..., file_N$ then $size(file_1) = R_1$, $size(file_2) = R_2$, ..., $size(file_N) = R_N$. These files are then sent over the Internet according to the sequence of timestamps, $T_1, T_2, T_3, ..., T_N$.

Sending Files

For the application of securing data transmission through the Internet, instead of attaching a complete file (encrypted or otherwise) at a given time, we first encrypt the file and then split a binary representation of the file into a number of blocks. Each block is then saved as a sequence of binary files whose size is determined by the fractal characteristics of the Internet through which the data is to transmitted using the Random Scaling Fractal model.

At the same time, a sequence of inter-submission times are generated using the same model. The data is then applied to the Internet as e-mail attachments and submitted according to the sequence of inter-submission times computed.

Here, it is assumed that the inter-submission times are compatible with the inter-arrival times in that they adhere to the same fractal model that is assumed to be stationary over the given period of interest.

To make the traffic associated with a transmitted file sequence (packets) compatible with the fractal nature of internet traffic, we send an e-mail with one attached file only, say file $file_i$, at a timestamp T_i, where $i = 1, 2, 3, ..., N$. Figure 8 shows the block diagram of the mechanism for sending a digital file over the Internet by splitting its binary representation into a trace of digital files.

Receiving Files

Once all the attached files have arrived at the recipients location, they are recombined into their original file. For the purpose of increasing the security of the communication, we first encrypt the whole file before binarization. Upon reception and after concatenation of the binary contents of the received files, we decrypt the resultant to recover the plaintext file. The operation of encryption and decryption can be based on any algorithm but here, we used the Crypstic^{TM} system.

5. Conclusion

We studied the series of byte counts on different aggregation level of 1 sec, 0.1 sec, 0.01 sec and 0.001 sec. and the sequence of packets inter-arrival times. The analysis of our trace demonstrated that the Internet traffic appeared to agree with the finding of previous studies, showing that fractal characteristics of Internet traffic.

For application of sending a digital file through the Internet we introduced a novel approach in which the fractal characteristics of Internet traffic have been considered, in a way, that the synthetic trace of split files fit the actual trace of Internet traffic. Indeed, to our knowledge, this approach is the first of its kind to use the self-affine nature of Internet traffic in order to camouflage the transmission of a digital file by splitting the file into a number blocks (files) whose size and submission times are compatible with the bursty lengths of Internet traffic. As a future work we could use other fractal model to capture the fractal behavior of Internet traffic as the Generalized Random Scaling Fractal *GRSF(q,g)* model in which we try to describe the self-similraity of Internet traffic depending on two paremters *q* and *g*.

6. References

[1] W. Leland, M. Taqqu, W. Willinger, and D. Wilson, On the Self-Similar Nature of Ethernet Traffic (Extended Version), *IEEE/ACM Transactions on Networking*, 2(1), 1994, 1-15.

[2] V. Paxon and S. Floyd, Wide-area traffic: The failure of Poisson modeling, *IEEE/ACM Transactions on Networking*, 3(3), 1995, 226-244.

[3] M. E. Crovella and A. Bestavros , Self-similarity in World Wide Web traffic: Evidence and possible causes , *IEEE/ACM Transactions on networking*, 5(6), 1997, 835-846.

[4] W. Willinger and V. Paxson, Where Mathematics Meets the Internet,*Notices of the AMS*, Vol 45, No.8, P961-970, 1998.

[5] The Internet traffic archive, Personal communication with Network manager at Computing Services Department/ Loughborough University, UK, 2009.

[6] J. M. Blackledge, *Digital Signal Processing:Mathematical and Computation Methods, Software Development and Applications*, (London: Horwood Publishing Limited, 2nd Edition, 2006).

[7]Beran J., *Statistics for Long-Memory Processes*, Chapman and Hall, New York, 1994.

[8] M. Turner, J. M. Blackledge and P. Andrew , *Fractal Geometry in Digital Imaging*, (UK: Academic Press Ltd., 1998).

[9] E. Marke, M. Crovella and A. Bestavors, Explaining World Wide Web traffic Self-similarity, *Technical Report TR-95-015.*, Boston University, Computer Science Department,1995.

[10] D. Chakraborty, A. Ashir, G. Suganuma, K. Mansfield, T.Roy and N. Shiratori, Self-similar and fractal nature of Internet traffic, *International Journal of Network Management*, 14(2), 2004, 119-129.

[11] Liu Shu-Gang, Wang Pei-Jin and Qu Lin-Jie, Modeling and simulation of self-similar data traffic, *Proc. of the 4th International conference on machine learning Cyberetics*, Guangzhou, 2005, 18-21.

[12] Bo R. and Lowen S., Fractal traffic models for Internet simulation, *Proc. of fifth IEEE Symposium (ISCC 2000)*2000, 200 - 206.

[13] Paxson V., Fast approximation of self-similar network traffic, *Lawrence Berkeley National Lab.*, Tech. Rep., LBL-36750, Berkeley, USA, Apr. 1995.