# REPRINT


## Discussion of a Statistical Channel

*Ira S. Moskowitz* and *Myong H. Kang*

FROM:
Proceedings of the IEEE-IMS Information Theory Workshop on Information Theory and Statistics, page 95, Alexandria, VA, October 27-29, 1994.

CONTACT:
Ira S. Moskowitz and/or Myong H. Kang, Information Technology Division, Mail Code 5540, Naval Research Laboratory, Washington, DC 20375.

E-MAIL:
moskowit@itd.nrl.navy.mil mkang@itd.nrl.navy.mil

# Discussion of a Statistical Channel

Ira S. Moskowitz & Myong H. Kang
moskowit@itd.nrl.navy.mil, mkang@itd.nrl.navy.mil
Information Technology Division—CHACS: Code 5540, Naval Research Laboratory, Washington, DC 20375,USA

*Abstract* — **This paper deals with a new type of covert channel problem that arose when we designed a multilevel secure computer (MLS) system, using a quasi-secure, asynchronous, communication device called the *Pump*. We call this new type of covert channel a statistical channel. It is our hope to get feedback from experts who work in the intersection of information theory and statistics.**

## I. Introduction

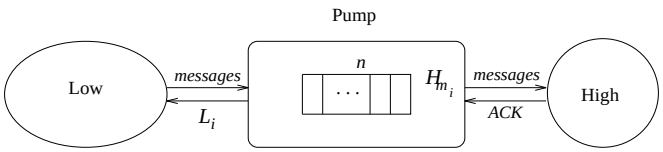
In a (MLS) system, Low may write to High, and High can read from Low, but High must never be able to write to Low. However, in a MLS system, the need for an acknowledgement (ACK), which is a write from High to Low, to a message sent by Low to High can violate the multilevel security policy by creating a covert (communication) channel.

Consider a case where Low sends messages to High. A simple approach that does not allow High to send an ACK to Low places a buffer between Low and High. Low submits messages to the buffer, the buffer sends the ACKs back to Low, and High then takes messages from the buffer. If the Low (sending) rate is faster than the High (receiving) rate, Low will write over unread data in the buffer (since the buffer is finite). An obvious solution to this problem is to not allow Low to send messages until there is a space in the buffer. This, however, results in a large capacity covert channel between High and Low (if Low is not allowed to send messages to a full buffer, then High can send symbols to Low by removing or not removing messages from the buffer and hence causing the buffer to be full or to have space on it).

## II. The Pump

Our approach, the Pump [1], still places a buffer (size $n$) between Low and High, but has the buffer give ACKs at probabilistic times to Low based upon a moving average of the past $m$ High response times ($H_{m_i}$). A high response time is the time from when the buffer tells High that it has a message to the time when High actually removes it. This has the double benefit of keeping the buffer from filling up and having a minimal negative impact upon performance.

Using a moving average is a very important part of the Pump. However, it gives rise to a new type of timing channel (for detail, see [1]). We will now sketch an implementation of the Pump. Let $O_v$ be the communication overhead for the Pump. By this we mean that $O_v$ is the minimum value for any $L_i$ (which is the $i$th response to Low). The $L_i$ are given by a random variable that has the density function $f_i(t)$.




There are two cases to discuss:

**Case 1:** The buffer is not full.

$$f_i(t) = \begin{cases} \alpha_i e^{-\alpha_i(t-O_v)}, & if\ \ O_v \le t, \\ 0, & otherwise. \end{cases}$$

The mean of the above density function is $O_v + 1/\alpha_i$. Since we wish for this mean of $f_i(t)$ to be equal to the moving average of the last $m$ High ACK times ($H_{m_i}$) we see that $\alpha_i = 1/(H_{m_i} - O_v)$. If $H_{m_i} = O_v$, then set $1/\alpha_i = \epsilon$, a small number.

**Case 2:** The buffer is full.
This case is not germane to this paper.

## III. Covert Channels

A timing (covert) channel exists when the output (Low) alphabet consists of the different times of the same response, these different times (e.g.,`yes` arriving at $3t$ or $5t$) being due to High behaviour. Historically, work on timing channels has used very simple tools from information theory, for example [2]. In the course of our work we have come upon a new type of timing channel that defies analysis by *our* research community. It is our hope that, by presenting a paper at this workshop, we will get feedback from experts who work in the intersection of information theory and statistics.

We introduce a new subspecies of timing channel referred to as a *statistical channel*. The Low alphabet consists of different time values and these time values are given by a random variable with certain parameters and these parameters are dependent upon High actions.

**Definition 1** *If High can affect a parameter in the distribution of some system response time to Low, we say that there is a* **statistical channel** *between High and Low.*

In the Pump, High can modify the moving average by affecting the last $m$ time values of High's responses to the Pump. It is possible for Low to detect differences in High's actions by trying to guess what the moving average is. This creates a statistical channel and, therefore, insecurity. For now, let us forget that the exponential density has been shifted by the communication overhead time, and simply view the inputs to the channel as the High response times. We state a simpler form of our problem as:

> *What is the capacity, in bits per unit time, of a communication channel where the output is an exponential random variable whose mean is the moving average of the past m input times?*

## References

[1] Kang, M. H. and I. S. Moskowitz. "A pump for rapid, reliable, secure communication," Proceedings of the 1st ACM Conference on Computer and Communications Security, pp. 119–129, 1993.

[2] Moskowitz, I. S. and A. R. Miller. "The channel capacity of a certain noisy timing channel," IEEE Transactions on Information Theory, vol. 38, number 4, pp. 1339–1344, 1992.