Adaptable Traffic Masking Techniques for Traffic Flow Confidentiality on Internetworks

by Brenda Timmerman
## Contents

Summary .......................................................... 2

1 Introduction .................................................. 3
   1.1 Problem Statement ...................................... 5
   1.2 Proposed Approach ..................................... 6

2 Related Work ................................................ 8
   2.1 Theoretical Security Models............................. 8
   2.2 Related Proposed Technology........................... 9

3 Preliminary Results .......................................... 12
   3.1 Traffic Masking Schemes................................ 12
       3.1.1 Categorizing Masking Schemes..................... 13
       3.1.2 Maintaining Protection with Statistical Anomaly Masking............. 16
       3.1.3 Enhancing Schemes for Improved Efficiency and Performance......... 18
   3.2 Traffic Masking Modules................................ 19
       3.2.1 Formal Description................................. 19
       3.2.2 Example of Traffic Masking Module.................. 21
   3.3 Formal DATM Security Model............................ 22
       3.3.1 Definition of Security Model....................... 22
       3.3.2 Constraint on the Transition Function.................. 24
       3.3.3 The secure state invariant........................ 24
       3.3.4 Example of DATM Security Model.................... 26

4 Research Plan .............................................. 28
   5.1 DATM and Security Models.............................. 28
   5.2 Implementation and Evaluation of Traffic Masking Modules.............. 28

5 Research Contributions .................................... 33

References .................................................... 35
Summary

Traffic flow confidentiality (TFC) provides protection from traffic analysis by masking frequency, length, and origin-destination traffic patterns of communications between network protocol entities. Secure end-to-end communication is now a possibility on internetworks, but some internet users need additional protection from traffic analysis and consider TFC too expensive for practical use. Recently proposed technology outlines policies for dynamic adaptation to reduce the costs of traffic masking and respond to application level service requirements. However, these policies do not adequately address the costs of protection or applications’ performance requirements. They also do not consider system protection needs and the reduction in protection that result from dynamic adaptation. To address these deficiencies, this proposal presents techniques for dynamic adaptable traffic masking (DATM) that include statistical anomaly masking (SAM) mechanisms. These new techniques provide traffic masking that meets users’ security needs with minimal cost while allowing dynamic adjustments for performance considerations. In addition, this proposal defines a formal security model that precisely specifies the security requirements for adaptable mechanisms. Further, it presents a plan to implement a traffic masking module in a UNIX network environment and verify the correctness of the design by formally proving the correspondence of the implementation to the formal model. The developed technology will integrate into existing security protocols as part of a solution to provide TFC in open internetworks.
1 Introduction

Traffic flow confidentiality (TFC) provides protection from traffic analysis by masking frequency, length, and origin-destination traffic patterns of communications between network protocol entities. Traditionally TFC has been provided by link encryption between protected sites, but as secure communication follows the trend of migration to public networks, new technology is needed. The projected evolution of security on networks is illustrated in Figure 1 showing the change in protection as secure hosts move outside of protective gateways, but still require operational security. TFC helps maintain privacy even on a public network.

Dedicated private networks are too expensive for wide scale network use that has become common practice. Public networks are not only cheaper, but are more available and reliable. Users of public networks now have the capabilities for end-to-end operational security, with such features as confidentiality, integrity, and varying degrees of privacy, but some internet users need additional protection from traffic analysis. Their internet traffic becomes vulnerable to unauthorized observation when they do not have authority over all the links on which the traffic flows. Internet users that may need TFC are military, other government agencies, highly competitive businesses that desire protection from industrial espionage, and other users that need the details of their communications protected. Some expected uses of TFC mechanisms are private communications
between secure LANs over unsecure internetworks, private email, file transfers among individual workstations, mobile users with secure IP, and networks with wireless links.

Even on public networks TFC is frequently considered too expensive or detrimental to performance to be considered practical. This is an issue it shares with other forms of security, but one that is particularly applicable to TFC because of its frequent use of use of padding to mask traffic patterns. Recently proposed technology outlines policies for traffic masking that can make dynamic adjustments in response to changing rates of original traffic. The rate of the masked traffic increases and decreases as the rate of the protected original traffic increases and decreases. Dynamic adaptation can reduce the costs of traffic masking and respond to application level service requirements, but the proposed policies do not address system’s protection needs or the reduction in protection that results from dynamic adaptation. Also, they do not adequately address the costs of protection or applications’ performance requirements. They require unrealistic constraints on their models, such as fully connected nodes with global synchronization, that are not scalable to internetworks.

As part of an improved solution for traffic analysis, this proposal presents new dynamic adaptable traffic masking (DATM) techniques for secure TFC mechanisms that have the capability to dynamically adapt to changing network conditions. It also defines a formal security model that precisely specifies the security requirements for DATM mechanisms within a network environment. The goals of the model are to satisfy system security requirements, minimize padding costs, and meet the throughput and delay requirements of the original traffic. The proposed techniques include statistical anomaly masking (SAM) that uses statistical methods to detect and prevent leaks of inference information that may occur when dynamic adjustments are allowed. In addition to specifying system protection requirements, the proposed security model precisely specifies the relationships between costs of protection and original traffic performance requirements and allows trade-offs between the two, both in the design process and in dynamic adjustments, that meet system security policies.

The rest of this section further describes the problem and presents an approach to solving it. Related work is discussed in section 2. Preliminary results are discussed in section 3, including
traffic masking schemes, DATM techniques, and a formal security model specifying protection requirements for DATM mechanisms. Section 4 presents a research plan including an implementation and evaluation of a traffic masking module based on the proposed techniques, and section 5 discusses the expected contributions of this research.

1.1 Problem Statement
The fundamental mechanisms of TFC are encryption, traffic padding and delays, and routing control. They introduce noise on connections between network protocol entities in the form of padding traffic and added delays, and protect network traffic from unauthorized analysis by masking frequency and length patterns. TFC mechanisms also protect origin-destination traffic patterns by masking, or otherwise obfuscating, protocol components (such as addresses). The location of TFC mechanisms in the network architecture determine what information is masked.

TFC is frequently considered too expensive for practical use, in terms of both bandwidth consumption and reduced performance. It has been recognized that computer system security mechanisms that are adaptable to their environment can reduce costs and improve both system and application performance [BAD90][BRO94][HW89][VEN94]. However, while it is acknowledged that such adjustments do introduce unwanted flows of inference information about the protected systems, solutions to this problem are not addressed. Previously implemented technology for end-to-end TFC protection on internetworks provides only padding mechanisms that leave decisions of how or when to pad to local authorities [MAC93]. Other proposed theoretical solutions for TFC do not address systems’ security policies or the network environment in which they will be placed. Recently proposed technology for transport layer protection from traffic analysis outlines policies for dynamic adaptation, but these policies do not address system’s protection needs or the reduced degree of protection that can be introduced by dynamic adaptation. While they can reduce costs of traffic masking and respond to application level service requirements, the policies do not adequately address protection costs or applications’ performance needs. In addition, there are unrealistic constraints on the models, such as fully connected nodes with global synchronization.

Traffic masking schemes that are adaptable, i.e. make adjustments for changing network condi-
tions or application service requirements, can cause *statistical anomalies* in the masked traffic patterns. Schemes that are not adaptable are independent of original traffic characteristics and therefore not vulnerable to anomaly detection, but they also do not make adjustments to improve performance or efficiency. Efficiency is the ratio of original traffic over masked traffic and approaches the value one as it increases. Schemes that do make adjustments for improved performance and efficiency can create patterns of traffic that, while different from original traffic patterns, may infer characteristics of the original traffic that can be analyzed. Such schemes can be attacked by detection of statistical anomalies in traffic characteristics from which the type of traffic being transmitted can be inferred.

### 1.2 Proposed Approach

The approach of this research is to first identify important traffic characteristics that can be analyzed for inference about original traffic and then present a plan for masking them. It presents solutions for traffic masking that meet users’ protection needs while minimizing the costs of protection and still meeting application performance requirements. It develops DATM techniques for schemes that produce a combination of deterministic and probabilistic patterns of traffic transmission made up of original traffic, padding traffic, and introduced delays. The resulting schemes have the capability to dynamically adapt to changing environmental conditions in order to meet the throughput and delay requirements of the original traffic and to reduce the cost of padding. The proposed SAM techniques use statistical methods to detect and prevent inference channels of information about original traffic characteristics that may be created by dynamic adjustments.

In addition to specifying system protection requirements, the proposed formal security model precisely specifies the relationships between costs of protection and original traffic performance requirements in terms of these characteristics. It allows trade-offs between the two, both in the design process and in dynamic adjustments, that meet system security policies and increase efficiency and performance.

This proposal outlines further examination and development of DATM schemes that make dynamic adjustments of padding and delays to meet changes in the rate and service requirements of original traffic, and outlines how covert inference channels of information about original traffic
can be controlled with SAM techniques. Anomaly detection mechanisms can be anticipated and exploited when designing traffic masking schemes that have the ability to make dynamic adjustments. In anomaly detection, statistics are kept as frequency tables, means, and covariances. If the statistics of observed data are sufficiently far from expected values with respect to the statistics of the profile, the data is considered anomalous. For example, a summary test statistic on collected data [JV91] can reflect both the extent which recent behavior differs from the historical profile and the degree of abnormality of the recent behavior based on a historical probability distribution of a recently collected statistic. The larger the summary test statistic, the greater the degree of abnormality of the recent behavior.

Important features of masked traffic that can be analyzed for information about the original traffic are: the size of the protocol data unit (PDU), the packet burst size, inter-packet arrival times, and throughput and delay. They can be manipulated in a traffic masking scheme to create a profile of “normal” behavior that will accept most anticipated adaptations as normal. A DATM scheme may have to do real time calculations based on events in systems environment. One advantage to using statistical analysis is that it is not necessary for schemes to keep or process extensive information about past behavior as this information is stored in the statistics. A summary test statistic can accumulate data in an additive fashion. Recent behavior can be stored as an exponentially weighted sum of changes that have occurred. This sum can be weighted so that recent history is dominant. Therefore the use of a summary test statistic can minimize processing and storage of state by DATM mechanisms.

The advantages of DATM mechanisms are: they meet systems protection requirements by masking well defined traffic characteristics in a systematic manner; they reduce processing and storage overhead of adaptable masking schemes; they improve efficiency, system performance, and the performance of the protected applications both in the design process and by allowing dynamic adjustments; and they allow secure dynamic trade-offs between the costs of protection and application performance requirements.
Section 2 Related Work

In the past, TFC has been provided by link encryption [BAR64]. Link encryption implies physical layer encipherment, i.e., external encryption/decryption devices are placed on lines. In a system that relies on link encryption, there is an underlying assumption that each switch and its cryptographic equipment is physically protected and only trusted personnel have access to messages as they pass through the switching centers. The source and destination of traffic flowing through these switches are openly readable inside switching centers even with end-to-end encryption. The assumption of physical secure switches may no longer be taken for granted as secure communication follows the trend to open data networks where secure switches can not be assumed.

Current theoretical approaches for solutions at and above the transport layer require constraints that are unrealistic for internetworks. While they may prove useful for developing future protection systems, they do not adequately address the protection and performance requirements of systems on public networks. Proposed technologies for TFC also do not address meeting adequate levels of protection and are not scalable to internetworks.

2.1 Theoretical Security Models

It has been recognized that computer system security mechanisms that are adjustable to their environment can reduce the costs of security and improve protected systems’ performance. While it is generally acknowledged that such adjustments can create a flow of covert information about the protected applications, correcting this problem is not addressed. Additionally, within the scope of traffic analysis, systems’ protection requirements are not taken into account. The question of what precisely constitutes covert information from which protected traffic characteristics can be inferred is not answered.

A high level mathematical model that doesn’t rely on link or network layer encryption is proposed by Balaji Venkataramen to provide protection from traffic analysis on public networks [Ven94]. The model assumes a fully connected network and global synchronization between network nodes. It obtains a “spatially neutral” traffic matrix, i.e., the same volume of traffic on each link of the network, by rerouting traffic away from heavily loaded links and inserting dummy packets when necessary. It introduces a notion of “temporal neutrality”, and proposes both static and
adaptive scheduling policies that ensure that observable traffic characteristics are temporally neutral. The static scheduling policy does not introduce covert channels, but it is unresponsive to fluctuations in system load. The adaptive scheduling seeks to improve throughput and provide for quality of service guarantees, but at the expense of allowing covert channels. Venkatramen informally proposes methods to estimate the covert channel capacity and derive bounds on this capacity. The covert channels he is concerned with can transmit messages by varying temporal or spatial relationships of packets. He does not address the reduced degree of protection that may result from the proposed adaptive scheduling. There is no formal proof that the proposed system meets any system security policy. It relies on the capacity measurement approach to covert channel analysis and mentions the reservations about this approach that have been raised by Morris and Brown [BRO95] and Moskowitz and Kang [MK95], but leaves this issue for future work.

Mode Security [BRO94] does not specifically address traffic flow confidentiality. The discussion is limited to a multilevel state machine organized into sets called modes that can be applied to computer systems in general. Each mode is totally secure when considered in isolation. Information leakages occur when the machine executes a mode change decision. Brown projects that a mode security model can allow an acceptable performance/security combination to be found in the early phases of system development. Brown’s Generalized Turing Test [BRO89] puts a limit on the information flow that occurs during a mode change and is similar in function to a security model’s secure state invariant. Brown’s reservation about the capacity measurement approach is that covert channel capacity is a steady state measure of channel severity and it is possible to have severe transient conditions of covert information flow while the steady-state capacity is zero or close to it. Mode changes in secure systems is one example of this. Mode security is closely related to Weber’s n-limited security [WEB88] that takes flow-security which specifically prevents downgrading (unauthorized information flows) and generalizes it to formulate a policy that allows limited downgrading of protection.

Relaxation Security [BAD90] presents a state machine formulation definition for trusted computer systems. The security is expressed in terms of the security guarantees that a trusted system may provide. Relaxation secure systems permit dynamic incremental relaxation of security constraints to meet performance requirements. Badger’s state machine incorporates a “relaxation lat-
Relaxation security requires that a system be secure at each level of security it offers and is not concerned with the covert information released during transitions.

2.2 Related Proposed Technology
The idea of using padding and delays to mask volume and frequency traffic patterns was presented by Chaum when discussing his “general purpose untraceable mail” system. Describing a system where users send continuous or periodic fixed size batches of messages through a cascade of mixes that reroutes messages in order to mask source/destination traffic patterns, Chaum suggests that to hide the number of messages it sends, each user of the system supplies the same number of messages to each batch, some of which might be randomly addressed dummies [CHA81]. When receiving messages each user privately searches the entire batch, looking for real messages directed to it. A user recognizes messages intended for it because they are encrypted with its public key. One way that Chaum suggests to reduce the cost of the dummy messages is that users add only a random number, chosen from a suitable distribution, of dummy messages in each batch, instead of sending the maximal number of messages.

One current security product that is in the standards process is SDNS, Secure Data Network System, a proposed security protocol that is consistent with the OSI computer network model [SDNS89]. For communicating across an open data network, it provides five security services: confidentiality; data integrity; non-repudiation; authentication; and access control. SDNS has a systems approach, but does not apply it to TFC. The network layer protocol, SP3, leaves packet headers with network addresses unprotected. Some TFC does occur in the case of a LAN behind an encryption gateway with only the address of the gateway visible on the open network. This approach is limited because associations of black addresses (gateway addresses) can be tracked. The amount of protection depends to a great extent on the granularity of the communicants behind the gateway.

Another current product is NLSP, Network Layer Security Protocol, [ISO-IEC DIS11577] that operates as a sub-layer of the network layer. It provides security services that support both connection-oriented and connectionless network services. NLSP can be implemented in end systems
and intermediate systems. NLSP uses two security mechanisms that address TFC. They are traffic padding and hiding the NLSP address [MAC93]. The traffic padding procedure is used to hide the presence of NLSP user-data. The decision to use this facility is made by the local NLSP machine and is dependent on local security policy. It is transparent to the NLSP-service-user. Traffic padding is carried out by the Secure Data Transfer Protocol Data Unit (SDT-PDU), which also carries user data. There is an optional mechanism for encipherment of user data and other NLSP protocol parameters including addresses. It is claimed that this feature could counter traffic flow analysis by sending PDU to a gateway where protection is provided. Traffic analysis could yield the organizations at the end points, but not the specific end systems [MAC93]. The main disadvantage of the traffic padding procedure provided by NLSP is that it is simply a padding device. The decisions of how or when to pad are left to the local designers. There is no theory to prove the padding provides the desired protection. There is no consideration of efficiency or adaptability. DATM mechanisms could provide these important elements to NLSP and would be a useful addition to it.
3 Preliminary Results

In the course of this research I have defined traffic masking schemes as the systematic introduction of noise on connections between two network protocol entities in the form of original traffic, padding traffic, and added delays. They mask the frequency and length of communications and traffic characteristics by creating fixed or random patterns of traffic transmittal. When combined with mechanisms that obscure source-destination traffic patterns, and protected by a confidentiality service with encryption, they provide end-to-end solutions for traffic flow confidentiality between network entities. Section 3.1 discusses traffic masking schemes, DATM techniques that provide the capability to dynamically adapt to changing network conditions, and SAM techniques that use statistical methods to mask anomalies. A plan for their evaluation is also presented in section 3.1.

I have defined a formal security model of a DATM module that is represented as an abstract machine description. It precisely specifies an acceptable range of system behavior satisfying the system’s security policy in terms of the flow of covert information about original traffic characteristics. The model also specifies ranges of behavior that satisfy the protected applications’ throughput and delay requirements, and allows trade-offs between costs of security and system performance. In addition, a desired degree of security can be calculated with respect to the cost of its maintenance, and when events occur in the environment of the module that make the current degree of security too costly, or are impossible to satisfy, the model provides a means for it to be modified that can be beneficial to the system while still satisfying the system’s security policy. Section 3.2 defines a DATM module as an input/output system followed by an example of the system in a workstation on a public network. Section 3.3 defines a formal security model of a DATM module as a state machine with constraints on the transition function and a secure state invariant.

3.1 Traffic Masking Schemes

Evaluations of TFC schemes measure and compare the protection they offer, their performance and their cost. In the case of DATM mechanisms, protection is the masking of characteristics of original traffic and can be measured formally as a summary test statistic. Performance is measured the throughput and delay of the original traffic. Cost is measured as additional bandwidth from padding, and efficiency is the ratio of original traffic over masked traffic. A good DATM scheme
satisfies the performance requirements of protected applications with minimal cost. Masking schemes are categorized into classes to evaluate the protection and performance they offer. Classes are based on whether masking schemes allow the transmittal of unmasked original traffic and whether original traffic characteristics determine dynamic adjustments to schemes. Within each class, further modifications are possible, such as basing the probability of transmitting unmasked original traffic on environmental factors like the time of day or network conditions. Additional fine tuning of parameters, such as the length of intervals of adjustment, results in improved performance.

3.1.1 Categorizing and Evaluating Masking Schemes

Before discussing classes of schemes, it is useful to define some fundamentals of masking procedures and basic schemes such as fixed interval transmittal, probabilistic transmittal, and traffic partitioning. These basic schemes form the foundation of other masking schemes.

![Figure 2. Flow of a Traffic Masking Scheme](image)

In traffic masking schemes a time unit is the time allotted by the scheme to output one PDU. An interval is made up of one or more time units. Figure 2 shows the flow of a traffic masking scheme. Masked traffic is called on when a PDU is transmitted during a time unit. Conversely, off traffic is no transmittal during a time unit of masked traffic. When schemes require intervals of on traffic with probability $p$, a PDU is transmitted each time unit in an interval with probability $p$, with padding PDUs transmitted when there are no original PDUs on queue for transmittal. When
schemes require intervals of off traffic, no PDUs are transmitted and original traffic is buffered on the queue.

Basic schemes:

- **Fixed interval transmittal** is a traffic masking scheme that has fixed length intervals of on and off traffic. Each time unit is predetermined to have on or off traffic, depending on the pattern and length of the intervals. An interval of on traffic of length $n$ has $n$ time units of on traffic. The length of an interval varies between zero and infinity.

- In **probabilistic transmittal**, each time unit may have on traffic with some given probability using an appropriate distribution function. For example, if a Bernoulli distribution is used then the probability of on traffic is $p$, and the probability of off traffic is $1 - p$. The probability $p$ may be fixed or may change statically or dynamically as described below.

- With **traffic partitioning**, traffic is partitioned between masked and unmasked original traffic. The transmittal scheme switches back and forth between unmasked and masked traffic either with a given probability distribution at fixed or varying intervals of time, or dependent on some other factor such as original traffic characteristics. The probability distribution may be fixed or may change statically or dynamically, depending on the scheme.

<table>
<thead>
<tr>
<th>Table 1:</th>
<th>Masked Traffic Only</th>
<th>Masked Traffic Partitioned with Original Traffic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Independent of Original Traffic Characteristics</td>
<td>Class 1</td>
<td>Class 3</td>
</tr>
<tr>
<td>Dependent on Original Traffic Characteristics</td>
<td>Class 2</td>
<td>Class 4</td>
</tr>
</tbody>
</table>

Table 1 shows how schemes can be categorized into classes based on whether masking schemes allow the transmittal of unmasked original traffic and whether original traffic characteristics determine dynamic adjustments to schemes.
In Class 1 schemes contain no unmasked original traffic, and the probability of on traffic is not based on original traffic characteristics. This class of schemes offer the maximum protection and have no facilities for protection/performance trade-offs. One example of a Class 1 scheme is fixed interval transmittal. Other examples are link encryption [BAR64] and Chaum’s “general purpose untraceable mail system” where messages are sent in continuous or periodic fixed-sized batches, padded with dummy messages if necessary [CHA81].

In Class 2 schemes output contains no unmasked original traffic, but the probability of on traffic can be based on original traffic characteristics allowing a flow of covert information about original traffic characteristics. The degree of protection can be decreased for improved performance. Schemes that use DATM techniques are examples of class 2, 3 or 4 schemes. For instance, if there is an original PDU waiting to be transmitted, the probability of on traffic increases for a given interval. Another example of a Class 2 scheme is “high level prevention of traffic analysis,” discussed in section 2.1. The scheme defines a traffic matrix, and then manipulates it so that each cooperating host appears to be sending every other cooperating host the same volume of messages, presenting observers with a neutral traffic mix. This is achieved by a combination of rerouting messages through intermediate hosts and the use of dummy packets. The probability of on traffic is determined by a combination of original traffic characteristics and network conditions.

In Class 3 schemes allow unmasked original traffic, but the probability of unmasked traffic is not based on original traffic characteristics. The presence of unmasked traffic implies the degree of protection is decreased with a trade-off for performance. An example of a Class 3 scheme is random partitioning, a traffic partitioning scheme with a random probability of original traffic at fixed intervals, where the masked traffic scheme is in Class 1, such as probabilistic transmittal. A similar scheme is presented in [GRA92] for preventing covert channels in CPU scheduling in a multilevel systems.

In Class 4 schemes contain unmasked original traffic and either the probability of on traffic or the probability of unmasked traffic, or both, are based on original traffic characteristics. These schemes provide the least protection, but have the greatest potential for a protection/performance trade-off. Examples of Class 4 schemes are probabilistic partitioning where the probability of on
traffic in the masking scheme depends on the time of day, and probabilistic partitioning where adjustment of the probability of unmasked traffic is based on environmental factors and the probability of *on* traffic in the masked traffic scheme is dependent on the rate of masked traffic.

In the above schemes, basing the probability of *on* traffic on environmental factors, such as the time of day and conditions on the network, can lower costs usually with a protection trade-off. However some systems can tolerate lowering the degree of protection.

### 3.1.2 Maintaining Protection with Statistical Anomaly Masking

Traffic masking schemes can be attacked by statistical anomaly detection. Class 1 masking schemes are independent of original traffic characteristics and therefore not vulnerable to anomaly detection, but they also do not make adjustments to improve performance or efficiency. Class 2 schemes do make adjustments based on original traffic characteristics and therefore have the potential to be successfully attacked by anomaly detection. The patterns of traffic that they create, while different from original traffic patterns, may imply characteristics of the original traffic from which inferences could be made.

**Statistical Anomaly Detection**

There are three steps in anomaly detection mechanisms that can be anticipated and exploited when designing traffic masking schemes that have the ability to make dynamic adjustments that are based on original traffic patterns. First they collect and abstract information on system behavior. Secondly, using the collected statistics, they evolve a *profile* of normal system behavior. What is defined as normal can vary with different time frames. Thirdly, they establish boundaries that define tolerance levels for variations in behavior. Behavior that deviates significantly from the profile is considered a potential anomaly.

In anomaly detection statistics can be kept as frequency tables, means, and covariances. If the statistics of observed data are sufficiently far from expected values with respect to the statistics of the profile, the data is considered anomalous. For example, a *summary test statistic* on collected data [JV91] can reflect both the extent which recent behavior differs from the historical profile and the degree of abnormality of the recent behavior based on a historical probability distribution.
of a recently collected statistic. The larger the summary test statistic, the greater the degree of abnormality of the recent behavior.

**Designing Traffic Masking Schemes**

Important features of masked traffic that can be analyzed for information about the original traffic are burst size, inter-PDU arrival times, throughput, and delay. These features can be manipulated in a traffic masking scheme to create a profile of “normal” behavior that will accept most anticipated adaptations as normal. In addition, a system’s security policy can allow changes in the *critical value* that determines the maximum acceptable summary test statistic, depending on events in the environment of the system. A low critical value reflects the importance of protection over performance and can be raised if the priority for improved performance or efficiency increases.

Traffic masking schemes may have to do real time calculations based on events in systems environment. One advantage to using statistical analysis is that it is not necessary for schemes to keep or process extensive information about past behavior as this information is stored in the statistics. A summary test statistic can accumulate data in an additive fashion. Recent behavior can be stored as an exponentially weighted sum of changes that have occurred. This sum can be weighted so that recent history is dominant. An appropriate *half life* size can be determined by the following method: Select an appropriate interval for categorizing behavior values of traffic characteristics. Assign an initial behavior value $B_0$. If $B_n$ is the behavior value after the $n$th interval then

$$B_{n+1} = D_{n+1} + 2^{rt} \times B_n$$

where $D_{n+1}$ is the change in measured values of the characteristic from the $n$th to the $n+1$th interval, $t$ is the size of the interval, and $r$ is the decay rate that determines the half life of the behavior. If $r$ is large the value of $B$ is mostly determined by the recent past. If $r$ is small, the value of $B$ is more influenced by the distant past.

This half life measure should be short enough so that it responds to rapid changes in behavior, but at the same time long enough to reasonably reflect recent behavior. A frequency distribution for traffic behavior such as burst size can be created by the masking scheme by manipulating burst sizes within appropriately selected intervals so that an adjustment to adapt to a burst in original traffic will not appear as an anomaly.
3.1.3 Enhancing Schemes for Improved Efficiency and Performance

Basic schemes can be enhanced for improved efficiency, i.e. minimal introduced delay and optimal throughput. One of the advantages of probabilistic transmittal over fixed interval transmittal is the avoidance of fixed blocks of on traffic that can cause problems with network protocols. However, probabilistic transmittal has the potential for unbounded delay. For improved performance, schemes can be modified to avoid fixed blocks of on traffic without introducing potential unbounded delay. Such schemes fit into Class 1 when they are not dependent on original traffic characteristics and therefore do not allow a performance/protection trade-off. For example, a type of modification of probabilistic transmittal that results in Class 1 schemes is the adjustment over periods of time of the probability of on traffic. Events that could kick in adjustments are: conditions on the network; the time of day; a random coin toss; a predetermined number of time units of off traffic or a predetermined number of units of on traffic. While adjusting the probability of on traffic randomly does not in itself improve performance, it is useful when used as the masking scheme with random or probabilistic partitioning to obscure the occurrences of unmasked original traffic. The above adjustable probabilities schemes improve on the performance of fixed interval transmittal and probabilistic transmittal without a reduction in protection.

Some schemes allow a protection/performance trade-off that may be desirable for systems with stringent performance requirements and whose protection needs have room for a trade-off. Such an improvement in performance can occur when the probability of on traffic is dependent on original traffic characteristics. A Class 2 solution is a modification of probabilistic transmittal where the probability of on traffic increases if there are PDUs of original traffic waiting for transmittal and decreases if there are none, for example when the number of time units where the rate of masked traffic is less than or more than the average rate of the application that is being masked. This scheme reduces the probability of unbounded delay. While dependent on original traffic characteristics, it does not transmit unmasked original traffic. A sample of events that could kick in one of these adjustments are: a predetermined number of PDUs on queue; a random coin toss, a predetermined number of time units without on traffic; a predetermined number of time units where the rate of masked traffic is less then the average rate of the original traffic; and the presence of a PDU on the queue of original traffic approaching the schemes maximum allowable
delay.

3.2 Traffic Masking Modules
I am proposing researching new technology to mask length and frequency traffic patterns, as a modular approach to practical problems that will fit into existing environments and protocols. Composable systems are constructed out of modular components with well defined interfaces. Mechanisms designed as modular components can be placed into evaluated systems with only the interface between the component and the rest of the system needing to be tested. This facilitates placing TFC protection into established systems. Modeling the rules for composing TFC technology, potentially eliminates costly duplication of effort present in other systems. A definition of a traffic masking module is presented in section 3.2.1.

3.2.1 Formal Description
A traffic masking module, as illustrated in Figure 3, is an input/output system that can be described by the following parameters:

- $I$, the input to the system, is the traffic originating from the protected applications which is characterized by its rate of transmittal, specifically the number of protocol data units, PDUs,
input over a given interval of time.

- **O**, the output from the system, is the masked traffic that leaves the module. It is characterized by its rate of transmittal. It consists of original traffic with the possible addition of bogus packets that provide padding during periods when the rate of the masked traffic is greater than the rate of the original traffic and delays to slow the rate of original traffic when desired.

- **E**, a set of events in the environment of the module that may change **I**, the input to the system, and may cause changes to **C**, the set of performance constraints, and to **M**, the masking scheme, that are described below. These events may be generated by network protocols or by the quality of service needs of the applications.

- **T** is the time allotted by the module to output one protocol data unit (PDU) of masked traffic. During each period of length **T** at most one PDU is output. **T** can be no smaller than the transmission delay of a PDU plus allowance for other system overhead and should not exceed the minimal maximum delay requirement of the applications making up the original traffic.

- **M** is a masking scheme that may use the rate and inter-PDU delays of the input to the module to determine the rate of its output during each time interval of length **T**. The scheme may change due to events in the environment of the module.

- **C** is a set of performance constraints on **O**, the output of the module. In order to meet the service requirements of the original traffic, the system may have to satisfy a set of application dependent performance constraints on the bandwidth, throughput, and delays of output from the module. As changes occur in the module’s environment, the constraints on its output may also change.

- **P** is the security policy that must be satisfied by the traffic masking module. The security policy is concerned with the amount of information about the input to the module, the original traffic, that can be obtained from the output of the module, the masked traffic. Ideally that amount is zero, but is not always possible if the system has performance constraints. The security policy determines the security constraints on the system. The security policy may allow a trade-off between the performance needs of the application and the costs of security for improved performance that depends on its tolerance for information about original traffic characteristics in the flow of the masked traffic.
3.2.2 Example of Traffic Masking Modules

Example 1

The following example illustrates a traffic masking module placed in a workstation, WS, on a public network, with a secure IP layer providing encryption and authentication. Typical traffic through WS is email, telnet, remote login, FTP, and video or audio conferencing. As part of the protection against traffic analysis, a traffic masking module is in the network layer of the OSI network protocol suite. The module uses a uses DATM techniques for improved performance. According to the masking scheme, at each interval of length $T$ there is a probability, $0 \leq p \leq 1$, that a packet will be output to the IP layer. In the event of an increase in the rate of input causing a queue of input of length $l \geq n$ to form, $p$ increases in order to increase the rate of output and shorten the queue. After an initial increase, $p$ decreases until it is back to its optimal value when $l < n$. The system performance constraints determine the value of $n$. The security policy of the module is to allow only a small, controlled amount of information about original traffic characteristics in the masked traffic.

The traffic masking module in WS can be described as an input/output system satisfying the parameters as follows:

- $I$ is packets of original traffic.
- $O$ is packets leaving the module that consists of original packets with the possible addition of bogus packets and added delays. The output goes to the secure IP layer for encryption.
- $E$ is a set of events occurring in the environment of the module that may be generated by transport layer protocols or by the quality of service requirements of the protected applications. For example, a “timeout” signalling congestion would effect input from some transport layer protocols. Another example is the initiation of input of packets from an application previously not transmitting might cause a change in the set of performance constraints on the module’s output.
- $T$, the time allotted by the module to output one packet plus allowance for other system overhead.
- $M$ is the scheme described above for determining the probability of a packet being output during an interval of length $T$. The parameters $n$ and $p$ in $M$ satisfy the current performance con-
straints on the output. The event of a queue of input exceeding $n$ packets causes the mean rate of traffic transmittal, $p$, to increase.

- $C$ in this example is the set of average rates of the output that are determined by events in $E$.
- $P$, the security policy of the example system, allows a measured amount of covert information flow about original traffic characteristics for improved performance. How much is determined by the size of the summary test statistic in the formal security model.

### 3.3 Formal DATM Security Model

The formal security model is represented as an abstract machine description of a traffic masking module that allows dynamic adjustments. A security model must have the properties of precision, simplicity, and generality, while still being intuitively representative of the systems’ security policies. In general security models are formal design techniques that precisely specify acceptable ranges of system behavior that satisfy a given security policy. In particular, this model also specifies ranges of behavior that satisfy performance requirements, and, additionally, specifies trade-offs between costs of security and system performance. A desired degree of security can be calculated with respect to the cost of its maintenance, and when events occur in the environment that make the current degree of security too costly, or are impossible to satisfy, the model provides a means for it to be modified that can be beneficial to the system while still satisfying the system’s security policy.

#### 3.3.1 Definition of Security Model

A **DATM State Machine** is a tuple $DSM = \langle Y, I, O, S, s_0, E, C, \Phi \rangle$ such that:

1. $Y$ is an ordered set of time points that are represented as a set of positive integers.
   
   The machine starts at the first time point considered time $t_0$. A time point occurs after each interval of length $T$ defined above. The $j$th time point $t_j$ denotes that $j$ intervals of $T$ have passed since the initial time point $t_0$.

2. $I$ is a finite set of positive integers, including 0. If $i_j \in I$ then $i_j$ stands for the input to the system during the interval between time points $t_{j-1}$ and $t_j$.

   The value $i_j$ represents the characteristics of the input and output traffic, i.e. the number of
PDUs transmitted over a given duration of time.

(3) \( O \) is the set \( \{0,1\} \). If \( o_j \in O \) then \( o_j \) stands for the output of the system during the interval between time points \( t_j \) and \( t_{j+1} \).

In the set \( \{0,1\} \), 0 stands for no transmittal and 1 stands for transmittal of a PDU between time points. There is a maximum of one PDU output from the system per interval of length \( T \). With output, as well as input, no transmittal during between time points is as important information as transmittal.

(4) \( S \) is an ordered, non-empty set of tuples, \( s_j = (i_j, o_j, H_j) \) where \( i_j \in I \), \( o_j \in O \), and \( H_j \) is a sequence of states of the model that can also be denoted as \( \langle s_0, s_1, \ldots, s_{j-1} \rangle \).

\( s_j \) is the state of the machine after \( j \) minimum time units have passed, i.e., the state between time points \( t_j \) and \( t_{j+1} \). The occurrence of time point \( t_j \) causes a state transition from state \( s_{j-1} \) to state \( s_j \). \( H_j \), called a history, denotes the history of transitions of the model for \( s_j \), starting in its initial state and including all intermediate states up to \( s_j \), i.e., the ordered sequence of states that have occurred before time point \( t_j \).

(5) \( s_0 \) is the initial state, with \( s_0 \in S \).

In \( s_0 \), \( H_0 \) is the empty sequence \( \langle \rangle \). This state which occurs at time point \( t_0 \) is predefined and not a function of the input or environment of the module. The values of the other state variables of \( s_0 \) are preset.

(6) \( E \) is a set of events in the environment of the DATM module.

This set \( E \) includes events that may effect the values of the input and the set of constraints on the output of the module. \( \Omega \) is a subset of \( E \) that is made up of all events that may change the set of performance constraints on the output.

(7) \( C \) is a set of all possible performance constraints on the DATM module.

Performance constraints are concerned with the quality of service needs of the protected applications. Security constraints are concerned with the amount of information about the original traffic that can be deduced from the masked traffic and are expressed as constraints on the transition function (below).
(8) $\Phi$ is the state transition function for DSM, $\Phi : T \times E \times C \times S \times S \to 1$. $\Phi$ is defined by two components $\Phi_1 : T \times E \times C \times S \times S \to 1$ and $\Phi_2 : T \times E \times C \to C$ such that:

$$\Phi(t_{j+1}, \phi, c, s_j) = \Phi_1(t_{j+1}, \phi, \Phi_2(t_{j+1}, \phi, c), s_j) \text{ where } \Phi_2(t_{j+1}, \phi, c) = c^* \text{ if } \phi \cap \Omega \neq \emptyset, \text{ else } \Phi_2(t_{j+1}, \phi, c) = c, s_j = (i_p, o_p, H_j) \in S \text{ and } s_{j+1} \in S.$$

$\forall j \geq 0$ a state transition occurs at time point $t_{j+1}, t_{j+1} \in \mathcal{Y}$. $\phi \subseteq E$ is a set of events occurring between time points $t_j$ and $t_{j+1}$. Since occurrences of time points cause state transitions even if no other events in $E$ or input transmittals have occurred, $c, c^* \subseteq C$ are possibly the same subset, and the conditions hold that $\phi$, $c^*$ and $c$ may be empty sets.

### 3.3.2 Constraint on the Transition Function

Some events occurring in the environment of the module cause a change in the set of performance constraints on the output, which in turn may cause a change in the way the masking scheme calculates the output. Thus the output $o_j$ might be different than it would be if the event had not occurred. The transition function assures that the changes in the constraints are made before the output is calculated. A security constraint on the transition function $\Phi$ is necessary to control the amount of information flow about the original traffic caused by a change in $c$, the current set of performance constraints on the output.

The following notation precisely defines the history of the state variables. For each state $s_j \in S$ where $s_j = (i_p, o_p, H_j)$ and each $n$ where $n \leq j$, let a period of recent history starting at $s_{j-n}$ and including all intermediate states up to $s_j$, the sequence of states $\langle s_{j-n}, s_{j-n+1}, \ldots, s_{j-1} \rangle$, be denoted by $H_{n,j}$. The history of inputs $\langle i_0, i_1, \ldots, i_{j-1} \rangle$ starting at $s_0$ and including all intermediate inputs up to $s_j$, where $i_k \in s_k$ when $0 \leq k < j$, is denoted by $H_{j}\mid i$, and the history of outputs $\langle o_0, o_1, \ldots, o_{j-1} \rangle$ starting at $s_0$ and including all intermediate outputs up to $s_j$, where $o_k \in s_k$ when $0 \leq k < j$, is denoted by $H_{j}\mid o$. A period of recent history of inputs $\langle i_{j-n}, i_{j-n+1}, \ldots, i_{j-1} \rangle$ starting at $s_{j-n}$ and including all intermediate inputs up to $s_j$, where $i_k \in s_k$ when $j-n \leq k < j$, is denoted by $H_{n,j}\mid i$, and...
and a period of recent history of outputs \( \langle o_{j-n}o_{j-n+1}...o_{j-1} \rangle \) starting at \( s_{j-n} \) and including all intermediate outputs up to \( s_j \), where \( o_k \in s_k \) when \( j-n \leq k < j \), is denoted by \( H_{n,j|o} \).

Then if \( s_j = (i_j, o_p, H_j) \) where \( s_p, s_{j-1} \in S \), and \( p \) is a probability function the constraint on the transition function \( \Phi \) is \( |p(o_j|H_{n,j|o}) - p(o_{j-1}|H_{n,j-1|o})| \leq \varepsilon \) where \( \varepsilon \) is sufficiently small.

The above constraint guarantees that when a transition from state \( s_{j-1} \) to state \( s_j \) occurs, the probability of the recent history of outputs of \( s_j \) is sufficiently close to the probability of the recent history of outputs of \( s_{j-1} \) considering a period of recent history where the value of \( n \) is determined by the system’s security policy. Thus even if an event in the environment of the module causes a change in performance constraints on output and the resultant output \( o_j \) may be different than the output if the event had not occurred, the constraint assures that the probability of the of the recent history of outputs will be sufficiently close to the probability of the expected recent history of outputs.

### 3.3.3 The secure state invariant:

**DATM** is secure if and only if for each state \( s_j \in S \) where \( s_j = (i_j, o_p, H_j) \), and \( p \) is a probability function, then \( |p(o_j|H_{n,j|o}) - p(o_{j-1}|H_{n,j-1|o})| \leq \delta \) where \( \delta \) and \( n \) are determined by the system’s security policy.

The secure state invariant states that the probability of an output \( o_j \) considering the recent histories of previous inputs \( H_{n,j|l} \) and outputs \( H_{n,j|o} \) is close enough to the probability of the output \( o_j \) considering only the recent history of previous outputs \( H_{j|o} \) so that a sufficiently small amount of information can be found out about the recent history of the inputs when only the outputs are observed. When considering only recent output history, the output’s apparent statistical dependence on recent input history is small enough to satisfy the system’s security policy.
3.3.4 Example of Security Model

Example 1
The example of the module described in section 3.3.2 is modeled by specifying its state machine tuple $DSM = \langle T, I, O, S, s_0, E, C, \Phi \rangle$:

1. $\Upsilon$: a set of positive integers denoting an ordered set of time points. The $j$th time point $t_j$ denotes that $j$ intervals of $T$ have passed since the initial time point $t_0$. In this example, $T$ is the time allotted by the module to output one packet plus allowance for other system overhead.
2. $I$: The number of packets $i_j$ arriving at the module between sequential time points $t_j$ and $t_{j+1}$ can be modeled as a Poisson process which is typically used to model the number of arrivals over a given interval, particularly if the arrivals are from a large number of independent sources. It has been argued that samples of network traffic such as in this example do not form a Poisson process as the resulting sequence of random variables are not independently distributed because the arrival of packets during an interval signals the arrival of packets during the following interval [JR86]. However, if the size of the interval is sufficiently small, i.e., smaller than the minimum maximum delay of the applications, then the arrival of a packet will not necessarily signal the arrival of other packets, and the Poisson distribution is appropriate because of the small number of arrivals each interval. Another distribution that is appropriate for the input of this model is the Pareto distribution that is useful to fit a distribution to observed data.
3. $O$: The output is determined by the masking algorithm $M$. Based on $M$, the probability distribution of $O$ is described as a Bernoulli distribution where $p$ is the probability of transmittal, i.e. $o_j = 1$ in state $s_j$, $0 \leq p \leq 1$.
4. $S$: is a set of tuples $s_j = (i_p, o_j, H_j)$ calculated using the above Poisson or Pareto probability distribution for $i_k$ and Bernoulli distribution for $o_k$ in the state transition function $\Phi$. $H_k$ is determined by the values of $i_k$ and $o_k$ for $0 \leq k < j$.
5. $s_0$: the initial state can be defined as $(0, 0, \langle \rangle)$
(6) \( E \): a set of events in the environment of the model. These events may be generated by transport layer protocols or by the quality of service needs of the applications and may or may not effect the values of \( i_j \) and \( o_j \). For example, a “timeout” signalling congestion has an effect on input to the module from some of the transport layer protocols. The event of a timeout is based on the masked traffic, i.e. the output and a response to a timeout does not compromise security. The event of an increase of application input belongs to \( \Omega \), a subset of \( E \) made up of all events that may change the set of constraints on the output.

(7) \( C \): a set of constraints that includes all possible performance constraints on the DATM machine. In this example the constraints are average rates of traffic transmittal.

(8) \( \Phi \): \( \Phi (t_{j+1}, \phi, c, s_j) = \Phi_1 (t_{j+1}, \phi, \Phi_2 (t_{j+1}, \phi, c), s_j) \) where \( \phi \subseteq E \) is the set of events that have occurred between \( t_j \) and \( t_{j+1} \). If \( \phi \subseteq \Omega \), where in this case \( \Omega \) is the event of increased application input sufficient to cause a queue of input packets of length \( l \geq n \), then \( \Phi_2 (t_{j+1}, \phi, c) = c^* \), where \( c^* \) is a larger rate of traffic than \( c \), and \( \Phi (t_{j+1}, \phi, c^*, s_j) = s_{j+1} \) where \( s_{j+1} = (i_{j+1}, o_{j+1}, H_{j+1}) \). The state variable \( i_{j+1} \) is determined by Poisson or Pareto distribution. The state variable \( o_{j+1} \) is determined by a Bernoulli distribution where, \( p \), the mean output, is a function of \( c^* \), the new rate requirement on the output. The value of variable \( H_{j+1} \) results from concatenating \( s_j \) to the end of \( H_j \).
Section 4 Research Plan

My goal is to formally model a traffic masking module in a packet network protocol environment, develop an implementation design, and verify the correctness of designed solutions by formally proving the correspondence of the implementation design to the formal model. I aim to develop technology that will integrate into existing security architecture as a solution to provide traffic flow confidentiality in open internetworks.

5.1 DATM and Security Models

I will continue to develop traffic masking schemes that use DATM techniques to make dynamic adjustments for performance and efficiency, but still maintain a satisfactory degree of security, by using SAM techniques to control the flow of covert information about original traffic that is introduced by the dynamic adjustments. I will continue to refine the security model to assure that it satisfactorily expresses traffic masking modules’ security requirements. It must maintain the properties of precision, simplicity, and generality, while still being intuitively representative of the systems’ security policies. I need to prove that the constraints on the transition function are sufficient and necessary. Constraints are particularly important in this model since it is concerned with specifying properties of transitions, and constraints are concerned with the relationship between values in two consecutive states, before and after each transition.

I plan to look into modifying “relaxation lattice” technology [HW89] for use in the model. A relaxation lattice can be used as a formal design technique for specifying ranges of behavior of implemented systems. Defining the set of constraints $C$ in the model as a lattice can provide a lever for the precise adaptation of the masking process while remaining within a system’s security policy.

5.2 Implementation and Evaluation of Traffic Masking Modules

I have defined a formal description of a traffic masking module as an input/output system and plan to implement this module in an existing network protocol environments protected with a secure IP layer. It will be implemented with modular components with well defined interfaces.
Interaction between Traffic Masking and TCP

It is desirable to place a traffic masking module below the transport layer for several reasons. It is more efficient in terms of additional consumed bandwidth to have one module per host or gateway. It can interface with the secure IP protocol and does not have to provide different interfaces for each application and each transport layer protocol. However, there are definite challenges that must be addressed when a traffic masking module is situated in the IP layer. The module must cooperate with underlying transport protocols to satisfy performance requirements of applications. If the module is unaware of TCP, it will respond to congestion that occurs on the links between sender and receiver. TCP slow start is designed to initiate data flow across a connection. It is used with congestion avoidance to avoid congestion that could drastically reduce the throughput of TCP connections [JAC88]. It assures that the rate that new packets are injected into a network is the same as the rate at which the sender receives acknowledgments (ACKs) of previously sent packets [STE94]. A traffic masking module that is implemented below TCP should be aware of TCP slow start in order to react to congestion.

To provide adequate protection, a masking mechanism must also mask the commonly ignored back channels generated by TCP protocols for both interactive and bulk data flows. In order for TCP to provide a reliable flow of data between two hosts, a constant flow of acknowledgments are sent between the two hosts. Interactive traffic such as rlogin and telnet that is frequently considered to be highly one way produces back channels that consist of at least one segment returned to the client for each keystroke sent to the server. Unless it is masked, this back channel is a source of covert information about connections. When transferring blocks of data, TCP uses a sliding window protocol for flow control that allows the sender to transmit multiple segments before it has to stop and wait for an acknowledgment. Even though there is not an ACK sent for each segment, back channels consist of periodic ACKs for data sent. Similar to the back channels of interactive data flow these periodic acknowledgments are another source of information about the connection if not masked. The implementation implications are that both ends of a connection must be protected with a masking mechanism.
Implementation Plans

I plan to implement prototypes of traffic masking modules in a controlled network environment with secure IP as illustrated in Figure 4. The module will be will interface with one of the available prototypes of secure IP such as swIPe [IB93]. A swIPe implementation described in [IB93] was made available on June 15, 1994 via anonymous ftp from ftp.csua.berkeley.edu. swIPe provides message privacy, message integrity, source network address authentication, replay protection and inner packet padding to assure all packets have a uniform size. The prototype module will interface with the secure IP protocol. For outgoing traffic secure IP mechanisms are usually placed immediately above the IP layer, where traffic targeted for protection is routed to the secure IP for encapsulation using an IP-inside-IP protocol. A secure IP packet is actually an IP packet with a normal IP header whose payload, which is usually encrypted, is the original packet encapsulated with a secure IP header. The packet format is shown in Figure 5.

Before encapsulation packets from outgoing traffic targeted for masking are rerouted to the traffic masking module. This traffic is recognized by its destination address. This is possible because TFC occurs between some combination of host and gateway (host-host, host-gateway, etc). Once
received by the module they are buffered on a queue and then transmitted to the secure IP protocol according to the masking scheme mechanism with the possible addition of bogus packets for padding. The packets may also be delayed in the buffer for measured periods of time depending on the masking scheme’s algorithm.

To prevent creating a covert channel of information about the original traffic, TCP must be aware of the padding. One solution is to place the padding mechanism in the application layer, but the composability of this solution needs to be studied.

Incoming traffic also goes to secure IP before going to IP processing. The traffic masking module for incoming packets is immediately above secure IP and interfaces with IP from below. Secure IP decapsulates and decrypts the packets. The module discards any padding packets and sends original incoming packets to the IP layer for further processing.

![Figure 5 Components of a secure IP packet](image_url)

**Figure 5 Components of a secure IP packet**

**Evaluation**

Evaluations of TFC schemes measure and compare the protection they offer and their performance. In the case of DATM schemes, protection is the masking of characteristics of original traffic and can be measured formally as a summary test statistic. Performance is end-to-end delay and
throughput of the original traffic. A traffic masking module must satisfy the protection and performance requirements of the protected applications. Good masking schemes satisfy the performance requirements of protected applications with minimum bandwidth. They must consume \textit{at most} enough bandwidth to meet the protection, throughput, and delay requirements of the original traffic. If they introduce delays, they still must meet the delay and throughput requirements of the original traffic. They must satisfy the performance requirements of protected applications with minimum cost. Therefore, I will evaluate prototype modules with schemes from each of the classes discussed in section 3.1, measuring the delay and throughput of the original traffic, and the resulting total bandwidth, in order to evaluate the cost of masking and the performance of the underlying original traffic. I will evaluate how the prototype modules interact with TCP/IP and plan to investigate the effects of various masking schemes under controlled work loads, on a wide-area network emulator such as \textit{hitbox} [ADLY95].

Important evaluation criteria are:

\begin{itemize}
  \item Comparison of applications’ performance when masked and also unmasked to evaluate how well schemes meet their delay and throughput requirements.
  \item Comparison of performance and bandwidth consumption of schemes from different classes. In order to assess the offered performance/protection trade-offs at varying degrees of protection, the bandwidth, delay, and throughput of the different classes is measured and compared.
  \item Comparison of performance of schemes from each class as they are fine tuned and/or enhanced for improved efficiency. In this case, the trade-off is usually between performance and consumed bandwidth as reduced bandwidth can come with increased delay or reduced throughput.
\end{itemize}
Section 5 Research Contributions

My thesis is that currently employed and recently proposed technology for traffic flow confidentiality (TFC) do not adequately address the costs of protection or applications’ performance requirements. They also do not consider system protection needs and the reduction in protection that result from dynamic adaptation. This research develops new dynamic adaptable traffic masking (DATM) techniques for secure TFC mechanisms that have the capability to dynamically adapt to changing network conditions, and defines a formal security model that precisely specifies the security requirements for DATM mechanisms within an internetwork environment.

This research presents solutions for traffic masking that meet users’ protection needs while minimizing the costs of protection and still meeting application performance requirements. It develops DATM techniques for schemes that have the capability to dynamically adapt to changing environmental conditions in order to meet the throughput and delay requirements of the original traffic and to reduce the cost of padding. The proposed research includes statistical anomaly masking (SAM) techniques that use statistical methods to detect and prevent statistical anomalies that contain information about original traffic characteristics.

The proposed security model precisely specifies system protection requirements and specifies the relationships between costs of protection and original traffic performance requirements allowing dynamic trade-offs between the two while still satisfying system protection needs. Using correctly identified traffic characteristics, the model allows trade-offs between costs of protection and original traffic performance requirements that meet system security policies.

The advantages of DATM mechanisms are: they meet systems protection requirements by masking well defined traffic characteristics in a systematic manner; they reduce processing and storage overhead of adaptable masking schemes; they improve efficiency, system performance, and the performance of the protected applications both in the design process and by allowing dynamic adjustments; and they allow secure dynamic trade-offs between the costs of protection and application performance requirements.
The implementation of a secure DATM module in a UNIX network environment can demonstrate the correctness of designed solutions by illustrating their correspondence to the formal model. In addition, the planned implementation will allow evaluation of the DATM mechanisms, measuring the protection they offer, their performance, and their bandwidth consumption. Schemes are evaluated by measuring delay and throughput of underlying original traffic, and the resultant total bandwidth of the output. Masking schemes should cooperate with underlying transport protocols to satisfy the performance requirements of the application.

The major contributions of this research are techniques for traffic masking mechanisms that meet user’s security needs with minimal cost while allowing dynamic adjustments for performance considerations, a formal security model that precisely specifies the security requirements for these mechanisms, and an implementation design that corresponds to the formal model. The developed technology will integrate into existing security protocols as part of a solution to provide TFC in open internetworks.
References


[BRO95] Randy Brown, Private Communication on the Relevance of Channel Capacity to Covert Channel Analysis, 1995


