# True Anonymity Without Mixes

Carlos Molina–Jiménez* and Lindsay Marshall

Department of Computing Science, The University of Newcastle upon Tyne, U.K.
{Carlos.Molina, Lindsay.Marshall}@newcastle.ac.uk

**Abstract.** Anonymity is an essential part of social structures. In the non–electronic world there are several services that are based on anonymous interactions between individuals. The migration of these services to the Internet world is unfeasible without the provision of anonymizers to guarantee anonymity.
Anonymizers based on mix computers interposed between the sender and the receiver of an e-mail message have been used in the Internet for several years by senders of e–mail messages who do not wish to disclose their identity. Unfortunately, the degree of anonymity provided by this paradigm is limited and fragile. First, the messages sent are not truly anonymous but pseudo–anonymous since one of the mixes, at least, always knows the sender's identity. Secondly, the strength of the system to protect the sender's identity depends on the ability and willingness of the mixes to keep the secret. If the mixes fail, the sender's anonymity is reduced to pieces.
In this paper, we propose a novel approach for sending truly anonymous messages over the Internet where the anonymous message is sent from a PDA which uses dynamically assigned temporary, non–personal, random IP and MAC addresses. Anonymous e–cash is used to pay for the service.

## 1 Introduction

Probably the single biggest fallacy about the Internet is that it is anonymous. Most computer users know pretty well how to surf the web and retrieve information, yet what most of them ignore is that while they visit web pages, they can be profiled by web page owners. Because of the design of the Internet, it is relatively easy for web pages' owners to keep log files to gather information about visitors and to convert such information into statistical data. And if you are a web page owner that does not want to know anything about log files, you can delegate this job to on–line companies like NBC Internet Inc. (NBCi) which in return for your personal data and a share of the collected statistics will monitor your web page for free [1]. Having installed a counter (a reference to NBCi's CGI script) in your web page, NBCi offers you sensitive information like the operating systems and the browsers used by visitors, time of visit, visitors' country, and the full host name of the last 100 visitors. Notice that this information is provided as bar graphics and is ready for use in marketing campaigns and all without the visitor being aware of it.

From the above discussion it follows that it would be desirable for web surfers to visit web pages anonymously. Moreover, there are other applications such as expression of political views, assistance with embarrassing diseases and electronic transactions where users do not wish to disclose their identity.

As a response to this concern, intensive research has been conducted in this direction which has resulted in several published papers suggesting solutions

* PhD student supported by The National Autonomous University of Mexico

to the problem. However, all proposals known so far are based on the use of mixes which were first introduced in the early 80s [2]. We argue that mix–based anonymizers (see section 2.1) are not a satisfactory answer to the problem. Therefore, in this paper we suggest an innovative and simple approach for sending truly anonymous messages which does not use of mixes, nor does it uses a home IP address assigned by the home network operator to identify the sending device. The anonymous message is sent from a mobile device identified by a non–personal, temporary, random identifier assigned by the Mobile Support Station (MSS). Anonymous e–cash is used to pay for the call.

As we will show later on, the idea is simple, clear, easy to understand, validated and implemented; surprisingly, probably because of its simplicity, it has been overlooked and has not been studied before.

It is worth noticing that in this paper we assume that Bob, the mobile user, is in possession of a Personal Digital Assistant (PDA), yet Bob's PDA represents any pocket–sized electronic device equipped with computational power and wireless communication facilities, i.e. a cross between a computer and a mobile phone which is called a *Mobile Internet Device* or a *Wireless Internet Device* by some vendors. Moreover, even if in section 5 we talk especifically of PDAs that comply with the IEEE 802 standard the main ideas of not using personalized identifiers to provide true anonymity are valid for any other network in the market.

## 2 Mobile hosts with and without home IP addresses

Several approaches has been suggested to provide host mobility in the Internet [3, 4, 5, 6]. All of them are grounded on the assumption that when a mobile host, Bob's PDA for example, is away from its home network it still needs to access data and services (personal files, local data bases, local web pages, forward messages, etc.) available only in or through his home network. Because of this, Bob's PDA must be assigned a permanent IP address in its local network which remains constant regardless of its current physical and logical location. Taking into account that the IP address is uniquely assigned to the PDA and assuming that a message received from the PDA has not been meddled with, it is not difficult to see that the source IP address contained in messages coming from Bob's PDA uniquely identifies his device. A side effect of this is that the IP address can be used by the people Bob exchanges messages with to trace Bob's identity. The problem is not new. Several solutions have been proposed, however, all of them are grounded on the use of mix computers.

### 2.1 Do mixes provide true anonymity?

Currently, the only possibility for sending anonymous messages from an office or home workstation computer is to connect to an Internet anonymizer before sending the message. There are several on–line companies offering anonymous services (see [7], for example), however, all of them base their services on the use of mixes.

A mix is a computer interposed between the sender and the receiver. Between Bob's PDA and Alice's computer for example. The job of the mix is to receive the message from the sender, delete the sender's identity (her IP address and her e–mail address for example), and forward the message to the receiver. In practice, to strengthen the security of the system, several computer are used to form a chain of mixes. If this is the case, a message sent by the sender is bounced from one mix to another until eventually one of the mixes delivers the message to its final destination.

Unfortunately, this solution is not satisfactory since the strength of the system depends on the the ability and willingness of the mixes to keep secret the sender's identity, consequently, the system is fragile and can be broken by subversion or a conspiracy of all the mixes.

Another flaw of mix–based anonymizers is that their degree of anonymity is limited since there are no means of hiding the IP address of the sender; one of the mixes, at least, will always know it. The problem here is that the sender is a computer uniquely identified by an IP address. Trying to send an anonymous message from an IP addressed computer is analogous to trying to make an anonymous call from a home telephone line by using the Calling Line Identification Blocking service (the 141 number in the U.K.). The calling number is hidden from the receiver by preceding the dialed number with the digits 141, but it is not hidden from the carrier, nor from anybody who has the means of persuading the carrier to disclose it nor from a miscreant with enough knowledge and resources to break the carrier's computer where the number is stored. To put it simply, the anonymous messages sent by using mix–based anonymizers are not anonymous but only pseudo–anonymous.

It is true that the strength of a mix–based anonymizer can be improved by increasing the number of mixes, however, this makes the whole system more complicated and at the end of the day the risk of subversion or conspiracy is always present. Likewise, the flaw that one of the mixes, at least, knows the sender's identity is not fixed. It seems obvious that incrementing the number of mixes and making it more complicated does not lead to true anonymity.

### 2.2 Lack of anonymity in postpaid communication services

One of the attractions of having a mobile device is that Bob can freely travel all over the world (where coverage for his PDA is provided) enjoying continuity of his communication service, sending and receiving messages and being billed by a single bill at home at the end of the month and on a postpaid service basis.

For this to be possible, regardless of its current geographical location and before full access to the Internet resources is granted, Bob's PDA must contact its home network and authenticate itself so that his home network operator can collect all charges that Bob incurs.

For the sake of authentication, Bob's PDA can use any hardware (the serial number of his CPU for example) or software (an assigned number, for example) identifier previously agreed upon with the operator of its home network, as long as this identifier unmistakenly leads to Bob's home account. A side effect of this scheme is that Bob's identifier can readily be used by the operator of his home network, to find out all about Bob's whereabouts (his geographical location, services used, and so on).

### 2.3 True anonymity in prepaid communication services

The arguments outlined in in section 2.2 about the need of a home IP address are certainly justifiable, however, there are situations where Bob might want to access Internet services under strict anonymity; i.e without disclosing his identity neither to Alice (the receiver of his message) nor to the operator of his home network or to the operators of the communication infrastructure located between him and Alice. If this is the case, the use of a PDA identified to its home network by an IP address is certainly not suitable and not necessary as long as the PDA accesses Internet services that do not require support from its home

network; examples of such services are reading local news, posting of messages to electronic lists, e–mailing anonymous messages and so on.

Probably the simplest way for accessing Internet services under complete anonymity is the use of a prepaid PDA. Although they are not yet in the market, a prepaid PDA would work as a prepaid GSM phone does: Bob would get his prepaid PDA in the supermarket and without the need to sign any contract at all with the communicator provider he would load it with a certain amount of money from which the cost of his calls is deducted. When his prepaid credit runs out he would recharge his device by purchasing a top–up anonymous scratch lottery–like card similar to the ones used by current prepaid phones [8, 9, 10]. Since Bob does not need to give away any personal data to buy his device and since he can recharge it anonymously, a prepaid PDA can be used for accessing Internet services under complete anonymity. It can be used for example for sending truly anonymous messages.

### 2.4 Anonymous and non–anonymous Internet access

From the discussion presented in sections 2.2 and 2.3 one can conclude that the provision of either non–anonymous and anonymous Internet access is a simple question as long as Bob uses two PDAs: one for each service. Although PDAs are pocket–sized, light, and easy to carry, using different a PDA for accessing different services in neither optimal nor practical. The question that immediately arises here is whether it would be possible to have true anonymity from a postpaid PDA?. In other words, can Bob use his IP addressed PDA for sending true anonymous messages? Fortunately, the answer is yes. The crucial idea here is to make Bob's PDA communicate with the MSS in two different modes: non–anonymous and anonymous.

In non–anonymous mode the PDA communicates with its MSS in the traditional way, i.e. it uses its home IP address and authenticates to its home network before being accepted by its current network. Whenever Bob's wishes to protect his identity, he switches his PDA into anonymous mode.

In anonymous mode Bob's PDA does not use its IP home address, neither does it authenticate to its home network; it does not need to contact its home network at all; it might contact it (under the cover of its anonymous hood) but only if Bob wishes to and its home network accepts anonymous visitors.

To communicate with its current MSS in anonymous mode, Bob's PDA uses a non–personal, temporary, random identifier (TmpId) assigned by the MSS on a per–communication session basis. The TmpId can be any number, such as a dynamic and temporary Internet address assigned by means of the Dynamic Host Configuration Protocol (DHCP) [11] for example.

At this stage one can ask what motivations would a MSS have to provide anonymous communication services to Bob. The answer is money. The MSS will not be bothered about Bob's identity as long as he or somebody else pays for the service. This issue is discussed in the next section.

## 3 Anonymous payment for the communication service

The method of payment for the anonymous communication services offered by the MSS play an important part in the algorithm for anonymous communication presented in section 4. However, since this issue is not intrinsic to the algorithm it is worth discussing separately.

There are two possible cases Bob can be faced with when he comes to the MSS to request an anonymous communication service. First, the MSS can offer free anonymous communication services to Bob. This mean that the owner of the MSS offers free of charge anonymous communication to PDA users in return for advertisements or that somebody else will be paying for Bob's call; the government or the called party for example. Why, how and whoever pays the MSS for the service is irrelevant to our algorithm. We will close this case by saying that a marketing company who is carrying out an anonymous survey is paying the MSS for all anonymous messages received. Secondly, following the general rule which states that the calling party pays, the MSS can charge Bob on–line on a pay–for–time–used basis. If this is the case, the MSS must support a mechanism for anonymous payment. Bob can use an anonymous prepaid card (see section 2.3). Alternatively, Bob can use anonymous e–cash [12, 13].

Since the use of anonymous e–cash to pay for the anonymous communication service is the most general case and because Bob might need an anonymous method of payment to pay for other services apart from the MSS's, we consider the use of anonymous e–cash in our approach presented in the next section and originally introduced in [14].

## 4 Sending true anonymous messages

In our system shown in Fig. 1, Bob is the anonymous sender, Alice is the recipient of the anonymous message on her work station (WS), and Doug is the owner of the MSS and offers communication services on a pay–for–time–used basis. Clare is a bank owner and offers support for anonymous e–cash payments to her account holders (Bob and Doug). Finally, Ebe is another PDA user. $Kpu$ is short for public key as $Ks$ is for secret key. The algorithm works as follows:

1. Bob turns on his PDA and learns the $Kpu$ of the MSS by listening to its advertisement
2. The PDA creates a $Ks$, encrypts it using the $Kpu$, and sends it to the MSS for approval, waiting $t$ units of time for a reply
3. The MSS checks that the $Ks$ suggested by Bob is correct and not in use. If so, it creates a TmpId for Bob, encrypts it using the $Ks$, and sends it to the PDA as a reply. If the $Ks$ suggested by Bob is incorrect, the MSS does not reply. If it is correct but has been assigned to an existing user, the MSS does not reply to Bob and additionally asks the user of the existing $Ks$ to renew his $Ks$. After $t$ units of silence, Bob can try again. The approved $Ks$ is used then to encrypt and decrypt messages between the PDA and the MSS until either the end of the session or until it has to be renewed. Messages encrypted with Bob's $Ks$ can be overheard by Ebe but they will be ignored
4. Bob sends an anonymous e–coin to Doug to pay for the communication session. Doug consults Clare about the authenticity of the coin before accepting or refusing it
5. If Bob wishes to anonymously e–mail Alice, he appends the message body to Alice's address, encrypts the result with $Ks$, and sends it to the MSS
6. The MSS decrypts the message, encrypts the enclosed message body together with Bob's TmpId using Alice's $Kpu$, and forwards it to her
7. Alice has no means to discover the identity of the TmpId holder. Yet, she can reply to Bob by addressing her response to the MSS and including Bob's TmpId
8. Bob's session ends when he turns off his PDA, leaves his current MSS, or his MSS times-out his session

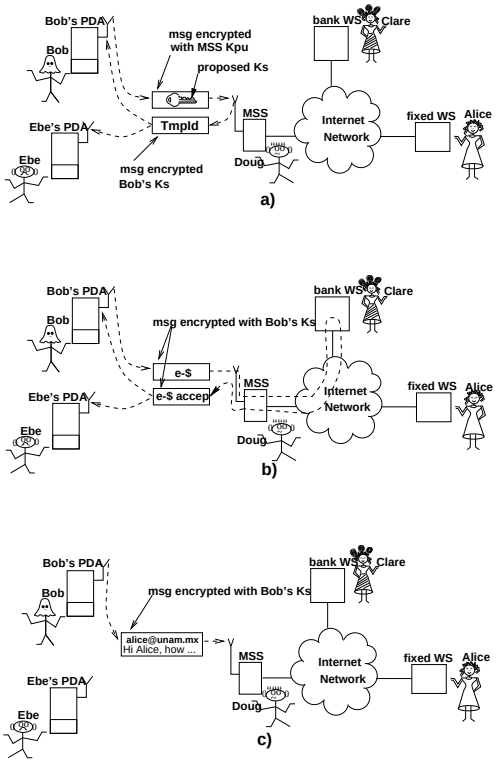


**Fig. 1.** Anonymous and confidential call from a PDA

To stop the MSS from reading the message addressed to Alice, Bob can encrypt the message body with Alice's $Kpu$. Likewise, if he does not want the MSS to read Alice's reply, he can create and append a $Ks$ to Alice's message body and ask Alice to encrypt the message body of her reply.

## 5 Can the MAC address reveal Bob's identity?

When communication takes place among multiple stations on a common communication channel (an Ethernet cable, a radio frequency channel, etc.) there is a need to identify both the sender and the receiver uniquely. In LANs that comply with the IEEE 802 standard, a station is identified by a string of either 16 or 48 bits assigned to its network interface controller (NIC) [15, 16, 17] and called the Media Access Control address (MAC address for short).

16 bit MAC addresses can be administered locally only, conversely, 48 bit MAC addresses can be administered either locally or globally. The main difference between local and global MAC address administration is that in the first case the addresses assigned to a NIC are determined by the administrator of the LAN, who is responsible for guaranteeing that no two NICs in his LAN have the same MAC address at the same time. On the other hand, globally administered addresses are assigned by the IEEE standardization body in coordination with the NIC's manufacturer. As their name implies, globally administered MAC addresses are globally unique. This means that no two NICs in the world can have the same MAC address.

The 802 standard does not encourage any specific implementation of the MAC address, however, in practice this address is normally stored in what is

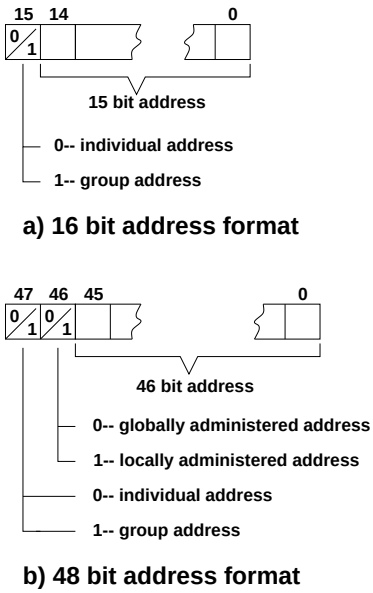


**Fig. 2.** Address field format of a IEEE 802 standard.

called the unicast address register of the NIC. For locally administered addresses the value of the 46th bit in the unicast address register is 1. Conversely, for globally administered addresses, this bit is set to 0 (see figure 2–b). At any time the actual MAC address of a NIC is determined by the contents of this register which can be read and written by the device driver software. Normally, at initialization time, the device driver software loads the unicast address register with a default value. Since 16 bit MAC addresses are always administered locally, the default value is always read from a set of switches configured manually by the LAN administrator. Similarly, the default value for a 48 bit MAC address administered locally is read from a set of switches configured manually by the LAN administrator. However, if the 48 bit MAC address is administered globally, the default value is read from a ROM chip embedded in the NIC.

During initialization, the default value of the MAC address can be ignored by the device driver software, so that the unicast address register can be loaded with a different value determined by the LAN administrator and under his responsibility [18]. Also, the contents of this register can be changed at any time (see figure 3).

The use of globally administered MAC addresses significantly reduces the LAN administrator's work. To initialize a NIC with a globally–unique 48 bit MAC address, it is enough to load the unicast address register of the NIC with the value read from the ROM chip.

The IEEE 802 standardization body is in charge of administering the $2^{48}$ address space. Upon request, it issues to NIC manufacturers what is called an Organization–Unique Identifier (OUI) which is a string of 24 bits to be used for the left 24 bits of the MAC address wired in the ROM of all NICs manufactured by him. The right 24 bits are assigned by the manufacturer.

### 5.1 Tracing Bob through the MAC address of his PDA

A side effect of the scheme used for assigning global MAC addresses is that by looking at the left 24 bits of the source address in a MAC frame, the receiver of

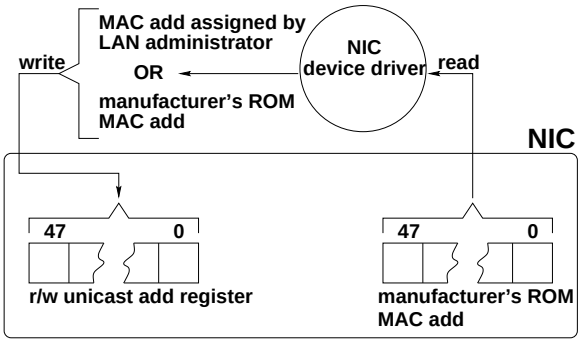


**Fig. 3.** Initialization of the unicast address register.

the frame can always learn the name of the manufacturer of NIC being used by the sender's computer.

In the algorithm presented in section 4 we assumed that messages exchanged between the MSS and its PDA are sent to a broadcast address, consequently, no MAC addresses were involved in the communication. However, it might be the case that Bob's PDA is required to use a MAC address to communicate with the MSS. If this is true, as stated above, the use of a globally administered MAC address would give the MSS the opportunity to trace Bob's identity with the help of the NIC's manufacturer and the PDA's seller. This would be certainly possible if the latter keeps records that link Bob to the PDA he bought.

A possible solution to this problem would be for the PDA to avoid using a globally unique MAC address when sending anonymous messages; for this purpose it can use a locally administered MAC address.

### 5.2 Random MAC addresses

Locally administered MAC addresses are assigned to his NICs by the LAN administrator. As stated in section 5, at run time what really matters is the value stored in the unicast address register of the NIC. If the LAN administrator decides to use locally administered MAC addresses, this value can be changed by him at will as long as it complies with the format presented in figure 2 and each NIC has a unique address within the extend of his LAN.

On this basis, to send anonymous messages, a PDA can be assigned a temporary, non–personal, random MAC address (TmpMAC) to communicate with the MSS at the MAC level.

The TmpMAC can be negotiated in the same way the PDA negotiates its TmpId. The main idea here is that at the beginning of the communication the PDA communicates with the MSS over a well–known group address, a broadcast address for example (see figure 2). Next, when the TmpMAC is assigned to the PDA, the PDA's device driver software configures its NIC by loading this TmpMAC into its unicast address register. Needless to say, the TmpMAC address is valid for the duration of the session.

## 6 Conclusions

In this work, we argued that anonymizers based on mix computers and similar systems that rely on a third party interposed between the sender and the receiver cannot provide true anonymity; moreover, they provide only fragile anonymity.

To address this issue, we presented a new approach for sending truly anonymous and confidential messages from a PDA served by an MSS. In our paradigm we send anonymous messages from a PDA which is not identified to its home network by a permanent home IP address but by a random, temporary, non–personal dynamically assigned identifier. Similarly, it uses a random, temporary, non–personal dynamically MAC address instead of the global one embedded in its NIC. Anonymous e–cash is used to pay the MSS for the communication service. A validation model for the algorithm has been written in PROMELA code and its basic correctness properties correctness were validated by using SPIN [19].

Aside from its obvious advantages, anonymity has several serious and negative side effects that make its deployment in the Internet a controversial issue. There are strong arguments for and against it. We believe that before saying that anonymity is good or bad, legal or illegal, we have to bring it into practice and test it rather than blindly approve or banish it.

## References


[1] Inc. NBC Internet. Xoom counter. http://counter.xoom.com/, July 2000.

[2] David L. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2), 1981.

[3] John Ioannidis, Dan Duchamp, and Jr. Gerald Q. Maguire. IP-based protocols for mobile internetworking. In *SIGCOM'91 Conference. Communications Architecture and Protocols*, Zurich, Swizerland, September 3-6 1991. ACM.

[4] Fumio Teraoka, Yasuhiko Yokote, and Mario Tokoro. A network architecture providing host migration transparency. In *SIGCOM'91 Conference. Communications Architecture and Protocols*, Zurich, Swizerland, September 3-6 1991. ACM.

[5] Charles Perkins. Providing continuous network access to mobile hosts using TCP/IP. *Computer Networks and ISDN Systems*, 26(3), 1993.

[6] Hiromi Wada, Takashi Yozawa, Tatsuya Ohnishi, and Yasunori Tanaka. Mobile computing environment based on internet packet forwarding. In *Proceeding of Winter Usenix*, San Diego CA, January 25-29 1993. USENIX Association.

[7] Inc. Anonymizer. Anonymizer. http://www.anonymizer.com/main.html, April 1998.

[8] Yi-Bing, Ming-Feng Chang, and Herman Chung-Hwa Rao. Mobile prepaid phone services. *IEEE Personal Communications*, 7(3), January 2000.

[9] Michael Collins. Telecommunications crime —part 2. *Computer & Security*, 18(8), 1999.

[10] Siegmund M. Redl, Matthias K. Weber, and Malcolm W. Oliphant. *GSM and Personal Communication Handbook*. Mobile Communications Series. Artech House, 1998.

[11] Douglas E. Comer. *Internetworking with TCP/IP. Principles, Protocols, and Architecture*, volume 1. PRENTICE HALL, third edition, 1995.

[12] N. Asokan, Phillipe A. Janson, Michael Steiner, and Michael Waidner. The state of the art in electronic payment systems. *Computer*, 30(9), September 1997.

[13] Bruce Schneier. *Applied Cryptography*. John Wiley & Sons, Inc., second edition, 1996.

[14] Carlos Molina-Jiménez and Lindsay Marshall. Anonymous and confidential communications from an IP addressless computer. In *Handheld and Ubiquitous Computing, Proceedings of the First International Symposium, HUC'99*, Karlsruhe, Germany, Sep 1999. Springer. Lecture Notes in Computer Science, 1707.

[15] ISO/IEC 8802-3. ANSI/IEEE Std 802.3. *International Standard. Information Technology–Local and metropolitan area networks. Part 3: Carrier sense with multiple access with collision detection CSMA/CD access method and physical layer specification*. IEEE, fourth edition, 1993.

[16] ISO/IEC 8802-4. ANSI/IEEE Std 802.4. *International Standard. Information processing Systems–Local and metropolitan area networks. Part 4: Token–passing bus access method and physical layer specification.* IEEE, first edition, 1990.

[17] ISO/IEC 8802-5. ANSI/IEEE Std 802.5. *International Standard. Information Technology–Local and metropolitan area networks. Part 5: Token ring access method and physical layer specification.* IEEE, fourth edition, 1992.

[18] Rich Seifert. *The Switch Book. The complete Guide to LAN Switching Technology.* Jonh Wiley & Sons, Inc., 2000.

[19] Gerard J. Holzmann. *Design and Validation of Computer Protocols.* Prentice Hall, 1991.