

FOREWORD

The strategic direction for development of the GTS, as endorsed by CBS, has since the early eighties, been based on the OSI standards, especially the ITU-T recommendation X.25. However, CBS now considers that the TCP/IP protocols as used on the Internet, should replace X.25 for supporting GTS operations in the future.

The change in strategic direction has evolved within the CBS in recent years. It has occurred for various reasons, including the expanding functional needs of the various WMO programs and the evolution of the Internet and its supporting technical standards, as a dominant force in the information technology industry, supplanting the OSI standards in many areas.

The transition to TCP/IP is considered appropriate because:

- Vendor support for X.25 technology is declining and becoming more expensive due to industry concentration on TCP/IP;
- TCP/IP supports numerous application utilities available off the shelf, which offer solutions to information communications needs of Members, such as file transfer, Web browsers, electronic mail and future applications such as multimedia communications;
- TCP/IP provides connectivity between Members in a more flexible and versatile manner than the X.25 based equivalent.

These benefits equate to direct savings in financial and human resource costs to Members by:

- reduced costs for communications equipment purchase and maintenance; and
- reduced software development work through use of industry standard software systems.

Considerable efforts have been applied in defining the framework for applying TCP/IP to the GTS and for the orderly transition from the OSI/X.25 based origin of the GTS. In particular, this Attachment on the Use of TCP/IP on the GTS has been produced.

Procedures are defined to ensure that the primary function of the GTS in carrying real time operational traffic with minimum delay is preserved. The issue of securing the GTS from interference via the Internet is also addressed in general terms. Reliance must however be placed on all Members with a TCP/IP based connection to the GTS, who are also connected to the Internet, to implement and maintain thorough security practices.

This Attachment was originally written as the culmination of work undertaken by CBS WG-TEL during 1997 and 1998. The TCP/IP procedures have since been implemented by many national Centres. The opportunity has been taken to capture the practical experiences gained in the use of TCP/IP and update material accordingly. In addition, a World Wide Web resource has been set-up which gives further details of the technical implementation of many of the concepts and procedures introduced within this Attachment. This is available on the ET-DCST information pages at <http://www.wmo.ch>.

Members are strongly advised to take account of the adoption of the TCP/IP based strategy for the future development of the GTS, in planning the future development of systems within their national Centres.

FOREWORD	1
1. INTRODUCTION.....	4
HISTORICAL PERSPECTIVE.....	4
PURPOSE OF THIS ATTACHMENT	4
RELATIONSHIP OF THE INTERNET AND GTS	4
EVOLUTION OF THE GTS.....	5
OTHER RELATED ISSUES.....	6
2. PRINCIPLES GOVERNING THE USE OF TCP/IP ON THE GTS.....	7
MANAGEMENT OF TRAFFIC ON GTS AND INTERNET	7
SECURITY ISSUES AND SEGREGATION OF INTERNET AND GTS TRAFFIC.....	8
ROUTING AND TRAFFIC MANAGEMENT.....	9
<i>Routing algorithms</i>	9
<i>Recommended routing method</i>	10
<i>Registered and private addresses</i>	10
<i>Implementation of GTS links via Internet</i>	12
<i>Summary of tasks to ensure proper use of IP on the GTS</i>	12
3. IMPLEMENTATION GUIDELINES.....	13
INTRODUCTION	13
ADDRESSING FOR X.25 PACKET SWITCHING BETWEEN CENTRES	13
ADDRESSING FOR IP OVER X.25	14
ADDRESSING FOR DIRECT IP.....	14
ADDRESSING FOR X.25 OVER IP	16
AUTONOMOUS SYSTEM NUMBERS.....	17
IMPLEMENTATION DETAILS	17
MANAGEMENT AND ALLOCATION OF ADDRESSES AND AS NUMBERS.....	27
<i>X.25 addresses</i>	27
<i>IP addresses</i>	27
<i>GTS nominated host/network addresses</i>	27
<i>AS numbers</i>	27
<i>Publication of addresses and AS numbers</i>	27
4 ADAPTING MESSAGE SWITCHING SYSTEMS TO TCP/IP.....	28
INTRODUCTION	28
TCP SOCKETS BASED MSS.....	28
FTP PROCEDURES	30
<i>Introduction</i>	30
<i>Accumulating messages into files</i>	30
<i>File naming conventions for existing message types (existing AHL)</i>	31
<i>File naming conventions for new message types (no existing AHL)</i>	31
<i>File renaming</i>	32
<i>Use of directories</i>	32
<i>Account names and passwords</i>	33
<i>FTP Sessions</i>	33
<i>Local FTP requirements</i>	33
<i>Use of file compression</i>	33
BACKUP WITH AN IP BASED GTS.....	33
5. TROUBLE SHOOTING AND PROBLEM RESOLUTION	34
IP LAYER TOOLS	34
<i>PING</i>	34
<i>TRACEROUTE</i>	35
<i>NETSTAT</i>	35
OTHER MONITORING TOOLS	36
<i>SNMP</i>	37

<i>MRTG</i>	38
<i>SYSLOG</i>	38
BANDWIDTH MANAGEMENT	40
APPENDICES	41
1. CISCO ROUTER CONFIGURATIONS.....	41
2. SAMPLE SOCKET SEND AND RECEIVE ROUTINES	46
3. SOME SECURITY ARRANGEMENTS FOR SMALL GTS CENTRES.....	55
<i>Security policy</i>	55
<i>Coexistence of Internet and dedicated GTS links</i>	55
<i>Protecting the GTS links from the Internet</i>	56
<i>GTS using the Internet</i>	57
4. REFERENCE MATERIAL.....	59
<i>General references on TCP/IP</i>	59
<i>References on Security</i>	59
5. SUGGESTED PASSWORD MANAGEMENT PRACTICES	60

1. Introduction

Historical perspective

The GTS at present is predominantly used to support the message switching application using message exchange in WMO format over a limited OSI transport service based on point to point X.25 supplemented by broadcasts. This limited implementation has been adequate for the legacy application of message switching but is not capable of meeting new requirements for support of various WMO programs, especially the World Weather Watch as developed within the CBS. These requirements include support for:

- Distributed Data Bases (DDB);
- Data exchange between non adjacent centres;
- Exchange of information that cannot readily be handled by message switching systems (MSSs).

The full list of requirements to be fulfilled by the Main Telecommunications Network (MTN) of the GTS were agreed upon by CBS-Ext. 1994. The use of TCP/IP services was endorsed by CBS-Ext. 1994 as a means of fulfilling these new requirements.

Purpose of this Attachment

This Attachment is intended to assist Centres to implement Transmission Control Protocol/Internet Protocol (TCP/IP) based services on the GTS. The aim of this Attachment is to describe those aspects of the application of TCP/IP that apply specifically to the GTS to meet new requirements and also the long established routine data exchange undertaken by Message Switching Systems (MSSs). The Attachment takes account of the technical evolution of the GTS from an X.25 based network, and maintains the philosophy that Centres continue to be autonomous as far as possible. It is recognised that the timing for implementation of new systems is determined by individual Members in the light of their available resources and relative priorities.

This Attachment does not cover fundamentals of TCP/IP but focuses on those aspects that are essential for successful application on the GTS. Such aspects include appropriate use of the GTS compared with the Internet, co-existence of the GTS and the Internet, IP and X.25 and Autonomous System addressing, router management, TCP/IP application services (such as FTP) and fault management. The Attachment gives an overview of recommended security practices with TCP/IP, but does not comprehensively address security issues and practices, this being a highly complex subject in itself. Some references on TCP/IP and on computer security are given in Appendix 3.

Relationship of the Internet and GTS

The recent and rapid emergence of the Internet poses issues to be decided as regards its role in relation to the GTS in meeting operational communications requirements of National Meteorological Services. The Internet has grown rapidly in capacity, penetration and diversity of applications. Its bandwidth greatly exceeds that of the GTS and it could potentially take over some functions of the GTS. The weakness of the Internet, as of 1999, is that its performance from day to day, even hour to hour is unpredictable due to its variable and rapidly growing traffic load. Furthermore its availability at various Centres differs in reliability and capacity. For some Centres it is quite possible that the absolute level of

Internet performance can be unacceptably low, while for others the Internet presents an adequate, cost-effective alternative to the traditional GTS point-to-point links. We must assume therefore that there will be a need for the Internet and the GTS to co-exist and plan accordingly.

The Attachment is based therefore on the assumption that the GTS with its limited but assured capacity will continue to be required for essential exchange between WMO Members. It should however where appropriate, adopt Internet technology and the Internet itself to improve versatility and maximise the scope for using standard software tools and services for the exchange of data and information. The limited capacity of the GTS creates a need for a practice of 'acceptable' use and for it to be engineered in such a way that it is protected from general Internet traffic and preserves security against inappropriate use and unauthorised access. In particular, the use of IP and dynamic routing protocols such as BGP4 (Border Gateway Protocol) on the GTS will have to be managed in such a way as to allow communication between non-adjacent Centres only with the knowledge and concurrence of all intermediate Centres. Otherwise there is a danger that large amounts of GTS capacity could be consumed by non-routine traffic, to the detriment of real time operational data exchange.

Evolution of the GTS

The use of the ISO/ITU standard X.25 was adopted by WMO in the early 1980's to facilitate the exchange of data and products encoded in WMO binary code forms (GRIB, BUFR etc) and to act as a base for higher level OSI applications. OSI was regarded at the time, as the strategic direction for the evolution of data communications. Since then X.25 at OSI layers 2 and 3 has been implemented on much of the GTS and virtually all of the MTN. The implementation has been predominantly one of permanent virtual circuits (PVCs) directly linking the MSSs of Members. There has been some movement towards switched virtual circuits (SVCs) as a result of the strategic deployment of packet switches by some centres as the first move towards making the GTS more of a true network and less a series of bilateral links. Such a strategy could be pursued but the emergence of the Internet and TCP/IP networking offers an alternative that appears much more attractive, particularly for non MSS requirements.

The evolution of the GTS to adopt TCP/IP is now appropriate because:

- it has become the dominant protocol suite in everyday use being now packaged with virtually all implementations of Unix and many PC operating systems such as Windows 95 and NT;
- it offers a wide range of standard applications (file transfer, electronic mail, remote logon, World Wide Web, etc.) that will greatly reduce the need for the WMO community to develop special procedures and protocols as it has had to do in the past.
- it provides useful features such as automatic alternate routing (in a meshed network) which could improve the reliability of the GTS.

This Attachment however takes account of the fact that centres have based plans and developed systems in line with the OSI standards, particularly X.25, as endorsed by WMO and specified in the Manual on the GTS. The adoption of TCP/IP based services must be implemented in an orderly transition from the X.25 based links in such a way that operation of the GTS is not disrupted or put at risk.

The Attachment provides for this by defining procedures for:

- an interim hybrid based on:
 - (i) carrying TCP/IP based services over an X.25 network service; or
 - (ii) carrying X.25 data over IP based network service via directly connected routers;
- subsequent transition to pure IP utilising directly connected routers, together with TCP/IP based application services, such as TCP sockets or File Transfer Protocol (FTP).

The transition to the second step (pure IP) is desirable because:

- Operating TCP/IP over X.25 may not provide expected throughput because of router processing overheads involved in packet encapsulation of IP frames within X.25 packets. This appears to become worse as line speeds increases. Limited tests which have been done between Centres in Region VI indicate efficiency less than 70% at 64Kbps.
- The management and maintenance activities required for the X.25 network and associated packet switches can be avoided.
- Carrying X.25 over IP requires use of proprietary features of specific router brands.

In order to move to pure IP, it is necessary to modify MSSs at each Centre to make use of TCP/IP services such as FTP and Sockets. This is covered in some detail in chapter 4.

Other related issues

Many Centres now have experience of TCP/IP on the GTS. Experience has shown that the main technical issues, which need to be addressed to establish widespread use of TCP/IP on the GTS, are:

- agreed methods for the message switching application to use TCP/IP either directly or via higher level applications e.g. FTP;
- an agreed file naming convention and standard for metadata associated with files;
- a community wide Naming and Addressing agreement.

It is the aim of this Attachment to make some progress with these issues, some of which lie in the domain of Data Management as much as Telecommunications. It must also be recognised that the existing GTS is not a true network, but a collection of discrete point-to-point links. Adoption of TCP/IP by some Centres has started to create a true network. Also managed networks using Frame Relay technology are now being introduced to the GTS. These developments introduce new issues regarding multi lateral co-operation in operating the GTS. While these issues are raised, they are beyond the scope of this Attachment.

2. Principles governing the use of TCP/IP on the GTS

Management of traffic on GTS and Internet

The TCP/IP protocol suite provides the potential to use the full range of TCP/IP applications on the GTS. Some applications such as file transfer, World Wide Web have potential to place heavy loads on the limited bandwidth circuits that comprise the GTS. Limits need to be applied to ensure that the GTS carries only important traffic such as the real time data and products currently exchanged on the GTS plus data to be carried to fulfil new requirements such as DDBs, and routinely exchanged large data files such as satellite imagery. Less important traffic such as ad hoc file exchange, e-mail, general World Wide Web and suchlike should be carried on the Internet. To protect the GTS, the full capabilities of TCP/IP connectivity and information exchange must be restricted. In practical terms, TCP/IP traffic carried on the GTS could be restricted on the basis of

- protocol type (e.g. FTP, HTTP, SMTP etc);
- originating and destination IP address;
- a combination of the above.

If the measures adopted are to be successful, it is necessary that they be:

- not confined to a single router brand since it cannot be assumed that all centres will have the same brand of router; and
- be reasonably straightforward to configure, so that there is minimum risk that configuration errors or omissions will endanger the GTS.

After considering these factors, the approach recommended is that only a small number of selected hosts in each Centre (e.g. the MSSs) be allowed to use the GTS, with no restriction placed on the protocol type. The concept is illustrated in figure 2.1. Host A_{NMC1} and A_{NMC2} , are “GTS designated” hosts. They are allowed to exchange traffic on the GTS using any TCP/IP protocol.

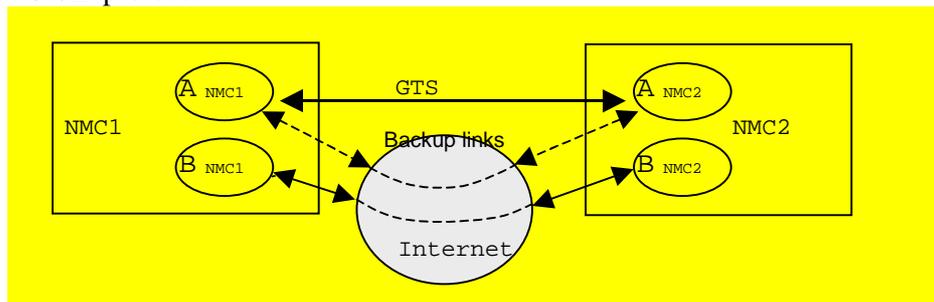


Figure 2.1 Connectivity between hosts in GTS Centres

Hosts B_{NMC1} and B_{NMC2} represents all other hosts in the two Centres which are not “GTS designated”. They must communicate via the Internet.

To achieve this functionality, the site manager at each centre must nominate which hosts are allowed to use the GTS. The GTS routers in each Centre must be configured so as to advertise only routes to “A” hosts, with packet filtering set to block packets from and to “B” hosts. A consequence of this is that A_{NMC2} will be unable to communicate with B_{NMC1} (and vice versa) even though both “A” and “B” hosts can access the Internet at large.

The actual configuration details to invoke the above functions in the Cisco family of routers are given in appendix 2.

In certain cases such as testing or backup to GTS dedicated links, it may be necessary for communication between "A" hosts to be via Internet. In this case, access lists in Internet access routers should permit only the specific "A" host in the neighbouring Centre to communicate via the Internet.

As a further precaution to protecting the GTS from traffic overload, the "A" hosts in a non adjacent Centre should only be permitted to communicate with the knowledge and concurrence of intermediate Centres.

Security issues and segregation of Internet and GTS traffic

Any Centre which has a TCP/IP based GTS connection and a connection to the Internet, is a potential weak point where the GTS could be exposed to deliberate or inadvertent interference through unwanted traffic or unauthorised connection to GTS hosts.

Centres are strongly encouraged to implement protective barriers such as firewall systems on the connection of their Centre with the Internet. It is important that every practical step is taken to prevent accidental or deliberate use of GTS links or unauthorised access to GTS Centres, by Internet users.

When setting up IP on the GTS, it is vital to ensure that the GTS does NOT become part of the Internet or an unintended transmission path for Internet traffic. Each Centre must consider the GTS and the Internet as two separate networks and ensure that inappropriate flow of traffic from one to the other cannot occur. This will ensure that the GTS is used only for transferring bona fide meteorological data between authorised hosts.

Some basic principles for implementing basic security measures for the GTS are shown in figure 2.2 below. It illustrates in a general way, how a Centre with TCP/IP connection to the GTS and an Internet connection might be set up. Functions to be implemented include:

- allowing only GTS designated hosts to communicate through the GTS router;
- blocking access to GTS designated hosts through the firewall and Internet router;
- firewall allows only approved hosts on the Internet to access B hosts and then, only for approved applications such as FTP;
- prevention of access to A hosts from Internet via B hosts.

The actual choice of routers and firewall and the setting up of these will require expertise in the design and configuration of networking and security systems. It is not intended here, to provide detailed coverage of security system implementation and management as it is a large and complex topic. It is simply emphasised that it is important that every Centre should implement the best practical security measures, appropriate to its system complexity and capabilities. Some additional material relevant to small Centres is given in Appendix 3

In addition to network security measures, it is vital that good security practices are followed in the management of all hosts in a Centre. Computer security is a complex subject in itself and Centres are encouraged to study this in depth and apply appropriate practices. Some references in computer security are given in Appendix 4. As a bare minimum, good password practices should be followed in the management of all host machines in a Centre. Some recommended practices are given in Appendix 5

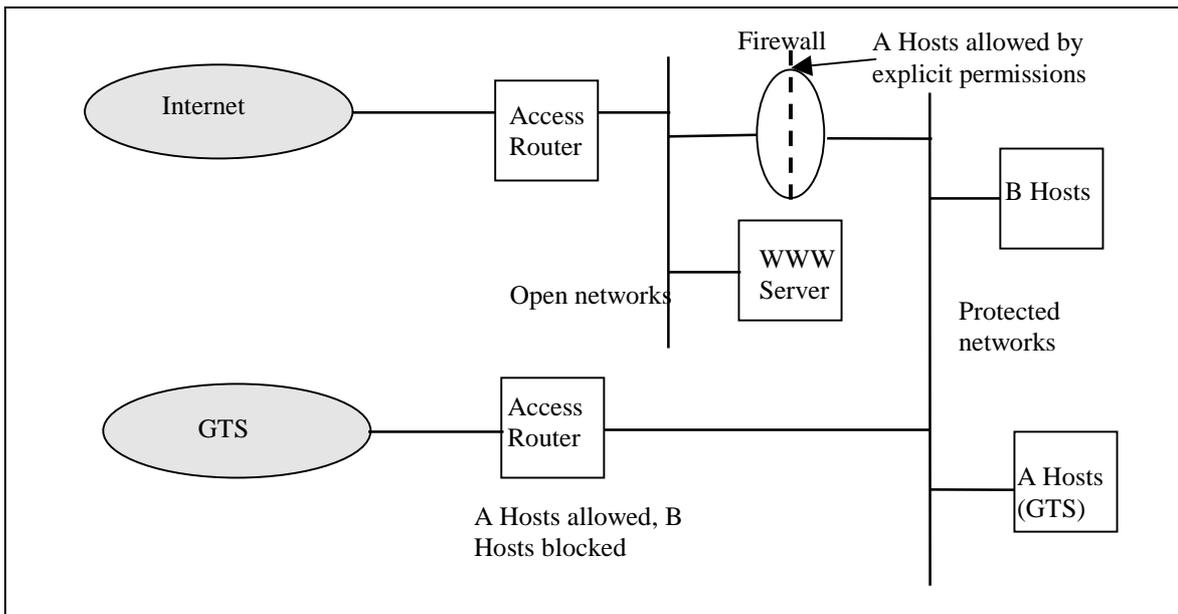


Figure 2.2 Generic arrangement for protecting GTS systems

Routing and traffic management

Routing algorithms

In order to be able to send a packet, every host, router or equipment connected on an IP network must have a routing table. The table tells the system where to send the packet. This may be achieved by:

- Static routing; or
- Dynamic routing.

Static routing

With static routing, every required destination and next hop must be entered in the routing tables by the system administrator. Alternatively, a default route can be declared, although this option is mainly applicable to sites with only one connection to the outside world. If a default route is set up, filters must be established to ensure that only authorised hosts can access the GTS.

Whenever a new Centre is connected to the GTS with IP protocol, the site managers of all other IP capable Centres must add the new address to their routing tables. This might become a major task as IP connectivity spreads over the GTS.

Dynamic routing

With dynamic routing, the routing information is automatically exchanged between routers. This enables the network to learn new addresses and to use alternative paths under fault conditions in a partially meshed network topology. The initial set-up of dynamic routing may be somewhat more complex, but the ongoing management task is greatly reduced.

Use of dynamic routing requires selection of an appropriate routing protocol to operate over the links of the GTS. The protocol must be an exterior gateway protocol (e.g. EGP, BGP) as opposed to an interior gateway protocol (such as IGRP, RIP, OSPF) because interior gateway protocols are intended for use within a single management domain. The GTS is an aggregation of many separate management domains. As such, it is necessary to select a gateway protocol that can be autonomously managed by each Centre to implement routing and hence traffic flow, consistent with its particular requirements.

Two exterior gateway protocols are defined by RFCs - EGP and BGP (now release 4 - RFC 1771). As the GTS is not a tree structure, setting up routing with EGP may be difficult. BGP 4 does not suffer topological constraints. It is more powerful, but a little more difficult to configure.

BGP can distribute subnetted routes. This feature might be very useful for the GTS. Instead of propagating host-based routes or full network routes, routing can be based on subnetted networks. Instead of declaring hosts eligible to use the GTS, a Centre could declare a full subnet of eligible hosts. In that case, the routing information consists of just an IP address and a subnet mask. For example, if a Centre has a class C addresses 193.168.1.0, by declaring that the subnet 193.168.1.16 with mask 255.255.255.248 is allowed to use the GTS, all hosts with IP address 193.168.1.17 to 193.168.1.22 will be routable on the GTS.

Recommended routing method

Based on consideration of the above factors the BGP4 routing protocol should be used between Centres on the GTS, unless an alternative is bilaterally agreed on individual links. Examples of BGP4 set-up for the Cisco router family are given in Appendix 2.

Registered and private addresses

It is recommended that Centres use officially registered IP addresses issued by their national Internet authority or by an Internet Service Provider (ISP). Nowadays, IP address space is administered by these organisations, rather than the global or regional authorities. These authorities are however a useful source of information on existing address allocations, through data base lookup services such as 'whois'. Major regional authorities are:

- ◆ Asia Pacific Network Information Centre (APNIC) <<http://www.apnic.net>>
- ◆ American Registry for Internet Numbers (ARIN) <<http://www.arin.net>>
- ◆ Reseau IP Europeens (RIPE NCC) <<http://www.ripe.net>>.

If Centres use private IP addresses on their internal networks, then Network Address Translation (NAT) must be adopted for any hosts requiring to communicate over the GTS or the Internet. A sufficient number of official addresses must be obtained to correspond to the number of hosts required to communicate externally, and the type of NAT supported by the Centre's access router. If static NAT is adopted, then a one to one correspondence of internal and official addresses is required. If dynamic NAT is used, then there can be more internal addresses than official addresses, with the router allocating the pool of official addresses dynamically as necessary. The documentation for the Centre's access router should be consulted to ascertain the NAT support provided.

Private addresses must not be visible on the GTS or Internet. Figure 2.3 shows simplified examples of allowable and non allowable arrangements.

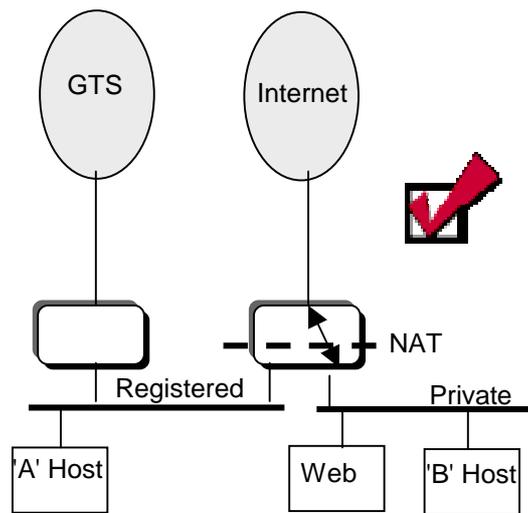


Figure 2.3 (a) (Allowed)

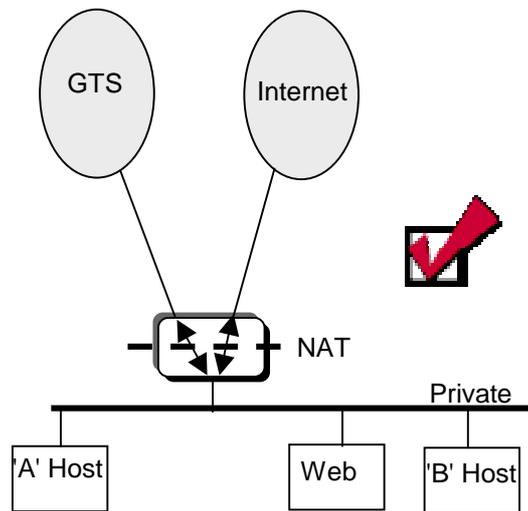


Figure 2.3 (b) (Allowed)

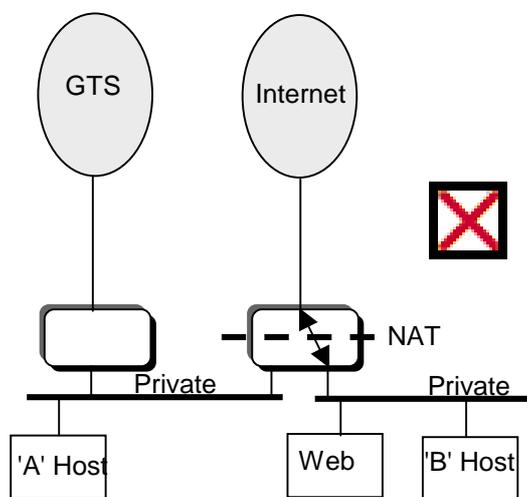


Figure 2.3 (c) (Not Allowed)

Implementation of GTS links via Internet

CBS has expressed the view¹ that the use of Internet for GTS links can be considered in circumstances where they are cost effective, offer an acceptable level of service and where adequate security measures are implemented. In general, the same principles for routing and security described above, apply where Internet links are used instead of dedicated links. Further details applying to the use of Internet based links, especially related to small GTS Centres, are given in Appendix 3.

Summary of tasks to ensure proper use of IP on the GTS

1. Use only official IP addresses for external communication on the GTS.
2. Declare which IP addresses in your Centre designates as eligible to use the GTS. (A list of allowed hosts and/or subnets is kept on the WMO FTP server.)
3. Obtain an autonomous-system number through the WMO Secretariat (which will maintain a list of AS numbers to be used on the GTS - refer Chapter 3) to be used for BGP configuration on the GTS.
4. Establish an IP connection with one or more Centres. This connection will be pure IP using PPP as a level 2 protocol on the link, (or a proprietary protocol such as Cisco HDLC by bilateral agreement) or IP over X.25 (RFC 1356). In this case use X.121 addresses as defined in Chapter 3.
5. Configure dynamic routing with BGP (unless you are a Centre with only one GTS connection and have agreed with your neighbouring Centre to use static routing)
6. Check the barrier between Internet and the GTS (prevent routing from the Internet to the GTS.)
7. Filter incoming and outgoing traffic in accordance with the requirements described above.

¹ CBS Ext Karlsruhe 1998, Final Report, paragraphs 4.4.35-45, and paragraph 4.4.40 in particular.

3. Implementation Guidelines

Introduction

The introduction of IP based services on the GTS will in many cases, be implemented initially by using a mixture of X.25 and IP, because of the technical evolution of the GTS as described in chapter 1. IP services may be carried over an X.25 network by encapsulating IP packets within X.25 packets. An appropriately configured router at each GTS Centre carries out this function. Alternatively, where routers of the same brand are used in adjacent Centres, X.25 data may be carried on an IP link using X.25 switching capability of the routers.

It is desirable ultimately for Centres to adopt through bilateral agreement, direct IP connections with TCP/IP application services (FTP, Sockets) superseding IP over X.25, or X.25 over IP as the case may be.

It is necessary to have an addressing framework for:

- X.25 packet switching between Centres;
- IP over X.25;
- direct IP (including X.25 over IP).

The use of BGP requires introduction of the concept of the Autonomous System (AS)². Each GTS Centre manages an AS number to enable the Centre to adopt BGP with neighbouring centres. In addition to addressing, this chapter shows allocation scheme of AS numbers.

Addressing for X.25 packet switching between Centres

Many centres have adopted X.25 for point to point connections between Message Switching Systems (MSS). A number of Centres have installed, or plan to install packet switches to provide capability for connections between non adjacent Centres. An addressing scheme has been developed for this purpose and has been adopted by WG-TEL 1994 (annex to para 5.4.1). It is a 14-digit scheme of the form:

0101xxxiiyyzz where:

0101	is a pseudo DNIC which does not correspond to any actual DNIC and therefore will ensure calls cannot be mistakenly switched to any network other than the GTS;
xxx	is the X.121 Country Code of the Centre;
ii	is a Protocol indicator, being 00 for MSS, 11 for TCP/IP, 22 for OSI CONS, 33 for OSI CLNS;
yyy	is the nationally assigned port number;
zz	is the nationally assigned sub address number;

This addressing scheme is to be used for setting up Virtual Calls (VCs) for MSS applications and for any other GTS applications including carriage of IP traffic over X.25.

² An Autonomous System is defined in RFC1630 as “a set of routers under a single technical administration, using an interior gateway protocol and common metrics to route packets within the AS, and using an exterior gateway protocol to route packets to other ASs.”

Addressing for IP over X.25

In order to carry IP traffic over X.25, two globally co-ordinated address schemes are necessary:

- an X.25 scheme as described above; and
- an IP address scheme to apply to the interface between the router and packet switch to enable the router to encapsulate the IP packets into X.25 packets.

The general arrangement is shown in figure 3.1.

For IP over X.25 to function correctly, it is necessary for the underlying X.25 network to be allocated a single IP network address and for each Centre to have an address within this network for the connection point between its router and its packet switch. **The Class C network address 193.105.177.0** has been allocated for this purpose, by agreement between Meteo France (the registered holder of this address) and WMO. Each IP node on the network will be assigned a sequential host address within this single Class C IP address as illustrated in Figure 3.1. The Class C address can provide for 254 Centres to be connected using a subnet mask of 255.255.255.0.

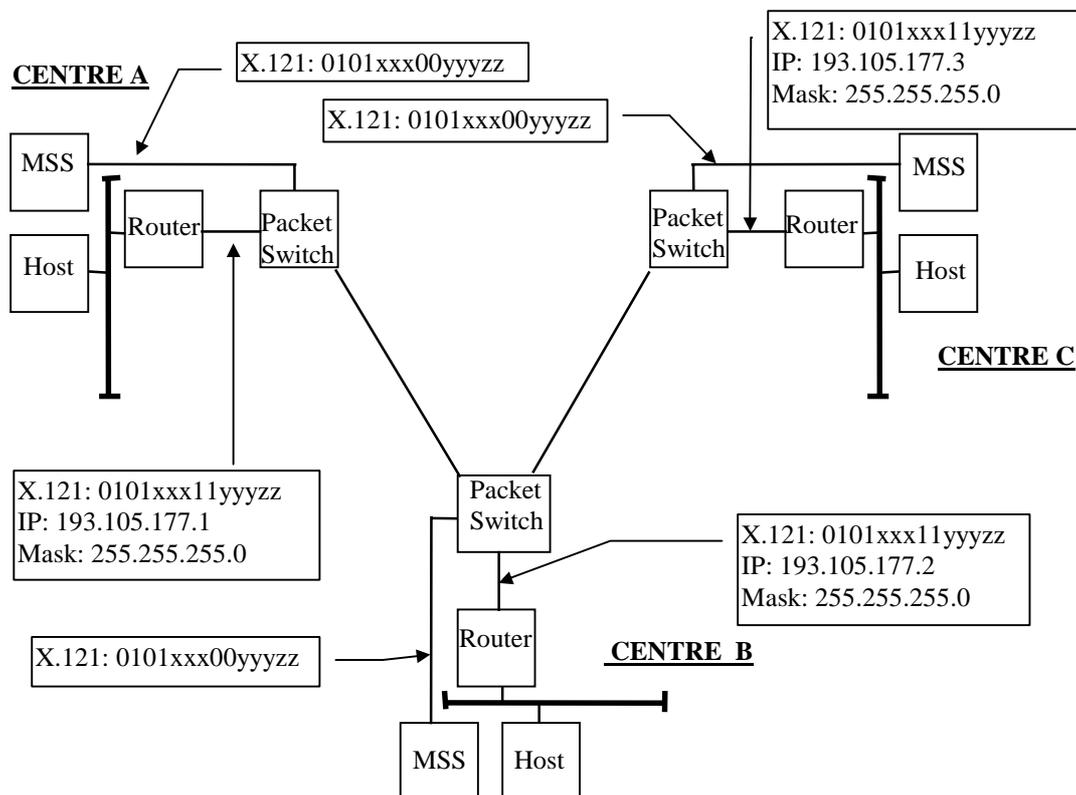


Figure 3.1 - IP implemented over X.25 Network

The Routers at each centre have to be set up so that they issue an X.25 call request to the X.25 port of the final destination Centre. This means that IP traffic passes through the packet switch only, and not the router of the intermediate Centre.

Addressing for Direct IP

At an appropriate future time, centres may wish to replace IP over X.25 with direct IP links with neighbouring centres under bilateral agreement. This transition would be appropriate

when the volume of IP traffic predominates and the MSSs are capable of communication using TCP/IP. A further seven Class C network addresses have been allocated for direct IP links between Centres, by agreement between Meteo France (the registered holder of these addresses) and WMO. Each Class C network address can provide 62 links (see box ‘Allocation of class C addresses for direct IP links’). The network addresses are:

MTN and Inter region links:	193.105.178.0
Links within RA I	193.105.179.0
Links within RA II	193.105.180.0
Links within RA III	193.105.181.0
Links within RA IV	193.105.182.0
Links within RA V	193.105.183.0
Links within RA VI	193.105.184.0

Further Class C addresses will be sought should the addresses available above be used up.

Figure 3.2 below illustrates how pair of Centres have agreed to implement a direct IP connection using the first available pair of ‘host’ numbers from the 193.105.178.0 network.

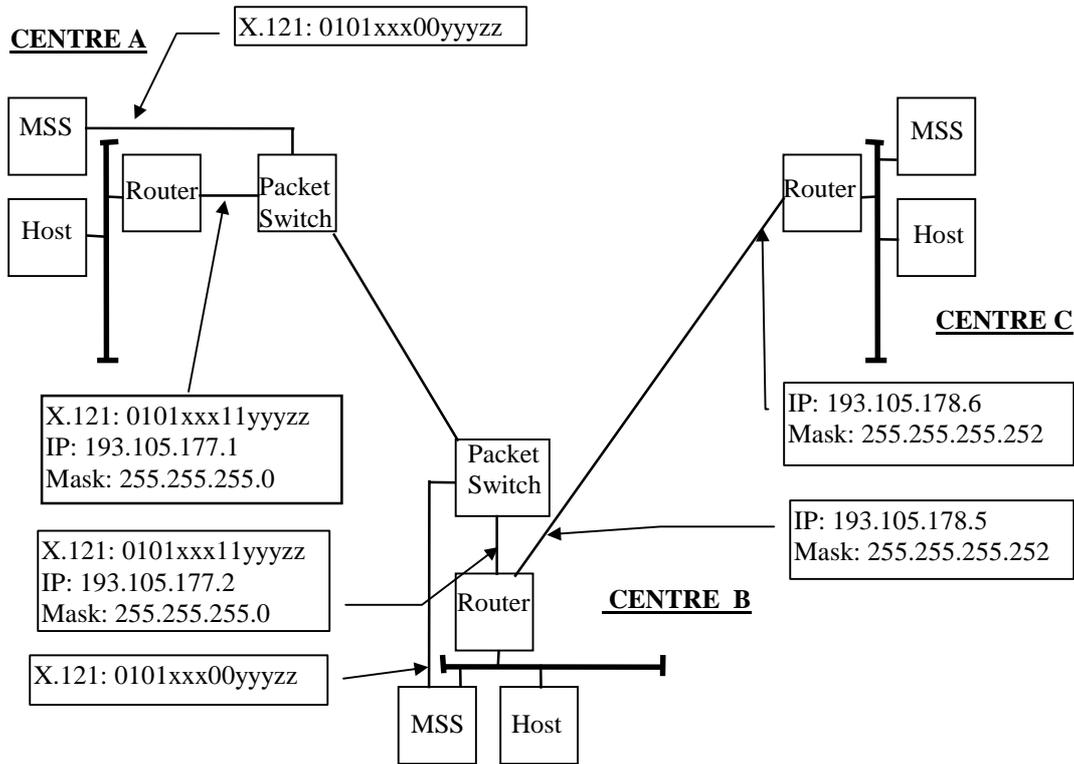


Figure 3.2. Direct IP link between centres B and C

Routers have to be connected by links having unique subnet numbers. To achieve this, a Class C address is used (for example 193.105.178.0) with a mask of 255.255.255.252. This provides 62 subnets each with 2 hosts. These two host numbers are allocated to the ends of the link connecting the routers between the two Centres. The lowest useable network number is 193.105.178.4, with host addresses of 193.105.178.5 and 6. The next network number is 193.105.178.8, with host addresses of 193.105.178.9 and 10, followed by 193.105.178.12, with host addresses of 193.105.178.13 and 14, followed by 193.105.178.16, with host addresses of 193.105.178.17 and 18, followed by 193.105.178.20, with host addresses of 193.105.178.21 and 22, and so on, up to 193.105.178.248, with host addresses of 193.105.178.249 and 250.

Allocation of class C addresses for direct IP links

Addressing for X.25 over IP

Where two Centres have a common brand of Router (e.g. Cisco), and the traffic is mostly IP with some X.25, it may be appropriate to carry the X.25 over the directly connected routers as shown for the link between Centre B and Centre C in figure 3.3. The X.25 packets are carried within IP packets over the serial link between the routers, which may be a proprietary HDLC protocol, or a standard protocol such as PPP. This functionality requires that routers in each Centre contain X.25 packet switching software and that the X.25 route details are included in the router configuration. Examples of typical configurations are given in appendix 2.

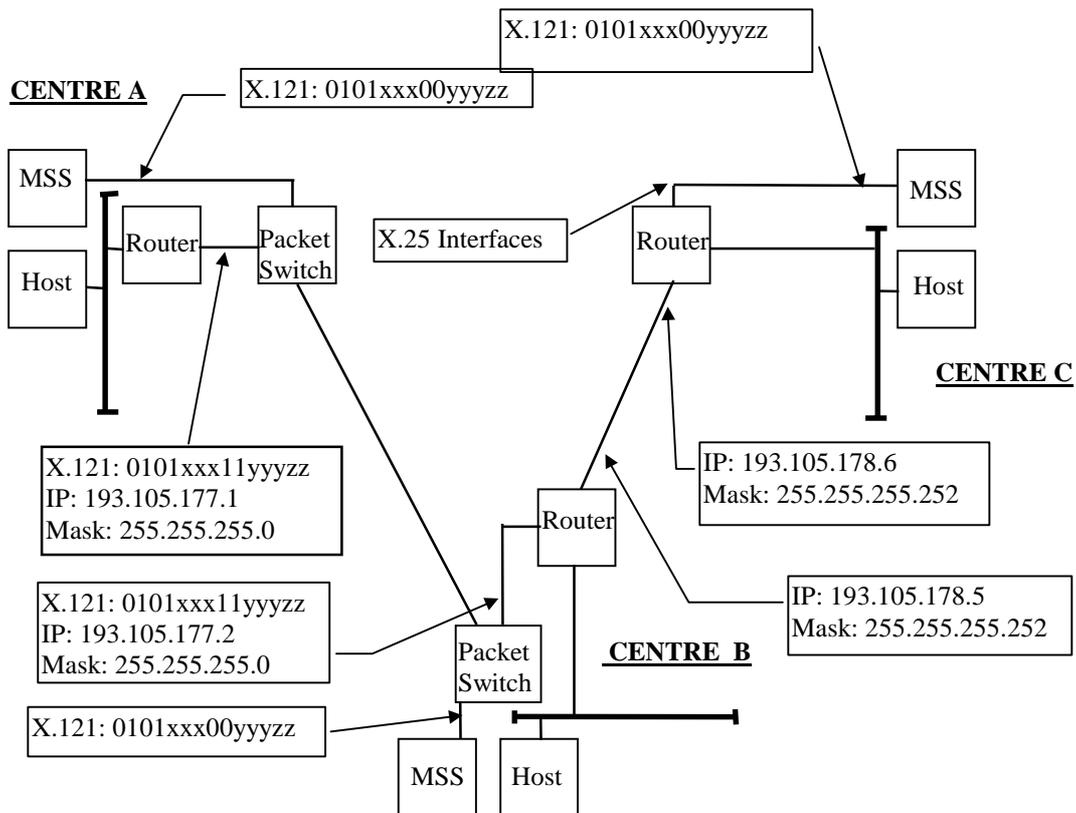


Figure 3.3, Combination of IP over X.25 and X.25 over IP

Autonomous System Numbers

The use of BGP4 as the recommended dynamic routing protocol for the GTS (Chapter 2) requires allocation of Autonomous System (AS) numbers to each GTS Centre.

The Internet Assigned Numbers Authority (IANA), through RFC1930, has reserved the block of AS numbers 64512 through 65535 for private use (not to be advertised on the global Internet). This provides 8 groups of 128 AS numbers to be assigned to GTS Centres, satisfying the current and foreseeable future needs of the GTS. The AS numbers will be assigned as follows:

MTN centres and reserve	64512 to 64639
Centres within RA I	64640 to 64767
Centres within RA II	64768 to 64895
Centres within RA III	64896 to 65023
Centres within RA IV	65024 to 65151
Centres within RA V	65152 to 65279
Centres within RA VI	65280 to 65407
Antarctic and reserve	65408 to 65471
*Private use by GTS Centres	65472 to 65535

* These AS numbers are for national use and are not to be advertised on the GTS.

Implementation details

In order to implement IP services Centres need to know certain details of IP and X.25 addressing at other Centres on the GTS. The following diagrams and associated tables explain in detail, the information required at various Centres:

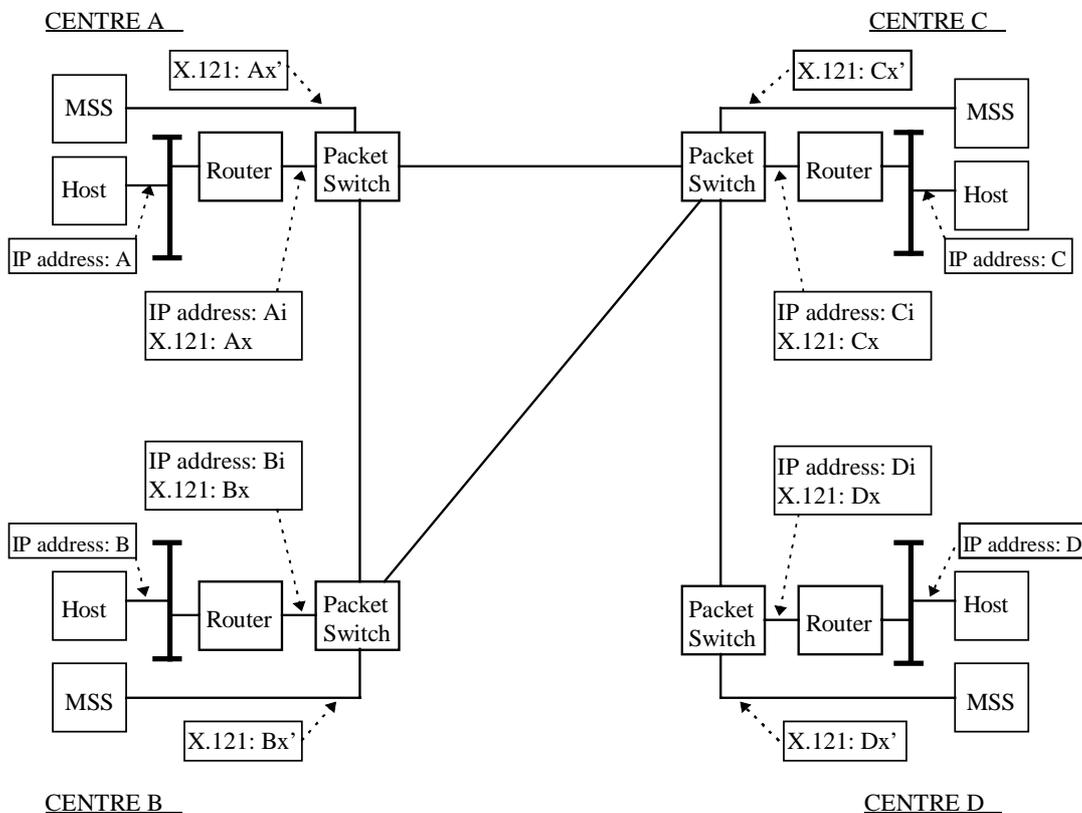


Figure 3.4 IP over X.25 Network

Table 3.4a. IP and X.121 addresses to be known at CENTRE A

Destination	Addresses to be known		Suitable route
	for communication between ends	for communication between Routers	
CENTRE B (Host to host)	IP address : B	IP address : Bi X.121 : Bx	CENTRE A – CENTRE B
CENTRE C (Host to host)	IP address : C	IP address : Ci X.121 : Cx	CENTRE A – CENTRE C
CENTRE D (Host to host)	IP address : D	IP address : Di X.121 : Dx	CENTRE A – CENTRE C – CENTRE D (Host [A] – Router [A] – Packet Switch [A] – Packet Switch [C] – Packet Switch [D] – Router [D] – Host [D]) [x] : CENTRE x
CENTRE B (MSS to MSS)	X.121 : Bx' (X.25 traffic)		CENTRE A – CENTRE B
CENTRE C (MSS to MSS)	X.121 : Cx' (X.25 traffic)		CENTRE A – CENTRE C
CENTRE D (MSS to MSS)	X.121 : Dx' (X.25 traffic)		CENTRE A – CENTRE C – CENTRE D (MSS [A] – Packet Switch [A] – Packet Switch [C] – Packet Switch [D] – MSS [D])

Table 3.4b. IP and X.121 addresses to be known at CENTRE B

Destination	Addresses to be known		Suitable route
	for communication between ends	for communication between Routers	
CENTRE A (Host to host)	IP address : A	IP address : Ai X.121 : Ax	CENTRE B – CENTRE A
CENTRE C (Host to host)	IP address : C	IP address : Ci X.121 : Cx	CENTRE B – CENTRE C
CENTRE D (Host to host)	IP address : D	IP address : Di X.121 : Dx	CENTRE B – CENTRE C – CENTRE D
CENTRE A (MSS to MSS)	X.121 : Ax' (X.25 traffic)		CENTRE B – CENTRE A
CENTRE C (MSS to MSS)	X.121 : Cx' (X.25 traffic)		CENTRE B – CENTRE C
CENTRE D (MSS to MSS)	X.121 : Dx' (X.25 traffic)		CENTRE B – CENTRE C – CENTRE D

Table 3.4c. IP and X.121 addresses to be known at CENTRE C

Destination	Addresses to be known		Suitable route
	for communication between ends	for communication between Routers	
CENTRE A (Host to host)	IP address : A	IP address : Ai X.121 : Ax	CENTRE C – CENTRE A
CENTRE B (Host to host)	IP address : B	IP address : Bi X.121 : Bx	CENTRE C – CENTRE B
CENTRE D (Host to host)	IP address : D	IP address : Di X.121 : Dx	CENTRE C – CENTRE D
CENTRE A (MSS to MSS)	X.121 : Ax' (X.25 traffic)		CENTRE C – CENTRE A
CENTRE B (MSS to MSS)	X.121 : Bx' (X.25 traffic)		CENTRE C – CENTRE B
CENTRE D (MSS to MSS)	X.121 : Dx' (X.25 traffic)		CENTRE C – CENTRE D

Table 3.4d. IP and X.121 addresses to be known at CENTRE D

Destination	Addresses to be known		Suitable route
	for communication between ends	for communication between Routers	
CENTRE A (Host to host)	IP address : A	IP address : Ai X.121 : Ax	CENTRE D – CENTRE C – CENTRE A
CENTRE B (Host to host)	IP address : B	IP address : Bi X.121 : Bx	CENTRE D – CENTRE C – CENTRE B
CENTRE C (Host to host)	IP address : C	IP address : Ci X.121 : Cx	CENTRE D – CENTRE C
CENTRE A (MSS to MSS)	X.121 : Ax' (X.25 traffic)		CENTRE D – CENTRE C – CENTRE A
CENTRE B (MSS to MSS)	X.121 : Bx' (X.25 traffic)		CENTRE D – CENTRE C – CENTRE B
CENTRE C (MSS to MSS)	X.121 : Cx' (X.25 traffic)		CENTRE D – CENTRE C

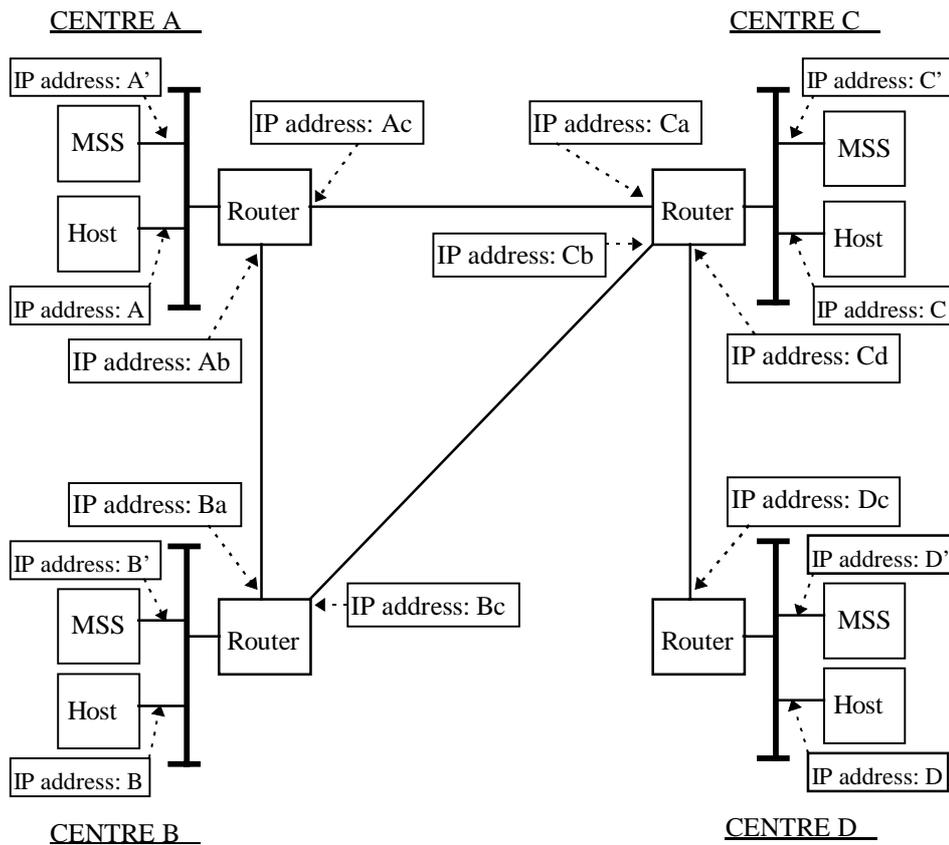


Figure 3.5. Direct IP Network

Table 3.5a. IP address to be known at CENTRE A

Destination	IP addresses to be known		Suitable route
	for communication between ends	for communication between Routers	
CENTRE B (Host to host)	IP address : B	IP address : Ba	CENTRE A – CENTRE B
CENTRE C (Host to host)	IP address : C	IP address : Ca	CENTRE A – CENTRE C
CENTRE D (Host to host)	IP address : D	IP address : Ca	CENTRE A – CENTRE C – CENTRE D (Host [A] – Router [A] – Router [C] – Router [D] – Host [D]) [x] : CENTRE x
CENTRE B (MSS to MSS)	IP address : B'	IP address : Ba	CENTRE A – CENTRE B
CENTRE C (MSS to MSS)	IP address : C'	IP address : Ca	CENTRE A – CENTRE C
CENTRE D (MSS to MSS)	IP address : D'	IP address : Ca	CENTRE A – CENTRE C – CENTRE D

Table 3.5b. IP address to be known at CENTRE B

Destination	IP addresses to be known		Suitable route
	for communication between ends	for communication between Routers	
CENTRE A (Host to host)	IP address : A	IP address : Ab	CENTRE B – CENTRE A
CENTRE C (Host to host)	IP address : C	IP address : Cb	CENTRE B – CENTRE C
CENTRE D (Host to host)	IP address : D	IP address : Cb	CENTRE B – CENTRE C – CENTRE D
CENTRE A (MSS to MSS)	IP address : A'	IP address : Ab	CENTRE B – CENTRE A
CENTRE C (MSS to MSS)	IP address : C'	IP address : Cb	CENTRE B – CENTRE C
CENTRE D (MSS to MSS)	IP address : D'	IP address : Cb	CENTRE B – CENTRE C – CENTRE D

Table 3.5c. IP address to be known at CENTRE C

Destination	IP addresses to be known		Suitable route
	for communication between ends	for communication between Routers	
CENTRE A (Host to host)	IP address : A	IP address : Ac	CENTRE C – CENTRE A
CENTRE B (Host to host)	IP address : B	IP address : Bc	CENTRE C – CENTRE B
CENTRE D (Host to host)	IP address : D	IP address : Dc	CENTRE C – CENTRE D
CENTRE A (MSS to MSS)	IP address : A'	IP address : Ac	CENTRE C – CENTRE A
CENTRE B (MSS to MSS)	IP address : B'	IP address : Bc	CENTRE C – CENTRE B
CENTRE D (MSS to MSS)	IP address : D'	IP address : Dc	CENTRE C – CENTRE D

Table 3.5d. IP address to be known at CENTRE D

Destination	IP addresses to be known		Suitable route
	for communication between ends	for communication between Routers	
CENTRE A (Host to host)	IP address : A	IP address : Cd	CENTRE D – CENTRE C – CENTRE A
CENTRE B (Host to host)	IP address : B	IP address : Cd	CENTRE D – CENTRE C – CENTRE B
CENTRE C (Host to host)	IP address : C	IP address : Cd	CENTRE D – CENTRE C
CENTRE A (MSS to MSS)	IP address : A'	IP address : Cd	CENTRE D – CENTRE C – CENTRE A
CENTRE B (MSS to MSS)	IP address : B'	IP address : Cd	CENTRE D – CENTRE C – CENTRE B
CENTRE C (MSS to MSS)	IP address : C'	IP address : Cd	CENTRE D – CENTRE C

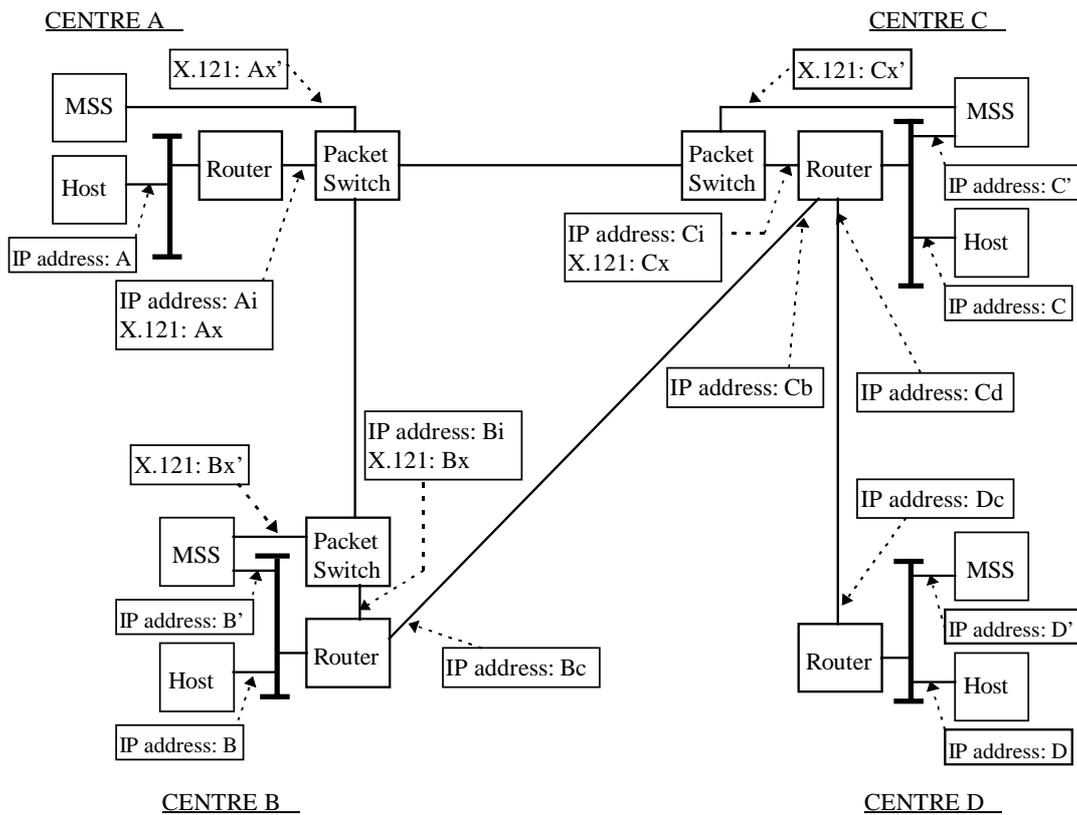


Figure 3.6. Coexistence of direct IP with IP over X.25

Table 3.6a. IP and X.121 addresses to be known at CENTRE A

Destination	Addresses to be known		Suitable route
	for communication between ends	for communication between Routers	
CENTRE B (Host to host)	IP address : B	IP address : Bi X.121 : Bx	CENTRE A – CENTRE B
CENTRE C (Host to host)	IP address : C	IP address : Ci X.121 : Cx	CENTRE A – CENTRE C
CENTRE D (Host to host)	IP address : D	IP address : Ci X.121 : Cx	CENTRE A – CENTRE C – CENTRE D
CENTRE B (MSS to MSS)	X.121 : Bx' (X.25 traffic)		CENTRE A – CENTRE B
CENTRE C (MSS to MSS)	X.121 : Cx' (X.25 traffic)		CENTRE A – CENTRE C
CENTRE D (MSS to MSS)	Possible only by store and forward via MSS at Centre C (X.25 traffic)		

Table 3.6b. IP and X.121 addresses to be known at CENTRE B

Destination	Addresses to be known		Suitable route
	for communication between end	for communication between Routers	
CENTRE A (Host to host)	IP address : A	IP address : Ai X.121 : Ax	CENTRE B – CENTRE A
CENTRE C (Host to host)	IP address : C	IP address : Cb	CENTRE B – CENTRE C
CENTRE D (Host to host)	IP address : D	IP address : Cb	CENTRE B – CENTRE C – CENTRE D
CENTRE A (MSS to MSS)	X.121 : Ax' (X.25 traffic)		CENTRE B – CENTRE A
CENTRE C (MSS to MSS)	IP address : C'	IP address : Cb	CENTRE B – CENTRE C
CENTRE D (MSS to MSS)	IP address : D'	IP address : Cb	CENTRE B – CENTRE C – CENTRE D

Table 3.6c. IP and X.121 addresses to be known at CENTRE C

Destination	Addresses to be known		Suitable route
	for communication between ends	for communication between Routers	
CENTRE A (Host to host)	IP address : A	IP address : Ai X.121 : Ax	CENTRE C – CENTRE A
CENTRE B (Host to host)	IP address : B	IP address : Bc	CENTRE C – CENTRE B
CENTRE D (Host to host)	IP address : D	IP address : Dc	CENTRE C – CENTRE D
CENTRE A (MSS to MSS)	X.121 : Ax' (X.25 traffic)		CENTRE C – CENTRE A
CENTRE B (MSS to MSS)	IP address : B'	IP address : Bc	CENTRE C – CENTRE B
CENTRE D (MSS to MSS)	IP address : D'	IP address : Dc	CENTRE C – CENTRE D

Table 3.6d. IP and X.121 addresses to be known at CENTRE D

Destination	Addresses to be known		Suitable route
	for communication between ends	for communication between Routers	
CENTRE A (Host to host)	IP address : A	IP address : Cd	CENTRE D – CENTRE C – CENTRE A
CENTRE B (Host to host)	IP address : B	IP address : Cd	CENTRE D – CENTRE C – CENTRE B
CENTRE C (Host to host)	IP address : C	IP address : Cd	CENTRE D – CENTRE C
CENTRE A (MSS to MSS)	Possible only by store and forward via MSS at Centre C (X.25 traffic)		
CENTRE B (MSS to MSS)	IP address : B'	IP address : Cd	CENTRE D – CENTRE C – CENTRE B
CENTRE C (MSS to MSS)	IP address : C'	IP address : Cd	CENTRE D – CENTRE C

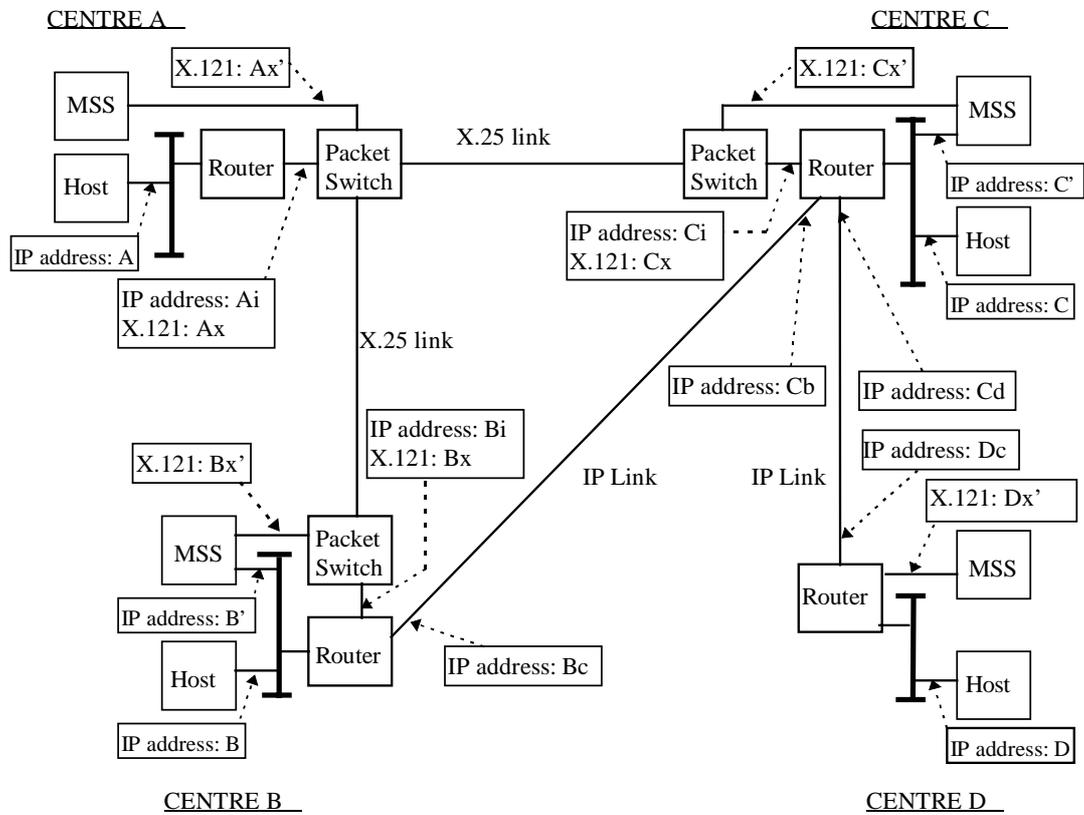


Figure 3.7. Coexistence of direct IP, IP over X.25 and X.25 over IP

Table 3.7a. IP and X.121 addresses to be known at CENTRE A

Destination	Addresses to be known		Suitable route
	for communication between ends	for communication between Routers	
CENTRE B (Host to host)	IP address : B	IP address : Bi X.121 : Bx	CENTRE A – CENTRE B (Host [A] - Router [A] - "IP over X.25" - Packet Switch [A] - "IP over X.25" - Packet Switch [B] - "IP over X.25" - Router [B] - Host [B])
CENTRE C (Host to host)	IP address : C	IP address : Ci X.121 : Cx	CENTRE A – CENTRE C (Host [A] - Router [A] - "IP over X.25" - Packet Switch [A] - "IP over X.25" - Packet Switch [C] - "IP over X.25" - Router [C] - Host [C])
CENTRE D (Host to host)	IP address : D	IP address : Ci X.121 : Cx	CENTRE A – CENTRE C – CENTRE D (Host [A] - Router [A] - "IP over X.25" - Packet Switch [A] - "IP over X.25" - Packet Switch [C] - "IP over X.25" - Router [C] - "Direct IP" - Router [D] - Host [D])
CENTRE B (MSS to MSS)	X.121 : Bx' (X.25 traffic)		CENTRE A – CENTRE B (MSS [A] - Packet Switch [A] - Packet Switch [B] - MSS [B])
CENTRE C (MSS to MSS)	X.121 : Cx' (X.25 traffic)		CENTRE A – CENTRE C (MSS [A] - Packet Switch [A] - Packet Switch [C] - MSS [C])
CENTRE D (MSS to MSS)	X.121 : Dx' (X.25 traffic)		CENTRE A – CENTRE C – CENTRE D (MSS [A] - Packet Switch [A] - Packet Switch [C] - Router [C] - "X.25 over IP" - Router [D] - MSS [B])

Table 3.7b. IP and X.121 addresses to be known at CENTRE B

Destination	Addresses to be known		Suitable route
	for communication between end	for communication between Routers	
CENTRE A (Host to host)	IP address : A	IP address : Ai X.121 : Ax	CENTRE B – CENTRE A
CENTRE C (Host to host)	IP address : C	IP address : Cb	CENTRE B – CENTRE C (Host [B] - Router [B] - "Direct IP" - Router [C] - Host [C])
CENTRE D (Host to host)	IP address : D	IP address : Cb	CENTRE B – CENTRE C – CENTRE D (Host [B] - Router [B] - "Direct IP" - Router [C] - "Direct IP" - Router [D] - Host [D])
CENTRE A (MSS to MSS)	X.121 : Ax' (X.25 traffic)		CENTRE B – CENTRE A
CENTRE C (MSS to MSS)	IP address : C'	IP address : Cb	CENTRE B – CENTRE C (MSS [B] - Router [B] - "Direct IP" - Router [C] - MSS [C])
CENTRE D (MSS to MSS)	X.121 : Dx' (X.25 traffic)		CENTRE B – CENTRE C – CENTRE D (MSS [B] - Packet Switch [B] - Router [B] - "X.25 over IP" - Router [C] - "X.25 over IP" - Router [D] - MSS [D])

Table 3.7c. IP and X.121 addresses to be known at CENTRE C

Destination	Addresses to be known		Suitable route
	for communication between ends	for communication between Routers	
CENTRE A (Host to host)	IP address : A	IP address : Ai X.121 : Ax	CENTRE C – CENTRE A
CENTRE B (Host to host)	IP address : B	IP address : Bc	CENTRE C – CENTRE B
CENTRE D (Host to host)	IP address : D	IP address : Dc	CENTRE C – CENTRE D (Host [C] - Router [C] - "Direct IP" - Router [D] - Host [D])
CENTRE A (MSS to MSS)	X.121 : Ax' (X.25 traffic)		CENTRE C – CENTRE A
CENTRE B (MSS to MSS)	IP address : B'	IP address : Bc	CENTRE C – CENTRE B
CENTRE D (MSS to MSS)	X.121: Dx' (X.25 traffic)		CENTRE C – CENTRE D (MSS [C] - Packet Switch [C] - Router [C] - "X.25 over IP" - Router [D] - MSS [D])

Table 3.7d. IP and X.121 addresses to be known at CENTRE D

Destination	Addresses to be known		Suitable route
	for communication between ends	for communication between Routers	
CENTRE A (Host to host)	IP address : A	IP address : Cd	CENTRE D – CENTRE C – CENTRE A
CENTRE B (Host to host)	IP address : B	IP address : Cd	CENTRE D – CENTRE C – CENTRE B
CENTRE C (Host to host)	IP address : C	IP address : Cd	CENTRE D – CENTRE C
CENTRE A (MSS to MSS)	X.121: Ax' (X.25 traffic)		CENTRE D – CENTRE C – CENTRE A
CENTRE B (MSS to MSS)	X.121: Bx' (X.25 traffic)		CENTRE D – CENTRE C – CENTRE B
CENTRE C (MSS to MSS)	X.121: Cx' (X.25 traffic)		CENTRE D – CENTRE C

Management and allocation of addresses and AS numbers

X.25 addresses

The framework described above allows Centres full autonomy in allocating X.25 numbers. The WMO Secretariat will maintain a current list of X.25 addresses which Centres have allocated for use on the GTS. Centres are requested to notify the Chief of Telecommunications and Monitoring Unit, WWW Department, WMO Secretariat by E-mail or fax of X.25 addresses allocated.

IP addresses

IP addresses for use with IP over X.25 or for pure IP links will be co-ordinated and issued by the WMO Secretariat as required. Centres should direct requests for IP numbers to WMO as described above.

GTS nominated host/network addresses

Host and subnet IP addresses for use with GTS nominated Centres should be notified to WMO as described above.

AS numbers

AS numbers for use on the GTS will be co-ordinated and issued by the WMO Secretariat as required. Centres should direct their requests for AS numbers to WMO as described above.

Publication of addresses and AS numbers

The WMO will publish updated lists of addresses and AS numbers in the monthly WWW Newsletter and will also make these lists available in ASCII text form for access by FTP on the WMO web server and in World Wide Web format at **<http://www.wmo.ch>** .

4 Adapting Message Switching Systems to TCP/IP

Introduction

Although there are new requirements emerging, for the time being GTS usage is dominated by the traditional Message Switching application, which has been developed to use X.25 packet switching. We now need to consider how best to migrate the message switching task to use TCP/IP to satisfy the new requirements by providing "Internet like" facilities on the GTS, and to stay aligned with IT industry trends. Additionally, migration of Message Switching Systems (MSS) to use TCP/IP means that X.25 infrastructure can be removed, greatly simplifying the technology of the GTS by moving to a pure IP network rather than a mixture of IP and X.25.

There are two possible technical approaches to this problem, one using TCP Sockets and the other FTP. In the long term the FTP approach is thought to be the most strategically attractive but may require more work to implement in operational Message Switching Systems. It may suit some Centres to adopt an approach based on TCP Sockets as the first step towards a TCP/IP based GTS.

The transition of the MSSs to TCP/IP does not imply any change in the basic store and forward architecture of the GTS. It is envisaged that the store and forward architecture, with automatic on forwarding based on routing tables, will remain. However, the adoption of FTP means there is an additional option for data exchange to be achieved through bilateral arrangements, by the use of FTP retrieve initiated by the receiving centre.

TCP Sockets based MSS

TCP Socket is an approach which is highly suitable for a programmatic implementation to provide regular exchange of messages. As such it should simply be regarded as an alternative protocol to X.25. A centre will be required to produce MSS application programs capable of transmitting and receiving via a TCP socket. Centres with current applications capable of driving an X.25 virtual circuit should be able to very quickly and simply produce a sockets version by changing a few system calls (see Appendix 3 for sample programs). The programming work involved is minimal and more importantly all other areas of the MSS such as queuing, routing, data management, operator interfaces etc. remain unchanged because the communication exchange is still based on the traditional message.

The protocol defined here, is based on the assumption that the physical circuit over which the data is to be transmitted has low error rate and is subject to interruptions rarely. On such circuits, the TCP protocol can be expected to deliver error free data. However, some GTS circuits may not be of sufficient quality for the standard TCP socket to function reliably. The development of special protocols for use on low quality circuits may be studied further.

Loss of data may occur if the TCP session is lost. This may be due to MSS hardware, application or communications failure. A special case of this is when a Centre with more than one MSS switches from the primary to the backup systems. Recommendations to avoid this problem are given below.

One useful feature of the X.25 based communication that is not available using TCP sockets, is the ability to detect start and end of message by reference to the M bit in the X.25 packet header. No such bit or any equivalent feature exists in TCP. Therefore, to enable receiving centres to detect end of message, each message is preceded with an 8 character string giving the message length, plus two characters indicating message type (binary, alphanumeric or fax). Thereafter the message is structured within an SOH/ETX envelope as for exchange via X.25. The complete structure is illustrated in figure 4.1. Note that the message length does

initialisation, and should not cause retransmission requests.

FTP Procedures

Introduction

FTP (File Transfer Protocol) is a convenient and reliable method for exchanging files, especially large files. The protocol is defined in RFC 959.

The main issues to be considered are:

1. Procedures for accumulating messages into files so as to minimise FTP overheads with short messages (applies only to existing message types);
2. file naming conventions for existing message types (existing AHL);
3. file naming conventions for new message types (no existing AHL);
4. file renaming;
5. use of directories;
6. account names and passwords;
7. FTP sessions;
8. Local FTP requirements;
9. File compression..

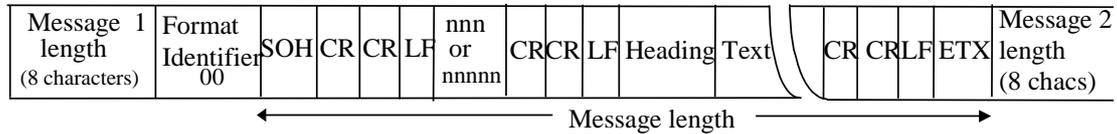
Accumulating messages into files

One of the problems with using FTP to send traditional GTS messages is the overhead if each message is sent in a separate file. To overcome this problem, multiple messages in the standard GTS message envelope should be placed in the same file according to the rules set out below. This method of accumulating multiple messages **applies only to messages for which AHLs have been assigned.**

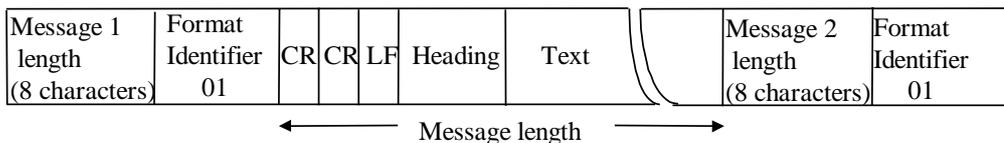
Centres have the option of including or deleting the Starting Line and End of Message strings and indicating which option they are using via the format identifier (refer points 2 and 4 below).

1. Each message should be preceded by an 8 octet message length field (8 ASCII characters). The length includes the Starting Line (if present), AHL, text and End of Message (if present).
2. Each message should start with either:
 - (i) the currently defined Starting Line and AHL as shown in figure 4.2, option 1; or
 - (ii) the AHL as shown in figure 4.2, option 2.
3. Messages should be accumulated in files thus:
 - (i) length indicator, message 1 (8 characters);
 - (ii) format identifier (2 characters);
 - (iii) message 1;
 - (iv) length indicator, message 2 (8 characters);
 - (v) format identifier (2 characters);
 - (vi) message 2;
 - (vii) and so on, until the last message;
 - (viii) If necessary, and subject to bilateral agreement, a 'dummy' message of zero length may be inserted after the last real message, to assist with end of file detection in certain MSS systems. This requirement does not exist in most cases and need only be implemented where necessary, and agreed between centres.
4. Format identifier (2 ASCII characters) has the following values:-
 - (i) 00 if Starting Line and End of Message strings present;
 - (ii) 01 if Starting Line and End of Message strings absent.

5. The sending centre should combine messages in the file for no more the 60 seconds to minimise transmission delays.
6. The sending centre should limit the number of messages in a file to a maximum of 100.
7. The format applies regardless of the number of messages, i.e. it applies even if there is only one message in the file.



Option 1. Starting Line and End of Message present
 Message length : Length from SOH to ETX (e.g. 00001826 = 1826bytes)



Option 2. Starting Line and End of Message absent
 Message length : Length from first CR to end of Text (e.g. 00001826 = 1826bytes)

Figure 4.2 Structure of a typical message in a file

File naming conventions for existing message types (existing AHL)

The file naming convention is:

CCCCNNNNNNNN.ext

where:

CCCC is the international four letter location identifier of the sending Centre, as defined in WMO publication No. 9, Volume C;

NNNNNNNN is a sequential number from 0 to 99999999 generated by the sending Centre;

ext is

‘ua’ for urgent alpha numeric information

‘ub’ for urgent binary information

‘a’ for normal alpha numeric information

‘b’ for normal binary information

‘f’ for facsimile information

Note: Where, through bilateral agreement, Centres allow alphanumeric and binary data in the one file, the b or ub extent shall be used.

File naming conventions for new message types (no existing AHL)

The procedure is based on transmission of file pairs, one file being the information file and the other being the associated metadata file. The concept of file pairs allows the communications function to be implemented independently data management requirements for structure of metadata, yet provides for the carriage of whatever metadata is required. It is not compulsory to always have a .met file, such as when the information file itself is self-specifying.

The name of the information file shall be:

CCCCNNNNNNNN.ext

where:

CCCC is the international four letter location identifier of the sending Centre, as defined in WMO publication No. 9, Volume C;

NNNNNNNN is a sequential number from 0 to 99999999 generated by the sending Centre;

ext indicates the file type, e.g.

.tif for TIFF file

.gif for GIF file

.ps for Postscript file

.mpg for MPEG file

.jpg for JPEG file

.txt for text file

.htm for HTML file

.bin for a file containing data encoded in a WMO binary code form such as GRIB or BUFR

.doc for a MS Word file

.wpd for a WordPerfect file

The name of the corresponding metadata file shall be:

CCCCNNNNNNNN.met

where **CCCCNNNNNNNN** is the same as the corresponding information file.

The structure of the '.met' file is not defined in this guide. It will be as specified by CBS WG-DM.

File renaming

The method used by receiving centres to detect the presence of a new file may depend on the type of machine used. However most centres will do this by scanning a directory for new files.

To avoid problems with the receiving centre processing a file before it has completely arrived, all sending centres must be able to remotely rename the files they send.

The file shall be sent with the extent '.tmp' and then renamed to use the appropriate extent defined above when the transfer is completed.

e.g.

(a) put xxxxx RJTD00220401.tmp (xxxxx = local file name)

rename RJTD00220401.tmp RJTD00220401.a

(b) put xxxxx AMMC09871234.tmp

rename AMMC 09871234.tmp AMMC09871234.gif

Use of directories

Some receiving centres may wish the files to be placed in specific sub-directories. This should be limited to require only that all files of the same type be delivered to the same directory. It is recommended that a separate directory be used for each host system which is initiating FTP sessions to avoid the possibility of filename duplication.

Account names and passwords

Using FTP the sender "logs in" to a remote machine using a specific account name and password. The receiving centre defines the account name and the password. There are potential security implications for centres so care needs to be taken.

The following general rules should however apply.

1. The receiving centre defines the user account and password for the sending centre.
2. Anonymous FTP may be used or a specific account may be created. (If anonymous FTP is used, each sending Centre must have its own sub directory on the FTP server).

FTP Sessions

To limit the load on both the sending and receiving systems, no more than one FTP session per file type should exist at the same time. If for example, Centre A wishes to send two files to Centre B of the same type (say .ua), the second file must not be sent until the first is finished. Centres should limit the number of concurrent sessions with a particular Centre to five maximum.

Local FTP requirements

All sending centres will need to allow for additional "static" FTP commands to be included in the FTP commands that they issue. For example some MVS centres may require the inclusion of "SITE" commands to define record and block lengths. Centres should support FTP commands as specified in RFC 959 unless some are excluded by bilateral agreement. There may also need to be bilaterally agreed procedures and commands.

It is the responsibility of receiving Centres to delete files after they have been processed.

Use of file compression.

If large files are to be sent then it is often desirable to compress them first.

Centres should only use compression by bilateral agreement.

If possible centres should support the following two compression methods. Other methods may be used by bilateral agreement.

1. UNIX "compress"
2. Gzip

This means that a received file could be of the form:

"xxxxxxx.ext.Z" or "xxxxxxx.ext.gz".

Backup with an IP based GTS

A final consideration is that of MSS backup. The new GTS will use IP addresses, where an individual address is usually associated with only one system. Should a system fail and an alternative be used there are implementation issues to be considered by transmitting centres. Ideally a transmitting centre should be unaffected by a receiving Centre's backup arrangements. This is a good principle, which all Centres should seek to adhere to. However it may not always be possible to achieve complete IP transparency. If this cannot be done sending Centres must be prepared to try an alternate IP address. Once using such an alternate address it must periodically try the primary address. It is suggested that such periodicity be established by bilateral agreement between centres because it will be heavily influenced by each centres backup strategy.

5. Trouble shooting and problem resolution

IP Layer Tools

In a large IP network, every router involved in the path between two hosts must know the next hop to be used to reach the destination address. As every router and/or link might be a point of failure, it is very important to determine rapidly where the problem is, and then how to solve it.

Suggested steps in resolving problems (not necessarily in the order given) are:

- check the remote centre (if the security policy of the remote centre allows it).
- check if the link to the “outside” network is reachable
- check the local network by trying to reach the next/default gateway
- check the local IP stack and configuration

Some basic tools that can be used such as Ping, Traceroute and Netstat are described below. PING and TRACEROUTE provide information on paths between hosts. They both use ICMP (traceroute also need UDP), but it should be noted that many sites block ICMP packets as part of their firewall security measures. To be able to locate problems in a network, it is necessary to have an exact documentation of the network.

PING

PING will check if the destination IP address can be reached. This tool is standard in almost every operating system with TCP/IP. On a Unix host the output looks like:

```
zinder# ping -s cadillac
PING cadillac : 56 data bytes
64 bytes from cadillac ( 193.168.1.17 ) : icmp_seq=0. time=3. ms
64 bytes from cadillac ( 193.168.1.17 ) : icmp_seq=1. time=2. ms
64 bytes from cadillac ( 193.168.1.17 ) : icmp_seq=2. time=3. ms
64 bytes from cadillac ( 193.168.1.17 ) : icmp_seq=3. time=3. ms
64 bytes from cadillac ( 193.168.1.17 ) : icmp_seq=4. time=5. ms
64 bytes from cadillac ( 193.168.1.17 ) : icmp_seq=5. time=3. ms
64 bytes from cadillac ( 193.168.1.17 ) : icmp_seq=6. time=3. ms
---cadillac PING statistics---
7 packets transmitted, 7 packets received, 0% packet loss
round-trip (ms) min/avg/max = 2/3/5
```

An useful test could be to ping the MSS of the neighbouring Centre. If this ping succeeds with an acceptable time delay, it would indicate that the network is operating correctly. If the ping fails, it could mean that the circuit is down or the ICMP ping packets are being blocked by the neighbouring Centre's router or firewall. In this event, it could be useful to ping the serial interface of the neighbouring Centre's router. If this succeeds, then the communications link to the neighbouring Centre is working. Any malfunction would then be within the neighbouring Centre.

Ping can be used to check whether the network performance is reasonable. The time is the delay between sending and receiving back the packet. It is not really possible to give an average value of the delay, but it is more important to notice any variation.

Finally, it might happen that packets are lost. In this case, there are missing numbers in the icmp_seq number. Either packet loss or variation in delays will badly degrade the performance.

TRACEROUTE

This tool is used to show which routers are transited on the network between A and B. As said above, traceroute need UDP and ICMP packets to work. Firewalls or packet filter on router may block such traffic as part of local security policy. It is not available on all systems, but is rather easy to compile. It is a free tool available on the Internet.

Traceroute output looks like:

```
cadillac 22: traceroute ftp.inria.fr
traceroute to ftp.inria.fr (192.93.2.54), 30 hops max, 40 byte packets
 1 antonio.meteo.fr (137.129.1.5) 3 ms 2 ms 2 ms
 2 clara.meteo.fr (137.129.14.249) 1 ms 2 ms 2 ms
 3 andrea.meteo.fr (193.105.190.253) 4 ms 3 ms 2 ms
 4 octares1.octares.ft.net (193.48.63.5) 30 ms 35 ms 10 ms
 5 192.70.80.97 (192.70.80.97) 9 ms 15 ms 27 ms
 6 stamand1.renater.ft.net (195.220.180.21) 40 ms 96 ms 29 ms
 7 stamand3.renater.ft.net (195.220.180.41) 56 ms 100 ms 108 ms
 8 stlambert.rerif.ft.net (195.220.180.10) 63 ms 56 ms 34 ms
 9 193.55.250.34 (193.55.250.34) 46 ms 28 ms 26 ms
10 rocq-gwr.inria.fr (192.93.122.2) 21 ms 147 ms 85 ms
11 ftp.inria.fr (192.93.2.54) 86 ms 58 ms 128 ms
```

When a router does not know where to send the packet, the result may be like the following:

```
cadillac 22: traceroute 193.105.178.5
traceroute to 193.105.178.5 (193.105.178.5), 30 hops max, 40 byte packets
 1 antonio.meteo.fr (137.129.1.5) 2 ms 1 ms 1 ms
 2 clara.meteo.fr (137.129.14.249) 1 ms 4 ms 1 ms
 3 andrea.meteo.fr (193.105.190.253) 4 ms 11 ms 4 ms
 4 octares1.octares.ft.net (193.48.63.5) 42 ms 39 ms 42 ms
 5 192.70.80.97 (192.70.80.97) 8 ms 7 ms 7 ms
 6 stamand1.renater.ft.net (195.220.180.5) 48 ms 86 ms 113 ms
 7 rbs1.renater.ft.net (195.220.180.50) 63 ms 107 ms 154 ms
 8 Paris-EBS2.Ebone.net (192.121.156.105) 146 ms 167 ms 140 ms
 9 stockholm-ebs-s5-2.ebone.net (192.121.154.21) 100 ms 80 ms 92 ms
10 Amsterdam-ebs.Ebone.NET (192.121.155.13) 249 ms 227 ms 205 ms
11 amsterdam1.NL.EU.net (193.0.15.131) 257 ms 249 ms 316 ms
12 * Amsterdam5.NL.EU.net (134.222.228.81) 300 ms 297 ms
13 Amsterdam6.NL.EU.net (134.222.186.6) 359 ms 218 ms 304 ms
14 Paris1.FR.EU.net (134.222.228.50) 308 ms 311 ms 388 ms
15 * Etoile0.FR.EU.net (134.222.30.2) 177 ms *
16 Etoile0.FR.EU.net (134.222.30.2) * * *
```

In the second case, cadillac would not be able to reach 193.105.178.5 because the router Etoile0.fr.eu.net failed to send the packet. With traceroute, it is not possible to know if it is a router failure or a link failure.

NETSTAT

This is a command available on most computing platforms. It gives information about the set up of the host's IP stack.

Netstat can be used to find out if the local IP address and subnet mask are configured correctly as well as if the routing information is still correct. There are many other options but it is not the intention of this guide to describe them all.

A sample output looks like:

```
$ netstat -rn
```

Routing tables

Internet:

Destination	Gateway	Netmask	Flags	Refs	Use	Interface
default	141.38.48.2		UG	12	4014211	ec0
127.0.0.1	127.0.0.1		UH	9	2321	lo0
141.38.48	141.38.48.12	0xfffff00	U	3	68981	ec0
141.38.48.12	127.0.0.1		UGH	10	253410	lo0
195.37.164.100	141.38.48.5		UGH	2	345	lo0
224	141.38.48.12	0xf0000000	U	1	19848	ec0

\$

The output shows that this particular host has the IP address 141.38.48.12 with a subnet mask of 24 bit (0xfffff00 or 255.255.255.0). It also shows that the host 195.37.164.100 can be reached via the gateway 141.38.48.5, and the flags indicate that the route is up (U), that it is a route to a gateway (G) and that it is a host route (H). The first line indicates that all other destinations are reachable via the hosts default gateway 141.38.48.2.

In the next output:

```
$ netstat -rn
```

Routing tables

Internet:

Destination	Gateway	Netmask	Flags	Refs	Use	Interface
default	141.38.48.2		UG	12	4014211	ec0
127.0.0.1	127.0.0.1		UH	9	2321	lo0
141.38.48	141.38.48.12	0xfffff00	U	3	68981	ec0
141.38.48.12	127.0.0.1		UGH	10	253410	lo0
195.37.164.100	141.38.48.2		UGHM	2	345	lo0
224	141.38.48.12	0xf0000000	U	1	19848	ec0

\$

The only difference to the first sample output is, that the host route to 195.37.164.100 is now flagged with a M, which means that this route was modified by an ICMP redirect message from the old gateway 141.38.48.5. This usually means that the router with the IP address 141.38.48.5 has lost its route to 195.37.164.100 and may indicate a problem with the link to the remote network.

Other monitoring tools

Verifying correct IP connectivity is a necessary first step. Other tools can be used to provide more information on what is happening. There are many options. It is possible to use protocol analysers and SNMP based software tools. For example, Sun Microsystems bundles with Solaris a tool called snoop who can replace in most cases a local area network analyser. Others tools such as TCPDUMP are available free on the Internet and can be installed on various systems. TCPDUMP is often bundled in various Linux distributions. These tools require a rather good knowledge of IP protocol. But, for example, TCPDUMP might be used to diagnose application level problems.

The following is a simple example on the host 'pontiac', of the capture of ICMP exchanges between zinder and cadillac.

```
pontiac# /usr/local/bin/tcpdump -i nf0 host cadillac and zinder and proto icmp
15:28:06.68 cadillac.meteo.fr > zinder.meteo.fr: icmp: echo request
15:28:06.68 zinder.meteo.fr > cadillac.meteo.fr: icmp: echo reply
15:28:19.45 cadillac.meteo.fr > zinder.meteo.fr: icmp: echo request
15:28:19.45 zinder.meteo.fr > cadillac.meteo.fr: icmp: echo reply
15:28:29.44 cadillac.meteo.fr > zinder.meteo.fr: icmp: echo request
15:28:29.45 zinder.meteo.fr > cadillac.meteo.fr: icmp: echo reply
```

SNMP

Simple Network Management protocol was developed in the late 80's in order to offer to network manager a standard tool for controlling networks. In most case SNMP could be used to replace more crude tools describe above. Unfortunately, good SNMP software is not cheap. SNMP is a client-server protocol. In order to be able to gather information with SNMP, the equipment connected on the network must have Management Information Base (MIB). These bases are catalogues of integer, counters, strings, etc.... The manager asks the agents to send it some values. These values might be for example, IP routing table. The example below is obtained by requesting with HP Open View (a commercial package) the routing table on the host monica.meteo.fr.

```
Title: : monica.meteo.fr
Name or IP Address: monica.meteo.fr
```

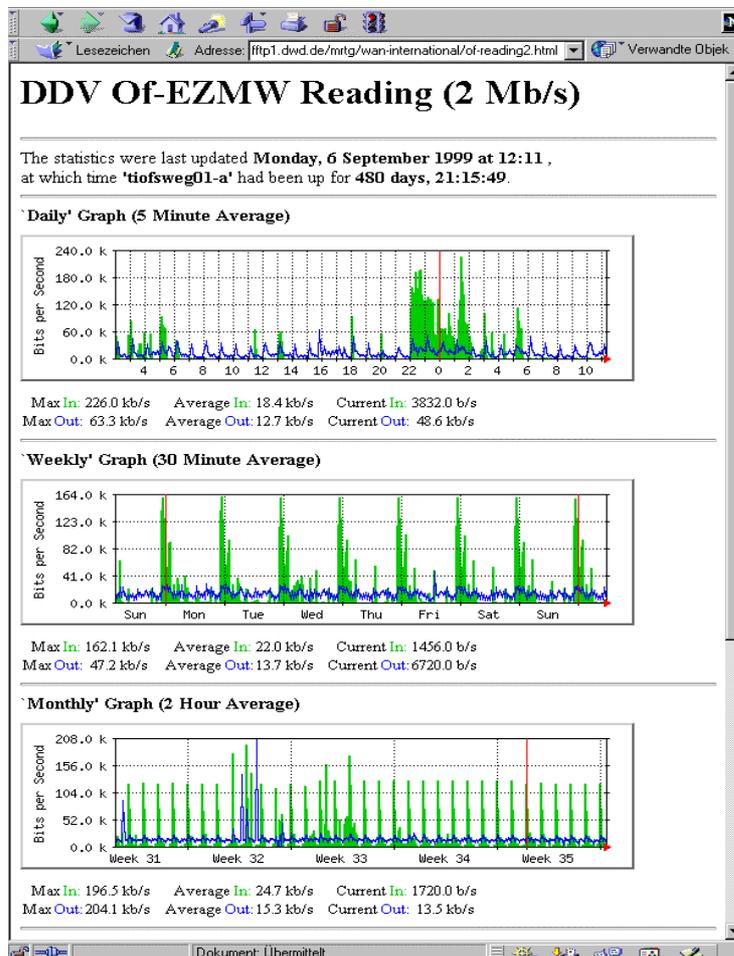
ipRouteDest	ipRouteMask	ipRouteNextHop	ipRouteProto	ipRouteMetric1
0.0.0.0	0.0.0.0	137.129.1.5	local	0
136.156.0.0	255.255.0.0	137.129.1.5	ciscoIgrp	8786
137.129.1.0	255.255.255.0	137.129.1.6	local	0
137.129.2.0	255.255.255.0	137.129.1.5	ciscoIgrp	1110
137.129.3.0	255.255.255.0	137.129.3.254	local	0
137.129.4.0	255.255.255.0	137.129.4.254	local	0
137.129.5.0	255.255.255.0	137.129.5.254	local	0
137.129.6.0	255.255.255.0	137.129.1.62	local	0
137.129.7.0	255.255.255.0	137.129.7.254	local	0
137.129.8.0	255.255.255.0	137.129.8.254	local	0
137.129.9.0	255.255.255.0	137.129.1.5	ciscoIgrp	1110

Information given above with TCPCDUMP might be obtained with SNMP but to do so, probes running the Remote Monitoring MIB must be connected on the network.

On a bilateral basis, it might be useful for Centres to allow SNMP access to their router from the other NMC. However, regular polling of other Centres' routers should be avoided to avoid overloading of circuits.

MRTG

Another public domain package, called MRTG, is a very helpful tool to gather information about the local network and about connected links. The Multi Router Traffic Grapher (MRTG) is a tool to monitor the traffic load on networks and links. It generates HTML pages containing images which provide a live visual representation of this traffic. It can also be implemented to indicate failures of network links. MRTG consists of a Perl script which uses SNMP to read the traffic counters of your router(s) and a fast C program which logs the traffic data and creates graphs representing the traffic on the monitored network connection(s). Below is a sample output. It shows traffic statistics for a dedicated link and gives information about the traffic pattern on the link. This is just one of many other graphs



one can create with MRTG. More information about MRTG can be found at <http://ee-staff.ethz.ch/~oetiker/webtools/mrtg>.

SYSLOG

Many of the possible problems can be located if one not only looks at the SYSLOG files on the hosts, but uses a SYSLOG server as well and lets the router(s) send their messages to it.

This file can then be checked regularly e.g. for messages that indicate high CPU load, processes that use up much memory or CPU cycles, lines going up and down, and messages about events regarding the used routing protocol.

There are 8 different levels of messages the router will log to the syslog server. They are:

Emergencies	0	System unusable
Alerts	1	Immediate action needed
Critical	2	Critical conditions
Errors	3	Error conditions
Warnings	4	Warning conditions
Notifications	5	Normal but significant condition
Informational	6	Informational messages only
Debugging	7	Debugging messages

The default logging facility on a cisco router is set to local7, this is important to know when configuring a host to be a syslog server and will be explained there.

The configuration commands on a Cisco router to activate logging are:

```
cisco-gts-1(config)#logging trap level-of-messages-to-log
cisco-gts-1(config)#logging 141.38.48.12
```

and can be checked with the command “show logging”:

```
cisco-gts-1#sho logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
Console logging: level debugging, 117892 messages logged
Monitor logging: level debugging, 8317 messages logged
Trap logging: level debugging, 117150 message lines logged
Logging to 141.38.48.12, 117150 message lines logged
Buffer logging: disabled
cisco-gts-1#
```

In this example, logging is set to the level debugging (“logging trap debugging”), and all messages from level 7 up to level 0 will be sent to the syslog server with the IP address 141.38.48.12.

To activate the SYSLOG server on for instance a SGI UNIX machine, the following entries should be there:

In the file /etc/services: syslog 514/udp

In the file /etc/syslog.conf: local7.debug /usr/people/cisco/logs/cisco.log

The local7.debug relates to the default facility of logging that is defined on a cisco router as mentioned (local7). The file above will be the file to which the syslog daemon writes all incoming syslog messages for local7.

The last action on the host is to have the syslog daemon reread it’s config file (kill -1 pid-of-syslogd).

Bandwidth Management

On an IP network, all packets will be routed over the links without any prioritisation mechanism. Therefore an FTP transfer can occupy all the bandwidth available starving all others applications. When traffic increases, it might therefore be needed to introduce some bandwidth management in the network configuration. Further information may be available on the online reference (<http://www.wmo.ch/>).

Appendices

1. Cisco Router Configurations

This appendix is not intended to be a complete description of all available commands in a Cisco, nor a full course on this equipment, but it is useful to describe more precisely the configuration tasks in order to comply with the policy outlined in chapter 2.

The configuration described below respects what is available in release 11.1 of Cisco IOS software. Some features are not available in previous releases, and some will be modified in the future.

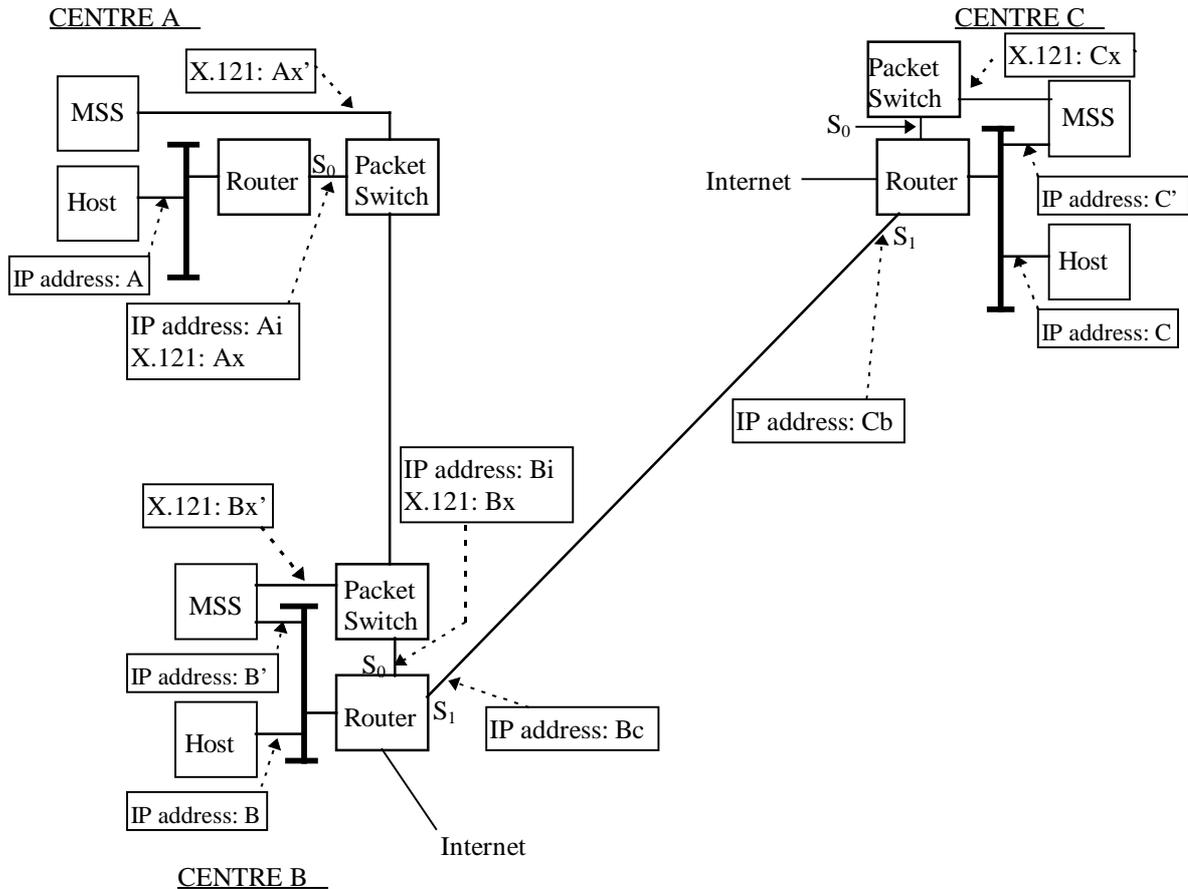
We are going to describe different steps:

1. Establishing IP connection
 - IP over PPP
 - IP over X25
 - X25 over IP (in fact it is X25 over TCP, the XOT protocol)
2. Routing configuration
 - leaf node with static routes (Centre A)
 - leaf node with dynamic routing (Centre C)
 - configuration in a non-leaf node (in our case two different GTS connections, Centre B)
3. Security configuration
 - filtering traffic based on declared IP addresses
 - controlling routing exchanges between GTS and the Internet

In our example A is connected to B with IP over X25 link, B is connected to C with IP over PPP. There is also the option for the MSSs at B and C to communicate using X.25 over TCP/IP. A is a leaf node, B and C are non-leaf nodes. B and C are also connected to the Internet. B and its Internet provider use static routes², C and its Internet provider use RIP³.

² B can't use EGP and BGP on the same router ; one router can't belong to more than one AS.

³ RIP is NOT a good choice for this type of configuration. But as RIP is the most basic protocol, it is used in this case too.



The following will be used along this appendix:

	X121 router address	IP router address	IP hosts address for GTS	Autonomous-System
Centre A	01016661166666	193.105.177.1	194.168.1.16/255.255.255.248	N.A.
Centre B	01017771177777	193.105.177.2	137.129.9.0/255.255.255.0	65001
Centre C	01018881188888	193.105.178.5	195.1.1.0/255.255.255.0	65200

Centres A and B use serial interface 0 to connect to the packet switches. Centres B and C use serial interfaces 1 for the PPP link.

Step 1: Establishing connections

Centre A:

```
interface serial 0
 encapsulation X25
 ! depending on local set-up (virtual channels,
 ! windows... ) extra configuration might be necessary
 x25 address 01016661166666
 ip address 193.105.177.1 255.255.255.0
 !
```

```
x25 map ip 193.105.177.2 01017771177777
```

Centre B:

```
interface serial 0
  encapsulation X25
  x25 address 01017771177777
  ! depending on local set-up (virtual channels,
  ! windows... ) extra configuration might be necessary
  ip address 193.105.177.2 255.255.255.0
  !
x25 map ip 193.105.177.1 01016661166666
  !
interface serial 1
  encapsulation PPP
  ip address 193.105.178.5 255.255.255.252
  !
  ! X25 over TCP commands
x25 routing
  x25 route 010188811* ip 193.105.178.6
  x25 route 010177711* interface serial 0
```

Centre C:

```
interface serial 0
  encapsulation X25
  x25 address 01018881188888
  ! depending on local set-up (virtual channels,
  ! windows... ) extra configuration might be necessary
interface serial 1
  encapsulation PPP
  ip address 193.105.178.6 255.255.255.252
  !
  ! X25 over TCP commands
x25 routing
  x25 route 010177711* ip 193.105.178.5
  x25 route 010188811* interface serial 0
```

After this first step, IP configuration between the routers is complete. The router in Centre A can then ping router in B. B can ping A and C, but A and C can't communicate because there is no routing set up.

MSS at B and C can communicate with IP (once end-to-end routing is established) or with X25 over TCP. Experience has shown that all the X25 parameters on router in centres B and C MUST be the same (packet size, window size) to avoid any strange behaviour.

Step 2: Routing

Centre A:

```
! Simply define a default route with a metric 10 (the
! price) via B
ip route 0.0.0.0 255.255.255.255 193.105.177.2 10
```

Centre B:

```
! First define static route with A
ip route 194.168.1.16 255.255.255.248 193.105.177.1 10
ip route 0.0.0.0 ip_provider_address 10
! BGP routing
router bgp 65001
  network 137.129.9.0 mask 255.255.255.0
  neighbour 193.105.178.6 remote-as 65200
  ! Route to A is static, force sending to C
```

```
redistribute static
```

Centre C:

```
! BGP routing
router bgp 65200
 network 195.1.1.0
 neighbour 193.105.178.5 remote-as 65001
! 196.1.1.0 is network address for non-GTS hosts in C
router rip
 version 2
 network 195.1.1.0
 no auto-summary
```

A defines a default route. So, when A wants to communicate with C, the router knows where to send packets. C is going to receive routing information from B, so A is also reachable from C. It is also important to note that if A tries to reach an Internet site, attempts will be made through B's Internet connection. It will fail because the Internet site A tries to reach can not return packets to A (only B's address is reachable on the internet via B's Internet connection). The link A to B link will thus carry some inappropriate data. Also note that we use RIP version 2 .

Step 3: Security

Centre A:

```
! Declare which hosts can use GTS
access-list 1 permit 194.168.1.16 0.0.0.7
! Declare which hosts can come from GTS
access-list 2 permit 195.1.1.0 0.0.0.255
access-list 2 permit 137.129.9.0 0.0.0.255
!
interface serial 0
 ip access-group 1 out
 ip access-group 2 in
```

Centre B:

```
! Declare which hosts can use GTS
access-list 1 permit 137.129.9.0 0.0.0.255
! Declare which hosts can come from GTS
access-list 2 permit 195.1.1.0 0.0.0.255
access-list 2 permit 194.168.1.16 0.0.0.7
! Only accept BGP updates from AS neighbour
ip as-path access-list 3 permit ^$
ip as-path access-list 3 permit ^65200
!
interface serial 0
 ip access-group 1 out
 ip access-group 2 in
!
interface serial 1
 ip access-group 1 out
 ip access-group 2 in
! Restrict BGP updates
router bgp 65001
 network 137.129.9.0 mask 255.255.255.0
 neighbour 193.105.178.6 remote-as 65200
 neighbour 193.105.178.6 filter-list 3 in
 neighbour 193.105.178.6 filter-list 3 out
 redistribute static
```

Centre C:

```
! Declare which hosts can use GTS
access-list 1 permit 195.1.1.0 0.0.0.255
! Declare which hosts can come from GTS
access-list 2 permit 137.129.9.0 0.0.0.255
access-list 2 permit 194.168.1.16 0.0.0.7
! Only accept BGP updates from AS neighbour
ip as-path access-list 3 permit ^$
ip as-path access-list 3 permit ^65001
!
interface serial 0
 ip access-group 1 out
 ip access-group 2 in
! Restrict BGP updates
router bgp 65200
 network 195.1.1.0 mask 255.255.255.0
 neighbour 193.105.178.5 remote-as 65001
 neighbour 193.105.178.5 filter-list 3 in
 neighbour 193.105.178.5 filter-list 3 out
```

In these configurations, there are two important features used:

- **BGP filtering**
The access-list 3 in both B and C checks the autonomous system number sent by its neighbour. By filtering in and out in the BGP process this guarantees that all known routes must be issued from one of these ASs.
- **IP filtering**
The access-list 1 list allows IP addresses issued from within each Centre. This list should be quite stable. The access-list 2 checks the incoming IP addresses. As new Centres are added to the IP network, the corresponding addresses must be added to these access-lists.

It must also be noted that despite Internet connections in B and C no extra attention is required to control routing exchange. A static default route is not sent even if « *redistribute static* » is enabled. RIP and BGP ignore routing information known via the other protocol.

2. Sample Socket Send and Receive Routines

```

/*****
* Sample TCP/IP Socket program that SENDS a single message
*****/

#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>
#include <signal.h>
#include <string.h>
#include <memory.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <netdb.h>

/* TCP/IP DESTINATION and SERVICE ARE DEFINED BY THE RECEIVING CENTRE */
#define DESTINATION "localhost"
#define SERVICE      39000
#define GTS_LENFIELD 8
#define MAX_MSGSIZE 15000 /* value of the send buffer size, recommended: 4096 */

static void GetDestinationInfo();
static void SetupSocket();
static void SendData();
static void MakeConnection();

static struct sockaddr_in dest;
static int      pr_sock;

/*****
*
*          MAINLINE
* 1. Ignore SIGPIPE signals. These are generated if a connection
*    is lost. By default they cause a program to terminate.
* 2. Get information about the destination (GetDestinationInfo):
*    - IP number (and name)
*    - Service/Port number
* 3. Create a TCP/IP Socket (SetupSocket)
* 4. Connect to the destination centre (MakeConnection)
* 5. Send the message (SendData)
* 6. Close the socket (shutdown + close)
*****/
main(int argc, char *argv[])
{

signal (SIGPIPE,SIG_IGN);

GetDestinationInfo();
SetupSocket();
MakeConnection();
SendData();
/* shutdown(pr_sock,1) */
close(pr_sock);

}

/*****
*
*          GET DESTINATION INFO

```

```

* Store the destination IP number and service number in a socket
* structure (dest).
* 1. Convert the destination name to an IP number (gethostbyname)
* 2. Store the IP number and service number in the "dest" structure.
*****/
static void GetDestinationInfo()
{
struct hostent *hp;

hp = gethostbyname (DESTINATION);
if ( hp == NULL ) {
    printf("host error\n");
    exit(1);
}

memset ((char *)&dest, 0, sizeof dest);
memcpy (&dest.sin_addr.s_addr, hp->h_addr, hp->h_length);
dest.sin_family = AF_INET;
dest.sin_port = SERVICE;
}

/*****
*                               SETUP SOCKET
* Setup a TCP/IP Socket
* 1. Create the socket
* 2. Set the socket KEEPALIVE option.
*   This enables the automatic periodic transmission of "check"
*   messages to be sent on the connection. If the destination
*   does not respond then it is considered broken and this process
*   is notified (by SIGPIPE or end-of-file)
*3. Set the socket REUSEADDR option. Enable quicker restarting of
*   terminated processes.
*4. Reduce the size of the Socket send buffer to reduce the amount of data lost
*   if the connection fails.
*****/
static void SetupSocket()
{
int    on = 1;
int    rc;
int    bufsize = MAX_MSGSIZE;

pr_sock = socket (AF_INET, SOCK_STREAM, 0);
if (pr_sock < 0) {
    printf("sock error\n");
    exit(1);
}

rc = setsockopt(pr_sock,SOL_SOCKET,SO_KEEPALIVE,(char *)&on,sizeof(on));
if (rc != 0) {
    printf("keepalive error\n");
}

rc = setsockopt(pr_sock,SOL_SOCKET,SO_REUSEADDR,(char *)&on,sizeof(on));
if (rc != 0) {
    printf("reuse error\n");
}

rc = setsockopt(pr_sock,SOL_SOCKET,SO_SNDBUF,(char *)&bufsize,sizeof(bufsize));
if (rc != 0) {

```

```

    printf("unable to set send buffer size\n");
    }
}

/*****
*
*           MAKE CONNECTION
* Attempt to make a TCP/IP Socket connection to the destination on
* the agreed service/port number.
*****/
static void MakeConnection()
{
    int    length;

    length = sizeof (dest);
    if ( connect (pr_sock,(struct sockaddr *)&dest,length) == -1 ) {
        printf("connection error\n");
        exit(1);
    }

    printf("connected\n");
}

/*****
*
*           SEND DATA
* Send a message on the socket (5 times actually).
*
* NOTE: A real program would check the return code from the write
* and if the write failed it would close the socket, raise an operator
* alarm, and then try to re-send from the start of the message
*****/
static void SendData()
{
    char    msg[MAX_MSGSIZE+1], buffer[MAX_MSGSIZE+GTS_LENFIELD+3];
    int     buflen, i, rc = 0;

    strcpy(msg,"\001\r\r\n001\r\r\nTTAA01 AMMC 000000\r\r\n");
    for (i=0;i<60;i++)
        strcat(msg,"THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG 0123456789\r\r\n");
    strcat(msg,"\r\r\n003");

    sprintf(buffer,"%0*dAN%s",GTS_LENFIELD,strlen(msg),msg);
    buflen = strlen(buffer);

    for (i=0; i<5; i++) {
        rc = write(pr_sock,buffer,buflen);
        printf("write. rc = %d\n",rc);
    }
}

```

```

/*****
* TEST TCP/IP SOCKET RECEIVING PROGRAM.
* Program is designed to give some ideas as to how to receive GTS
* style messages on a TCP/IP Socket connection.
*****/

#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>
#include <signal.h>
#include <string.h>
#include <memory.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <netdb.h>

#define SERVICE          39000
#define MAX_MSGSIZE     15000
#define MAX_BUFLEN      MAX_MSGSIZE + 100
#define SOH              '\001'
#define ETX              '\003'
#define GTS_LENFIELD    8
#define GTS_SOCKET_HEADER 10

static void SetupService();
static void RecvData();
static void AcceptConnection();
static int ExtractMsg(char *buffer, int *buflen);
static int CheckMsgBoundaries(char *, int);
static int FindMessage(char *, int, int *);
static void ShiftBuffer(char *, int *, int);

static struct sockaddr_in dest;

static int pr_sock, msgsock;
static char buffer[MAX_BUFLEN+1];
static int buflen = 0;

/*****
*
*                               MAIN
* Listen for incoming IP calls and read any incoming messages on
* the first call established.
*
* 1. Ignore SIGPIPE signals. These are generated if a connection
*    is lost. By default they cause a program to terminate.
* 2. Set-up a listening socket for incoming msgs (SetupService)
* 3. Accept the first call received (AcceptConnection)
* 4. Read any messages on this connection (RecvData)
* 5. Close the call and close the listening socket.
*****/
main(int argc, char *argv[])
{

signal(SIGPIPE,SIG_IGN);

SetupService();
AcceptConnection();
RecvData();

```

```

close(msgsock);
/* shutdown(pr_sock,1) */
close(pr_sock);
}

/*****
*
*          SETUP SERVICE
* Listen for calls on a given Service/Port.
* 1. Create a socket
* 2. Set the socket KEEPALIVE option.
*   This enables the automatic periodic transmission of "check"
*   messages to be sent on the connection. If the destination
*   does not respond then it is considered broken and this process
*   is notified (by SIGPIPE or end-of-file)
* 3. Set the socket REUSEADDR option. Enable quicker restarting of
*   terminated processes.
* 4. Bind the socket to the required Service/Port
* 5. Start listening for calls.
*****/
static void SetupService()
{
int     on = 1;
int     rc;
/* adjust the TCP receive buffer size
int     bufsize = MAX_MSGSIZE; */

memset ((char *)&dest, 0, sizeof dest);
dest.sin_addr.s_addr = INADDR_ANY;
dest.sin_family = AF_INET;
dest.sin_port = SERVICE;

pr_sock = socket (AF_INET, SOCK_STREAM, 0);
if (pr_sock < 0) {
    printf("sock error\n");
    exit(1);
}

rc = setsockopt(pr_sock,SOL_SOCKET,SO_KEEPALIVE,(char *)&on,sizeof(on));
if (rc != 0) {
    printf("keepalive error\n");
    exit(1);
}

rc = setsockopt(pr_sock,SOL_SOCKET,SO_REUSEADDR,(char *)&on,sizeof(on));
if (rc != 0) {
    printf("reuse error\n");
    exit(1);
}

/* adjust the TCP receive buffer size
rc = setsockopt(pr_sock,SOL_SOCKET,SO_RCVBUF,(char *)&bufsize,sizeof(bufsize));
if (rc != 0) {
    printf("unable to set send receive size\n");
}
*/

rc = bind(pr_sock,(struct sockaddr *)&dest,sizeof dest);
if ( rc < 0) {
    printf("bind error\n");
    exit(1);
}

```

```

rc = listen(pr_sock,1);
if ( rc < 0 ) {
    printf("listen error\n");
    exit(1);
}

printf("listening\n");
}

/*****
*
*          ACCEPT CONNECTION
* Wait for an incoming call (accept).
* Return the socket of the call established.
*****/
static void AcceptConnection()
{
    int    addrlen;

    printf("waiting connection\n");

    addrlen = sizeof(sockaddr_in);
    msgsock = accept (pr_sock,&dest,&addrlen);
    if ( msgsock < 0 ) {
        printf("accept error\n");
        exit(1);
    }
    printf("connected\n");
}

/*****
*
*          RECV DATA
* Read data from the message/call socket.
* Extract GTS messages from this data.
* Keep reading until the sender drops the call or there is an error.
*****/
static void RecvData()
{
    int    numr = 1;
    int    rc = 0;

    while (numr > 0 && rc >= 0) {
        numr = read(msgsock,buffer+buflen, MAX_BUFLEN-buflen);
        if (numr > 0) {
            buflen += numr;
            buffer[buflen] = '\0';
            printf("buffer = %s\n",buffer);
            rc = ExtractMsg(buffer,&buflen);
        }
    }
}

/*****
*
*          EXTRACT MSG
* DESCRIPTION
* This function accepts a buffer of data on input, along with the
* amount of data in the buffer, and extracts GTS messages from this
* buffer.
*****/

```

```

*
* Messages that are in the buffer are identified as follows...
*
* - The first 8 bytes of the message buffer HAVE to be a message
*   length in character format.
*   If the length exceeds the GTS defined maximum message size, or
*   does not consist of numeric characters, then an error is returned
*   (lost synchronisation).
*
* - Immediately following the message length is a 2 character
*   Message Type: "AN" = Alphanumeric, "BI" = binary, "FX" = Fax
*
* - The GTS message begins with a SOH character, and is terminated
*   with a ETX character, if this does not occur, then an error is
*   returned (lost synchronization).
*
* - If a GTS message is identified, then it is extracted and the
*   message is shifted out of the buffer.
*
* - As there may be more than 1 message in the buffer, this function
*   will loop (extracting messages) until either an
*   error or incomplete message is detected.
*
* RETURNS   = 0 - Not a complete message in the buffer.
*           < 0 - Fatal error in the format of the buffer.
*           > 0 - Success, the message(s) have been extracted
*****/
static int ExtractMsg(char *buffer, int *buflen)
{
int    rc, msglen;
char   msg[MAX_MSGSIZE+1];

/* FIND THE FIRST MESSAGE IN THE BUFFER */
rc = FindMessage (buffer, *buflen, &msglen);

/* WHILE A VALID MESSAGE LENGTH IS FOUND IN THE MESSAGE BUFFER... */
while ( rc > 0 ) {

/* ENSURE THAT THE FIRST CHARACTER AFTER THE MESSAGE LENGTH IS
   A 'SOH' CHARACTER, AND THE LAST CHARACTER AS INDICATED BY
   THE MESSAGE LENGTH IS AN 'ETX' CHARACTER. */
if ( (rc = CheckMsgBoundaries (buffer, msglen)) < 0 )
    continue;

/* PRINT THE EXTRACTED MESSAGE */
memcpy(msg,buffer+GTS_SOCKET_HEADER,msglen);
msg[msglen] = '\0';
printf("GTS MSG = \n%s\n",msg);

/* SHIFT THE JUST INJECTED MESSAGE OUT OF THE MESSAGE BUFFER,
   AND LOOP BACK TO LOOK FOR A NEW MESSAGE. */

ShiftBuffer (buffer, buflen, msglen);

/* FIND THE FIRST MESSAGE IN THE SHIFTED BUFFER */
rc = FindMessage (buffer, *buflen, &msglen);

}

```

```

return (rc);
}

/*****
*
*           FIND MESSAGE
* Check that the complete message is at the start of the buffer.
* 1. Check the first 8 characters which are the message length
* 2. Check the next 2 characters - Message Type
* 3. Check that the complete message, as defined by the "message length"
*   field, is in the buffer.
* Return codes:
*   0 = message incomplete
*   1 = message complete
*  -1 = error
*****/
static int FindMessage (char *buffer, int buflen, int *mlen)
{
    char charlen[GTS_LENFIELD+1];
    int intlen;

    *mlen = 0;

    /* IF THE LENGTH OF THE PASSED MESSAGE BUFFER IS NOT GREATER THAN
       10 CHARACTERS THEN RETURN 'INCOMPLETE'. */
    if ( buflen < GTS_SOCKET_HEADER ) {
        return (0);
    }

    /* CHECK THAT THE MESSAGE TYPE IS VALID */
    if (strncmp(buffer+GTS_LENFIELD,"AN",2) && strncmp(buffer+GTS_LENFIELD,"BI",2) &&
        strncmp(buffer+GTS_LENFIELD,"FX",2)) {
        printf("ERROR: Message Type field invalid");
        return (-1);
    }

    /* EXTRACT THE MESSAGE LENGTH */
    strncpy (charlen, buffer, GTS_LENFIELD);
    charlen[GTS_LENFIELD] = '\0';

    /* CHECK THAT THE MESSAGE LENGTH CHARACTER STRING COMPRISES
       ENTIRELY OF DIGITS. RETURN AN ERROR IF THIS IS NOT THE CASE. */
    if ( strspn (charlen, "0123456789") != strlen (charlen) ) {
        printf("ERROR: length not numeric");
        return (-1);
    }

    /* CONVERT THE MESSAGE LENGTH CHARACTER STRING TO AN INTEGER. */
    intlen = atoi (charlen);

    /* CHECK THAT THE LENGTH EXTRACTED FROM THE BUFFER IS NOT GREATER
       THAN THE GTS DEFINED MAXIMUM MESSAGE SIZE - RETURN AN ERROR IF
       THIS IS THE CASE. */
    if ( intlen > MAX_MSGSIZE ) {
        printf("ERROR: message overlength");
        return (-1);
    }

    /* CHECK IF THE ENTIRE MESSAGE HAS BEEN RECEIVED. RETURN IF NOT */

```

```

if ( buflen < intlen + GTS_SOCKET_HEADER ) {
    return (0);
}

*mlen = intlen;
return (1);
}

/*****
*
*          CHECK MSG BOUNDARIES
* Confirm the first character after the Socket Header is
* a SOH, and the last character in the message (given by the message
* length) is an ETX.
*****/
static int CheckMsgBoundaries (char *buffer, int msglen)
{

    /* CHECK THAT THE FIRST CHARACTER (AFTER THE MESSAGE LENGTH
    FIELD) IS AN SOH CHARACTER - RETURN AN ERROR IF IT ISN'T. */
    if ( buffer[GTS_SOCKET_HEADER] != SOH ) {
        printf("ERROR: SOH not found\n");
        return (-1);
    }

    /* CHECK THAT THE LAST CHARACTER (ACCORDING TO THE MESSAGE LENGTH
    FIELD) IS AN ETX CHARACTER - RETURN AN ERROR IF IT ISN'T. */
    if ( buffer[msglen+GTS_SOCKET_HEADER-1] != ETX ) {
        printf("ERROR: ETX not found\n");
        return (-1);
    }

    return (1);
}

/*****
*
*          SHIFT BUFFER
* Shift the leading message in the buffer out of the buffer. This may
* either empty the buffer, or move all or part of a new message to the
* start of the buffer.
*****/
static void ShiftBuffer (char *buffer, int *buflen, int msglen)
{
    int shiftlen;

    /* CALCULATE THE AMOUNT OF DATA TO BE SHIFTED OUT OF THE BUFFER. */
    shiftlen = msglen + GTS_SOCKET_HEADER;

    /* SHIFT THE 'PROCESSED' DATA OUT OF THE BUFFER BY MOVING THE
    UNPROCESSED DATA OVER THE TOP OF IT.
    CALCULATE THE NEW AMOUNT OF DATA IN THE BUFFER. */
    *buflen = *buflen - shiftlen;
    memcpy (buffer, buffer + shiftlen, *buflen);
}

```

3. Some security arrangements for small GTS Centres

This Appendix provides information on low-cost measures to secure GTS centres, when they are connected to the Internet. The traditional GTS with Message Switching Systems passing bulletins over point-to-point circuits is inherently secure, while the Internet is inherently insecure. So, it is important to prevent Internet users from being able to traverse GTS links, where they may be able to cause damage to neighbouring centres.

Security policy

In a mixed Internet/GTS environment, a security hole at a GTS centre may compromise other GTS segments. It is very likely that sooner or later, most of the GTS centres will be connected to the Internet, so a solution for the security aspects must be found, which are practical for all Centres and especially, small centres. Different levels of network security can be achieved with the tools available today at costs that vary from a few dollars (freeware) to expensive proprietary solutions.

There are, however, other important aspects besides costs. Technical expertise and strong management support are indispensable elements to implement and enforce an effective security policy. Without them, no protection will be achieved, even with the best and most expensive firewall systems.

Coexistence of Internet and dedicated GTS links

There are many options for the general configuration of systems in small Centres. Figure 1 shows an arrangement where separate routers are used to provide access to Internet and GTS.

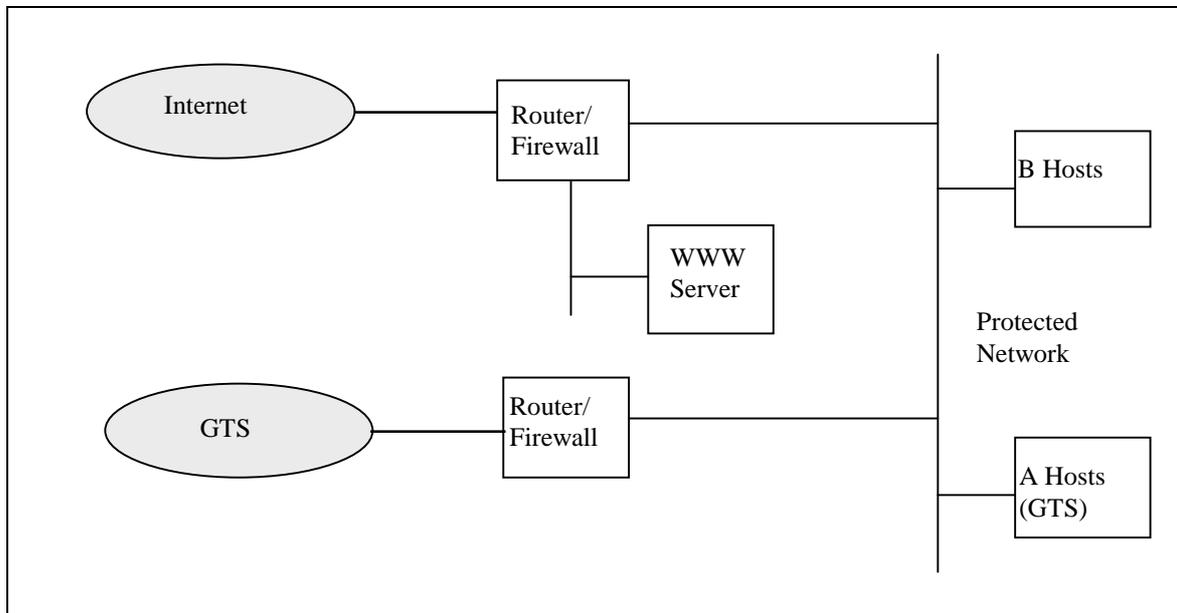


Figure 1 - Coexistence of GTS and Internet – separate access routers

To achieve cost reduction, GTS centres, may wish to consolidate the GTS and Internet networks, whilst still providing a level of security for their GTS systems. Figure 2 depicts a low cost configuration that may meet this objective.

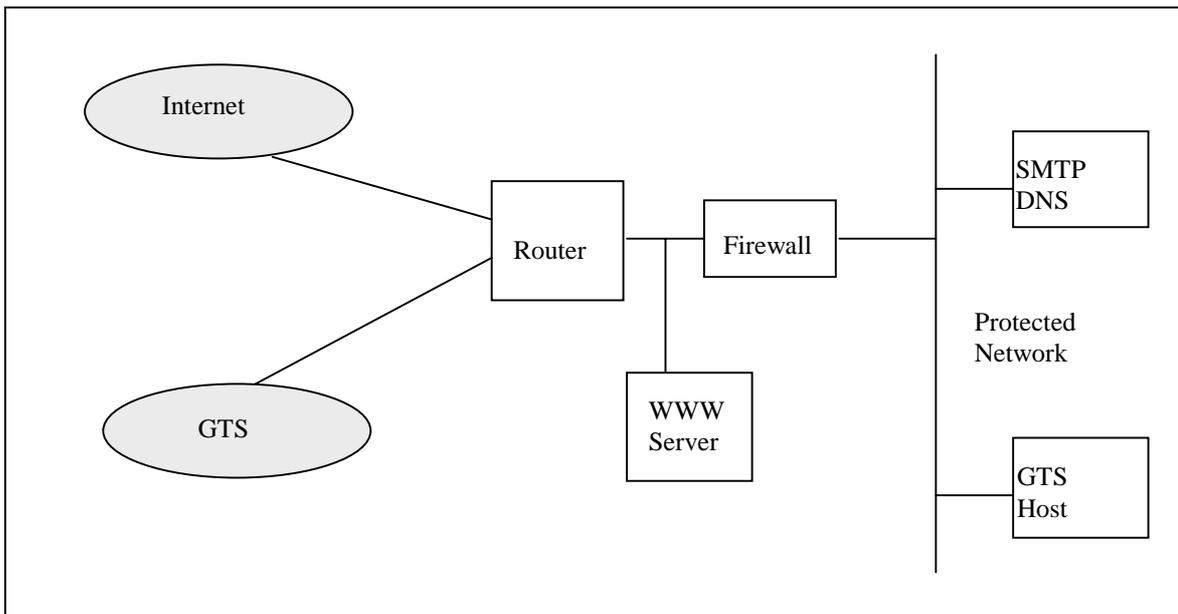


Figure 2 - Coexistence of GTS and Internet – common access router

Protecting the GTS links from the Internet

It is important that the exterior router(s), connected to both GTS and Internet links be securely passworded, and protected so that it may not be configured via the Internet. Additionally, no Internet traffic should be allowed to propagate down GTS links, nor GTS traffic be sent to the Internet unless specifically intended to do so. This can be achieved by carefully filtering routing updates.

There should be a definite separation between general Internet services (www/http, e-mail access) and the GTS system (e. g. Message Switch). They should be in separate machines. Additionally, use of firewalling technology should be undertaken to limit general Internet access to the GTS Centre internal network, possibly restricting incoming connections to SMTP on the mail server, HTTP on the web server and DNS on the Domain Name servers.

In between the exterior router and critical systems, a firewall should be deployed. This firewall must have the capacity to limit, proxy or redirect access to internal hosts in order to protect them. Several brands of firewall are on the market, with ranges of capabilities. In most cases because of the simple nature of the network in small centres, a simple firewall may be deployed.

When connecting to the Internet, deploying some sort of firewall is virtually mandatory. The risks for internal data and systems would justify this. In order to allow the access control some low-cost options are available:

Linux computers

Linux Operating system is free, and runs on a variety of hardware platforms, notably on PCs. The newest versions of Linux (Kernel version 2.2) come with firewalling software called *ipchains*. Additionally, they support routing protocols through a routing program called *gated*. Centres with some experience with UNIX will be able to get a working firewall setting up linux from scratch.

Windows NT

A variety of commercial packages exist. The familiarity with the Windows and relatively low cost of PC hardware is seen as main advantage.

Free Toolkits

A company called TIS (Trusted Information Systems) has released a set of source code, mainly for UNIX/LINUX hosts, which is freely available. This requires access to UNIX/LINUX machines, compilers, and requires good knowledge of Internet security issues.

Routers

Many routers have packet filtering capabilities. It is possible to deploy one of these as firewall, although they aren't very flexible. Smaller centres may want to consider this.

Desirable solution

Some firewall vendors have been providing firewall solutions, based on their hardware. Cisco's IOS firewall is a notable example. This type of solution is depicted in figure 3.

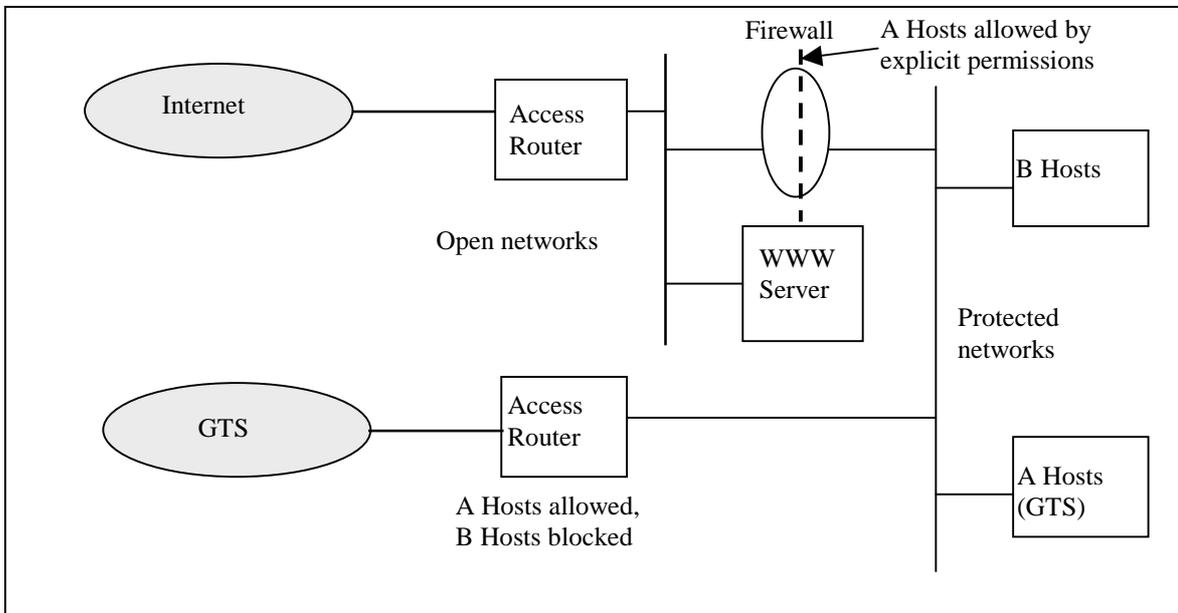


Figure 3 - Coexistence of GTS and Internet – separate access routers plus firewall

GTS using the Internet

There will be situations where GTS centres will use Internet to transport data and products. Security concerns are also applicable here. The arrangement shown below in Figure 4 represents a simple and safe way to use the Internet to connect neighbouring GTS centres that may become popular in small centres in future. Firewalling is done using access lists.

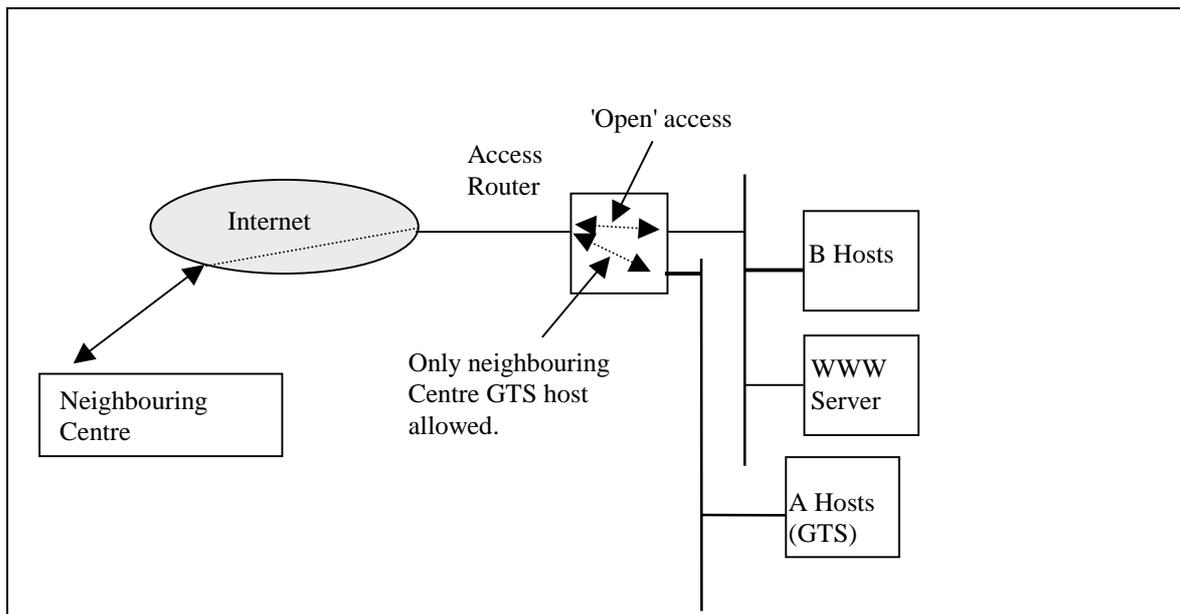


Figure 4 - The use of Internet between neighbouring GTS centres.

4. Reference material

General references on TCP/IP

1. Internetworking TCP/IP Vol. 1 (2/E) - Douglas Comer - Prentice Hall
2. TCP/IP Illustrated Vol. 1. - Stevens - Addison-Wesley
3. TCP/IP Architecture, Protocols and Implementation - Feit - McGraw Hill
4. TCP/IP and Related Protocols - Black - McGraw Hill
5. TCP/IP Running a Successful Network - Washburn and Evans - Addison-Wesley
6. TCP/IP and ONC/NFS (2/E) - Santifaller - Addison-Wesley
7. Inside TCP/IP - Arnett et. al. - New Riders Publishing
8. Teach Yourself TCP/IP in 14 days - Parker - SAMS
9. Introduction to TCP/IP - Davidson - Springer

References on Security

1. Firewalls and Internet Security- Cheswick & Bellovin - Addison-Wesley
2. Building Internet Firewall – Chapman – O'Reilly
3. Practical Unix Security - Garfinkel & Spafford - O'Reilly
4. Internet RFC 2196 (Site security Handbook)
5. <http://www.securityportal.com> : a web site with a lot of reference documents on implementing security (include Cisco routers, Windows NT, Linux boxes, and various flavours of Unix)

5. Suggested password management practices

Passwords are the system's first line of defence against unauthorised intrusion. While it is possible to violate system security without logging in, a poorly protected or chosen password can make a hacker's task a lot easier.

GOOD PASSWORDS:

1. Have both upper-case and lower case letters, and/or
2. Have digits and/or non-alphanumeric characters.
3. Are 6 to 8 characters long.
4. Should consist of at least 2 words or groups of characters.
5. Should not be shared or used by more than one user.
6. Should not be used on more than one computer.
7. Should be changed regularly, eg monthly.
8. Can be typed quickly and easily, so that an observer cannot follow the keystrokes.
9. Are easy to remember - so that they should not have to be written down. (e.g. use first letter of words in a well known phrase)

BAD PASSWORDS:

1. The name of: yourself, your spouse, your children, your parents, your pet, your friends, your favourite film stars/characters, anyone associated with you, your workstation or its host.
2. The number of: your telephone, your car's license plate, your user ID, any part of your credit cards' numbers, or any number associated with you.
3. The birthday of yourself or anyone associated with you.
4. Any word from any dictionary, any place name, any proper noun.
5. The name of a well known public identity such as a sporting hero, entertainer or well known fictional character.
6. Simple patterns: aaaaaaa, qwerty.
7. Any of 1 - 6 spelled backwards.
8. Any of 1 - 6 preceded or followed by a digit.
9. Any password that has been written down and left in an unlocked drawer or unsecured computer file.
10. Any password that has been on a machine that may have been successfully hacked (except as part of authorised exercises).
11. Any password on a machine that has been left unattended when any user is logged on.