Internet Protocol Static Routing

External Reference Specification

November 13, 1985

AUTHOR: _____
John R. Lyman III

TDRB APPROVAL: _____

AD&C APPROVAL: _____

INSTRUCTIONS

ecause all our documentation is being eproduced on reduction, it is ssential that this form is either YPED OR PRINTED LEGIBILY IN BLACK INK.

ill out only those fields marked by ullets (o).

OCUMENT TYPE: One of the following:
AR - Archictecture Requirments (CYBER 80)
BSL - Baseline Documentation Change(s)
DAP - Design Action Paper
DDD - Design Direction Document
DR - Design Requirements
EPP - Evalutaion Project Plan
ERS - External Reference Specifications
FTP - Feature Test Plan
GDS - General Design Spec (CYBER 80)
GID - General Internal Specifications
IHB - Installation Handbook
IPP - Implementation Project Plan
PDA - Product Development Assurance Plan
PJP - Project Plan
STD - Standards Document
STP - System Test Plan
TIR - Transmittal, Integration & Release Criteria
OTH - Other, specify type

ITLE: Be concise. Be sure to include the name of the product involved.

BSTRACT: A brief description of the contents. If the item is extremely short, it may be written in designated space with nothing else attached.

UBLICATION #: If a document with a pub. # is being changed, write the pub. # here.

IP & PRM Numbers: If any are associated with this document, list them here.

PPROVALS: (Minimum DCS Requirements)
Project leader for all.
Unit Manager for all.
Section Manager: Project Plans only.
DCS will assign referee.

ROJECT: The project name that the author belongs to, e.g., CLUE.

CS Mailing List Pasted Here -
(On top of instructions)

*For ARHOPS Use Only!

CONTROL DATA CORPORATION

DOCUMENT CONTROL FORM

o DOCUMENT TYPE: ERS

o LOG ID: ARH7118

o TITLE: INTERNET PROTOCOL STATIC ROUTING ERS

o ABSTRACT: This document describes the external design of the CDCNET IP static routing.

o PRODUCT AFFECTED: CDCNET

o AUTHOR: J. R. Lyman

o MAIL STATION: ARH207 o EXTENSION: 3087

o PROJECT: IP/TCP/Telnet - CSU Joint Venture

o SECTION: ACSD

o RSM or PSR #

o SIP Number(s)

o PUB #

o PRM #

o Redesign* o Reimplementation*

	Name	Approval Initials	Date
o Internal Reviewer*			
o Project Leader			
o Unit Manager	T. C. McGee	TCM	12/13/85
o Section Manager			
Design Team Referee	B. S. Sekhon	BSS	12-10-85

DESIGN TEAM: CDCNET DEFAULT CYCLE: 0 WORKING DAYS

o Special Distribution: M. R. Sanders SVL069

Referee's Distribution Codes: AC-B

NUMBER	DATE	TYPE	SUBMITTED BY	DEFAULT APPROVAL DATE	CURRENT STATUS: REVIEW	HOLD	APPROVED	WITHDRAWN
1	12-12-85	Orig	J. R. Lyman	None			A	

Please submit your comments to DCS, SVL107 or ARH230 (as appropriate) before the default approval date.

Table of Contents

1.0 INTRODUCTION 1-1
1.1 PURPOSE 1-1
1.2 REFERENCES 1-1

2.0 SERVICE OVERVIEW 2-1
2.1 SERVICES PROVIDED 2-2
2.1.1 PROCEDURE ACCESS 2-2
2.1.2 ROUTING SERVICES 2-4
2.1.3 COMMAND ACCESS 2-4
2.2 FUNCTIONAL RELATIONSHIPS 2-5
2.3 EXTERNAL SERVICES UTILIZED 2-6
2.3.1 INTRANET LAYER (3A) 2-6
2.3.2 INTERNET PROTOCOL (IP) 2-6

3.0 SERVICE DESCRIPTIONS 3-1
3.1 PROCEDURE ACCESS 3-1
3.1.1 ADD HOST FUNCTION 3-1
3.1.2 DELETE HOST FUNCTION 3-3
3.1.3 PROCESS ICMP DATA 3-4
3.1.4 ADD NETWORK FUNCTION 3-5
3.1.5 DELETE NETWORK FUNCTION 3-6
3.1.6 SEARCH_HOST_TABLE 3-7
3.1.7 SEARCH_NETWORK_TABLE 3-8
3.2 ROUTING SERVICES 3-9
3.2.1 IP ROUTING FUNCTION 3-9
3.3 COMMAND INTERFACE 3-11
3.3.1 DEFINE_IPSR_HOST 3-11
3.3.2 CANCEL_IPSR_HOST 3-13
3.3.3 DEFINE_IPSR_NETWORK 3-14
3.3.4 CANCEL_IPSR_NETWORK 3-16
3.4 ERROR RECOVERY 3-17

4.0 PERFORMANCE 4-1
4.1 OPERATING CHARACTERISTICS 4-1
4.2 OPERATIONAL MEASUREMENTS 4-2
Summary report 4-2
Expanded report 4-2

5.0 FINITE STATE MACHINE 5-1

6.0 LOG MESSAGES 6-1
6.1 INTERNAL ERROR CONDITION 6-1

7.0 INSTALLATION OPTIONS 7-1

8.0 NEW DATA TYPES 8-1

9.0 GLOSSARY 9-1

PROPRIETARY NOTICE

The technical content set forth in this document is the property of Control Data and is not to be disseminated, distributed or otherwise conveyed to third parties without the express written permission of Control Data.

This document is to be used for planning purposes only. It does not include any explicit or implied commitments for any specific release dates or feature content. These matters are currently under active development, and as such are subject to revision and replanning as circumstances warrant.

RECORD OF REVISION

Revision	Description	Author	Date
01	Draft Version	JRL3	01/28/85
02	Revised due to TDRB comments	JRL3	05/20/85
03	Revised due to TDRB comments	JRL3	10/23/85
04	Revised due to TDRB comments	JRL3	11/13/85

1.0 INTRODUCTION

1.0 INTRODUCTION

1.1 PURPOSE

This document describes the external interfaces of the internet protocol static routing (IPSR) module. This module contains the routing tables that are used by all other DoD protocols. Routines are provided to maintain the list of directly connected hosts kept in the routing tables. Command processors are also provided to maintain this list of directly connected hosts. The major purpose of this module is fulfilled by a routine provided to determine the next hop for a given datagram.

The routines contained in this module are not intended to be the only users of the routing tables, any dynamic routing protocol (EGP for instance) may access the routing tables directly.

1.2 REFERENCES

The following manuals contain material that either defines the operations of the IPSR module and the modules it interfaces to, or provides additional insight into the use of the module.

[1] RFC-791 SRI Internet Protocol
[2] RFC-792 SRI Internet Control Message Protocol
[3] MIL-STD-1777 DoD Internet Protocol Standard
[4] ARH6265 CDC Internet Protocol ERS
[5] ARH7016 CDC Internet Protocol IDS

2.0 SERVICE OVERVIEW

In the DoD network environment the following characteristics control the routing of datagrams. It should be noted that in the CDCNET environment the distinction between hosts and gateways is not adhered to, the IP module performs both functions.

1. A DoD host is by definition connected to only one network.

2. A DoD gateway is by definition connected to two or more networks.

3. The routing tables used by a host are also used by a gateway and include two tables. The first contains a list of all directly connected hosts and gateways, and their local addresses. The second contains a list of all reachable networks and the local address of the gateway to each network.

4. As each host or gateway is added to a DoD network, its existence must be manually indicated to every other entity connected to that network. There is no protocol that a new host can use to automatically inform its neighbors of its existence.

5. The network table used by a host is only updated by redirect messages. Hosts do not participate in any routing protocols.

6. The network table used by a gateway may be updated dynamically by various routing protocols, with EGP and ~~GGP~~ IGP as examples. Additional tables may be used by these routing protocols that do not appear in a host.

2.0 SERVICE OVERVIEW
2.1 SERVICES PROVIDED

2.1 SERVICES PROVIDED

The static routing module provides services to allow the modules which implement other DoD protocols in the DI to function according to the characteristics noted above. There are three base services provided, each service is described in one of the following sections.

2.1.1 PROCEDURE ACCESS

The IPSR module provides other modules with a set of subroutines that can be called to modify the contents of the routing tables. These routines are provided to minimize the amount of information that an outside user needs to know about the routing tables.

ipsr_add_host	This routine allows an outside module to add a host to the routing table. The various addresses of the host must be specified, as well as the type of host.
ipsr_delete_host	This routine allows an outside module to delete a host from the routing table. The IP address and type of the host must be specified.
ipsr_process_icmp_data	This routine allows the IP module to inform the IPSR module that it has received an ICMP datagram.
ipsr_add_network	This routine allows an outside module to add a new network to the network table.
ipsr_delete_network	This routine allows an outside module to delete a network from the network table.
ipsr_search_host_table	This routine allows an outside module to locate a specific host and determine all

	known information about that host.
ipsr_search_network_table	This routine allows an outside module to locate a specific network and determine all known information about that network.

2.1.2 ROUTING SERVICES

The IPSR modules primary task is to provide routing decisions to the IP module This task is handled by the ipsr_get_route routine. The ipsr_get_route routine is called by the IP module to determine where a datagram should be sent. Each datagram that the IP module receives from the upper layer or the lower layer will be processed to determine the next destination. The routing options that are part of the datagram will be processed by this routine.

2.1.3 COMMAND ACCESS

There are four command processors which are logically part of the IPSR module. The first two allow the list of directly connected hosts/gateways to be modified, and the second two allow the list of known networks to be modified. These command processors will call the routines that are provided to the rest of the system to perform their functions. This service is intended to allow the network operator to modify the routing tables manually.

Define_IP_host	This command allows the addition of a host to the routing table. The various addresses of the host must be specified, as well as the type of host.
Cancel_IP_host	This command allows the deletion of a host from the routing table. The IP address of the host must be specified.
Define_IP_network	This command allows the addition of a network to the routing table.
Cancel_IP_network	This command allows the deletion of a network from the routing table.

2.2 FUNCTIONAL RELATIONSHIPS

The services of the IPSR module may be accessed by anyone, at any time, no SAP is needed. All routines are global addresses. The routing tables do not have any locks. It is assumed that all users of the routing tables will be running as a non-preemptive tasks, but if the IPSR module detects that its user is preemptive then the NOPREMPT and OKPREMPT common subroutines will be called.

2.3 EXTERNAL SERVICES UTILIZED

2.3.1 INTRANET LAYER (3A)

The IPSR module uses the services of the 3A module. The IPSR module requires the ability to differentiate between CDC network solutions and IP network solutions. When the IPSR module opens a SAP with the 3A module it will specify that it is only concerned with IP network solutions. The IPSR module will specify a routine that 3A can call to inform it that the status of a network has changed and will expect to receive the following information from the 3A module.

1. The IP address of the network connection.
2. The 3A network ID.
3. The 3A system ID.
4. The maximum datagram size for the network.
5. The current status of the network.

The IPSR module will use the information that it receives in a 3A status indication to create a network entry for the directly connected network and a host entry for its own address on the particular network.

2.3.2 INTERNET PROTOCOL (IP)

The IPSR module expects that the IP module will present an indication of each Internet Control Message Protocol (ICMP) datagram that it receives. The format of ICMP datagrams is documented in reference [2]. By processing ICMP datagrams the IPSR module will be able to keep the network table up to date in a somewhat dynamic manner.

3.0 SERVICE DESCRIPTIONS

3.0 SERVICE DESCRIPTIONS

3.1 PROCEDURE ACCESS

The DoD protocols do not at this time provide any method for the IPSR module to obtain the addresses of directly connected hosts. It is therefore necessary for some other module which knows these addresses to inform the IPSR module. The routines described in this section allow the caller to add to and delete from the routing table. There are also two routines which allow reachable networks to be added to and deleted from the routing table.

3.1.1 ADD HOST FUNCTION

This routine is used to add the information about a directly connected host into the routing table. Each IP host on a directly connected IP network must be entered into the routing table with this routine. All hosts in a connected CDC Catnet to whom this gateway provides direct service, are considered to be directly connected and must be entered into the routing table with this routine. The format of the CYBIL interface is as follows:

```
PROCEDURE ipsr_add_host (
   host_address : ip_address;
   host_type    : ipsr_host_type;
   egp_active   : BOOLEAN;
   igp_active   : BOOLEAN;
   3a_network   : net_id_type;
   3a_system    : sys_id_type;
   VAR status   : ipsr_status_type);
```

host_address	In	This is the IP address of the host being defined.
host_type	In	This is the type of host being defined.
egp_active	In	This flag is set if an external

3.0 SERVICE DESCRIPTIONS
3.1.1 ADD HOST FUNCTION

		gateway protocol is active in this host.
igp_active	In	This flag is set if an internal gateway protocol is active in this host.
3a_network	In	This is the 3A net_id of the network solution that the host is connected to.
3a_system	In	This is the 3A system_id of the host.
status	Out	This is the status of the request. The following values may be returned: ipsr_successful ipsr_host_exists ipsr_invalid_address ipsr_invalid_type ipsr_unknown_network

3.1.2 DELETE HOST FUNCTION

This routine is used to delete a host from the routing table. When an existing entry needs to be changed, the current entry must be deleted and then the new entry added. The format of the CYBIL interface is as follows:

```
PROCEDURE ipsr_delete_host (
   host_address : ip_address;
   VAR status   : ipsr_status_type);
```

host_address	In	This is the IP address of the host.
status	Out	This is the status of the request. The following values may be returned: ipsr_successful ipsr_unknown_host ipsr_invalid_address

3.1.3 PROCESS ICMP DATA

This routine is called by the IP module to indicate that it has received an ICMP datagram. This is the only dynamic routing information that a standard DoD host will receive. The format of the CYBIL interface is as follows:

```
PROCEDURE ipsr_process_icmp_data (
   error_type  : ipsr_icmp_ind;
   source      : ip_address;
   destination : ip_address;
   new_gateway : ip_address);
```

error_type	In	This is the type of ICMP datagram that is being reported.
source	In	This is the source address from the datagram that the ICMP datagram pertains to.
destination	In	This is the destination address from the datagram that the ICMP datagram pertains to.
new_gateway	In	This is the IP address of the gateway that the datagram should have been sent to.

3.1.4 ADD NETWORK FUNCTION

This routine is used to add information about a reachable network into the routing table. This routine will be used at first by the command processor which adds networks. It may also be used by modules that provide dynamic routing services. The format of the CYBIL interface is as follows:

```
PROCEDURE ipsr_add_network (
   network    : 0..0FFFFFF(16);
   gateway    : ip_address;
   hop_count  : INTEGER;
   owner      : ipsr_net_owners;
   VAR status : ipsr_status_type);
```

network	In	This is the network number of the network that is being added to the routing table.
gateway	In	This is the IP address of the gateway that datagrams destined for this network should be sent to.
hop_count	In	This is the number of gateways that the datagrams must pass through to reach this network.
owner	In	This is the owner of the table entry. Dynamic routing modules may use this value to insure that they only delete entries that they added.
status	Out	This is the status of the request. The following values may be returned: ipsr_successful ipsr_invalid_owner ipsr_network_exists ipsr_unknown_gateway

3.1.5 DELETE NETWORK FUNCTION

This routine is used to delete information about a reachable network from the routing table. This routine will be used at first by the command processor which removes networks. It may also be used by modules that provide dynamic routing services. The format of the CYBIL interface is as follows:

```
PROCEDURE ipsr_delete_network (
   network    : 0..0FFFFFF(16);
   owner      : ipsr_net_owners;
   VAR status : ipsr_status_type);
```

network	In	This is the network number of the network that is being removed from the routing table.
owner	In	This is the owner of the table entry. Dynamic routing modules may use this value to insure that they only delete entries that they added.
status	Out	This is the status of the request. The following values may be returned: ipsr_successful ipsr_invalid_owner ipsr_unknown_network

3.1.6 SEARCH_HOST_TABLE

This routine will search the network/host table; the host_address will be used as the key. If an entry is found then a pointer to the host information record will be returned, else the pointer will be NIL.

```
PROCEDURE ipsr_search_host_table (
   host_address   : ip_address;
   VAR host_entry : ↑ipsr_host_info);
```

host_address	In	This is the address of the host that we are searching for.
host_entry	Out	This is a pointer to the information about the host if its found in the table. If the host is not found then this pointer is returned as NIL.

3.1.7 SEARCH_NETWORK_TABLE

This routine will search the network table. The network number will be used as the key. If an entry is found then a pointer to the network information record will be returned, else the pointer will be returned NIL.

```
PROCEDURE ipsr_search_network_table (
   network_number : 0..FFFFFF;
   VAR network_entry : ↑ipsr_network_info);
```

network_number	In	This is the network number of the network that we are searching for.
network_entry	Out	This is a pointer to the information record of the network being searched for, if the network is not found then the pointer will be returned NIL.

3.2 ROUTING SERVICES

The primary purpose of the IPSR module is the routing service that it provides to the local system. This service is provided by a single routine which accepts the header and options of an IP datagram and determines the next host/gateway on the datagrams route.

3.2.1 IP ROUTING FUNCTION

This routine is called by the IP module when a datagram is received. The IPSR module uses the destination address, the source address, and the ip routing options to determine the best place for the datagram to go next. The source address is optional and the IPSR module will determine it and return it to the IP module if not specified. The 3A network_id and the 3A system_id of the local destination are returned to the IP module along with the type of destination. This routine will also process the IP routing options, if there are any, and updates them according to the IP protocol. The format of the CYBIL interface is as follows:

```
PROCEDURE ipsr_get_route (
   from_network      : BOOLEAN;
   header            : ip_header;
   VAR source        : ip_address;
   destination       : ip_address;
   VAR options       : ip_option_record;
   VAR host_info     : ipsr_host_rec;
   VAR max_data_size : INTEGER;
   VAR status        : ipsr_status_type);
```

from_network	In	This flag should be TRUE if the datagram that is being routed was received from a network.
header	In	This is the header of the IP datagram.
source	I/O	This is the IP address of the source, this should be the address of the DI on a directly connected

		network. This parameter is optional and will be chosen by the IPSR module if not specified.
destination	In	This is the destination of the IP datagram.
options	I/O	This is an array which contains the IP routing options.
host_info	Out	This is a record which contains the routing information that the IP module requires.
max_data_size	Out	This is the maximum number of bytes that a datagram being sent to the indicated destination can contain.
status	Out	This is the status of the request. The following values may be returned: ipsr_successful ipsr_unable_to_route

3.3 COMMAND INTERFACE

The DoD protocols do not at this time provide any method for the IPSR module to obtain the addresses of directly connected hosts. It is therefore necessary for these addresses to be entered into the routing table manually. This management of the routing tables can be done through the procedure interface described in the previous sections, or it may be done by the network operator with the commands described in this section Two of the commands described in this section allow the user to add to, and delete from the list of directly connected hosts and gateways. Unless another module is provided to do dynamic routing, ICMP data will be the only automatic way to add networks to the routing table. Two commands are therefore provided to allow the network operator to add reachable networks into the routing table manually.

3.3.1 DEFINE_IPSR_HOST

This command will enter the information about a directly connected host into the routing tables. If the address is not that of a directly connected network an error message will be returned.

Command_name:

Define_IPSR_Host, DEFIH

Parameters:

```
I_P_address, ipa      : LIST 4 OF 0..255

host_type, ht         : KEY
                          ip_host
                          ip_gw
                          cdc_host
                          cdc_gw
                          local = local

network_3_A, n3a      : INTEGER
```

3.0 SERVICE DESCRIPTIONS
3.3.1 DEFINE_IPSR_HOST

0..0FFFFFFFF = 0

host_3_A, h3a : INTEGER
0..0FFFFFFFF = 0

Parameter descriptions:

I_P_ADDRESS	This is the IP address of the host expressed in the format used by SRI.
HOST_TYPE	This is the type of host being defined.
NETWORK_3_A	This is the 3A network id.
HOST_3_A	This is the 3A system id.

Responses:

Addition of host completed.
Address already in use.
Invalid IP address.
Invalid 3A address.
Invalid 3B address.
Invalid type of host.

Example:

```
DEFINE_IPSR_HOST
   IPA=5 123 44 76
   HT=IP_GW
   N3A=09C4
   H3A=367D956F
```

3.3.2 CANCEL_IPSR_HOST

This command will remove the information about a directly connected host from the routing tables.

Command_name:

Cancel_IPSR_Host, CANIH

Parameters:

I_P_address, ipa : LIST 4 OF 0..255

Parameter descriptions:

I_P_ADDRESS	This is the IP address of the host expressed in the format used by SRI.

Responses:

Removal of host completed.
Host not defined.
Invalid IP address.

Example:

```
CANCEL_IPSR_HOST
   IPA=5 114 244 141
```

3.3.3 DEFINE_IPSR_NETWORK

This command will enter the information about a reachable network into the routing tables.

Command_name:

Define_IPSR_Network, DEFIN

Parameters:

Network_number, nn	:	INTEGER 0..0FFFFFF(16)
G_W_address, gwa	:	LIST 4 OF 0..255
Hop_count, hc	:	INTEGER 0..0FFFF(16)

Parameter descriptions:

NETWORK_NUMBER	This is the network number of the network that is being added to the routing table.
G_W_ADDRESS	This is the IP address of the gateway that datagrams should should be directed to, expressed in the format used by SRI.
HOP_COUNT	This is the number of gateways that a datagram must pass through in order to reach this network.

Responses:

Addition of network completed.
Network already in use.
Invalid hop count.

3.0 SERVICE DESCRIPTIONS
3.3.3 DEFINE_IPSR_NETWORK

Unknown gateway.

Example:

```
DEFINE_IPSR_NETWORK
   NN=1823D5
   GWA=5 23 123 12
   HC=4
```

3.3.4 CANCEL_IPSR_NETWORK

This command will remove the indicated network from the routing table.

Command_name:

Cancel_IPSR_Network, CANIN

Parameters:

Network_number, nn	:	INTEGER 0..0FFFFFF(16)

Parameter descriptions:

NETWORK_NUMBER	This is the network number of the network that is being removed from the routing table.

Responses:

Network removed.
Unknown network.

Example:

```
CANCEL_IPSR_NETWORK
   NN=128AF
```

3.4 ERROR RECOVERY

The IPSR module will test all input parameters for legality before attempting to use them. In each routine there will be tests for those "impossible situations" that usually indicate that something unexpected has gone wrong. If one of these unexpected errors is found then an appropriate log message will be generated and an attempt will be made to pick up where we left off.

4.0 PERFORMANCE

4.0 PERFORMANCE

4.1 OPERATING CHARACTERISTICS

The IPSR module is called once for each datagram processed by the IP module. It is therefore obvious that the time it takes to determine the next stop for a datagram has a large impact on the processing time of a datagram. The routing options must be as fast as possible, therefore, the routing tables will be designed to minimize routing time. The time needed to make updates to the routing tables will be considered relatively unimportant. The major goal for the routing routine is that the ipsr_get_route should not take more than 100+(10*log2(N))+(10*log2(H)) assembler instructions for N networks with a maximum of H hosts per network.

4.2 OPERATIONAL MEASUREMENTS

The IPSR module does not keep any statistics, however, the information in the routing tables can be displayed by the CDNA status command processor. The content and basic format of the information provided is shown below.

Summary report

IP Static Routing Table Status

```
IP_hosts  = ####        Reachable networks = ####
IP_gws    = ####        Connected networks = ####
CDC_hosts = ####
CDC_gws   = ####
Locals    = ####
```

Expanded report

IP Static Routing Table Status

Directly Connected Hosts

Type	IP Address	CDC Address
IP_host	001.011.047.167	000000000001
IP_gw	127.001.000.203	000000000002
CDC_gw	003.000.034.054	000001D3-00000000000A
CDC_host	192.000.00A.024	
Local	004.037.242.108	

Reachable networks

Network	Gateway Address	Hops	Owner
003	003.000.034.054	3	SRM
127.003	127.001.000.203	34	IGP
192.000.003	127.001.000.203	7	EGP

5.0 FINITE STATE MACHINE

The IPSR module does not need an FSM it has no external protocol.

6.0 LOG MESSAGES

6.0 LOG MESSAGES

The IPSR module issues the following two log messages to indicate that an error has occurred in the IPSR module.

6.1 INTERNAL ERROR CONDITION

Message Id: ???

Descriptive Message: "IPSR Internal Error"

Fixed Text	Type	Value	Description
"Procedure: "	Character Octet	1..30 Octets	Name of the procedure the error occurred in.
"Description: "	Character Octet	1..60 Octets	Short description of the error that occurred.

Example: 85/01/28 15.47.00 1234567890ABC
IPSR Internal Error
Procedure: Get_route
Description: Bad table pointer

7.0 INSTALLATION OPTIONS

7.0 INSTALLATION OPTIONS

The IPSR module does not require any installation options. The only data contained in the module is in the routing tables and these tables are initialized to be empty.

8.0 NEW DATA TYPES

8.0 NEW DATA TYPES

The IPSR module uses a number of ordinals. The status returned by its routines is an ordinal, and there are other identifiers that are defined as ordinals. The CYBIL definition of the various ordinals follows.

```
CONST
   unreachable_hop = -1,

   ipsr_net_unreachable = 20,
   ipsr_source_quench = icmp_source_quench,
   ipsr_redirect = icmp_redirect,
   ipsr_time_exceeded = icmp_time_exceeded;


TYPE
   ipsr_status_type = (
      ipsr_successful,          { Function completed.
      ipsr_host_exists,         { The host is in the list.
      ipsr_invalid_address,     { Illegal host address.
      ipsr_invalid_owner,       { Owner parameter incorrect.
      ipsr_invalid_type,        { Illegal host type.
      ipsr_network_exists,      { Network already exists.
      ipsr_unable_to_route,     { No route is available.
      ipsr_unknown_gateway,     { Gateway not in the list.
      ipsr_unknown_host,        { No such host in the list.
      ipsr_unknown_network),    { No such net in the list.

   ipsr_host_type = (
      ip_host,       { IP host on an IP network.
      cdc_host,      { CDC host in a CDC Catnet.
      ip_gw,         { IP gateway on an IP network.
      cdc_gw,        { IP gateway in a CDC catnet.
      local,         { IP address of this gateway.
      none),

   ipsr_icmp_ind = (
      ipsr_net_unreachable, ipsr_source_quench,
      ipsr_redirect,        ipsr_time_exceeded);
```

8.0 NEW DATA TYPES

```
ipsr_net_owners = (
   all,              { Everyone owns this entry.
   ipsr,             { The routing module owns it.
   egp,              { The External Gateway owns it.
   igp),             { The Internal Gateway owns it.

ipsr_connect_status = (
   unknown,            { The status is unknown.
   available,          { Status up and available.
   unavailable,        { Status up but unavailable.
   down),              { Status down.

ipsr_network_type = (
   direct,           { Directly connected.
   remote);          { Connected thru a gateway.
```

8.0 NEW DATA TYPES

The information kept by the routing module includes networks and hosts/gateways. The structure used is a nested binary table. The first level table is keyed by network number. Each entry describes a known network. IF the network is directly connected then there will be a field which points to a second level table which contains all of the hosts/gateways that reside on the network; the host number being used as the key.

```
TYPE
   ipsr_network_info = RECORD
      status : ipsr_connect_status,
      CASE net_type : ipsr_network_type OF
         =direct=
            local_address : ip_address,
            max_data_size : INTEGER,
            3a_network    : net_id_type,
            3a_system     : sys_id_type,
            host_root     : root,
         =remote=
            next_hop  : ip_address,
            hop_count : INTEGER,
            owner     : ipsr_net_owners,
            timeouts  : INTEGER,
      CASEND,
   RECEND,

   ipsr_host_info = RECORD
      host_type  : ipsr_host_type,
      egp_active : BOOLEAN,
      igp_active : BOOLEAN,
      3a_system  : sys_id_type,
      status     : ipsr_connect_status,
   RECEND;


VAR
   default_gateway    : ip_address,
   ipsr_routing_table : root;
```

9.0 GLOSSARY

9.0 GLOSSARY

The following term definitions are provided to aid the reader in understanding this document.

CDNA: Control Data Distributed Network Architecture.

Datagram: This is a block of ULP data, and is the unit of data processed by the IP module.

DoD: Department of Defense.

IP: DoD Internet Protocol. This is the level three protocol used by the DoD. It provides the serves of an OSI internet layer.

OSI: Open System Interconnect. This is a model developed by the International Standards Organization (OSI) which defines a method of layering modules within networking software.

Protocol: Each user of the IP module is identified by a specific protocol identifier. This identifier is referred to throughout this document as the protocol.

SAP: Service Access Point. A SAP is defined in this manual as the point of access to the services that a module provides. In the case of the IPSR module this point of access is defined by a set of routine address provided by the IPSR module through the open_sap request.

TCP: DoD transmission control protocol. This is the transport protocol used by the DoD. TCP provides an error free orderly end to end data transmission.

UDP: DoD user datagram protocol. This is a protocol which provides an single datagram transfer service with no promises.

9.0 GLOSSARY

ULP: User Level Protocol. Used at various times to indicate the user of the module being discussed.