APPENDIX B

Baseline Integrity Checks

This appendix describes the currently implemented baseline integrity checks. The Connectivity Tools categorize feedback by transport modes (IP, IPX, SRB, AppleTalk, and SNA STUN) which are then further sub-categorized (possibly subjectively) as High Priority or Warning-Level. High Priority Reports are critical reports that are believed to result in *major* network problems. These problems *must* be fixed to ensure the network is fully operational and functional. Warning-Level Reports describe problems that are not considered severe, but should be fixed in order to prevent inadvertent side effects and potential network performance degradation from occurring.

IP Integrity Checks

This section describes the IP integrity checks performed by the Connectivity Tools. The checks are organized into High Priority and Warning-Level groupings and are listed alphabetically within each Report group.

IP High Priority Reports

The following IP integrity checks are performed by the Connectivity Tools. If a potential problem is uncovered while performing these checks, an IP High Priority problem report is generated and placed in the Diagnostic Report.

Addresses of an Interface in Different OSPF Areas

The **network area** OSPF commands are sequentially evaluated to determine which area an interface is to be assigned. All addresses on an interface must map to the same area.

Bad Address/Mask on Router Interface

The **ip address** *ip-address mask* **[secondary]** command sets an IP address for an interface. The *mask* restricts the network or subnet to which routing updates are broadcast or received. The mask must be compatible with the class (e.g. A, B, or C) of the interface address. This check identifies erroneous IP address/mask combinations. For example, an interface address of `198.23.56.0` and a mask of `255.255.0.0` are not compatible since a class B mask is provided for a class C address. This check also verifies that the broadcast address for a subnet was selected as a hostid.

Bad Masking in Access List

Within the **access-list** *access-list-number* {**deny**|**permit**} commands, a setting of `1` within a mask position is an indication to ignore the corresponding bit in the associated address. A `1` in the same position of the address and mask is dubious, since it is an instruction to pay attention as well as ignore the associated bit. For example, the command `access-list 99 deny 131.108.34.0 0.255.255.255` is suspect.

Duplicate Addresses

All primary and secondary addresses assigned to router interfaces via the **ip address** *ip-address mask* **[secondary]**command are checked for duplication. Duplicate addresses can result in serious problems both in a live internetwork and in the Connectivity Solver simulation of an internetwork.

Noncontiguous Mask on Router Interface

The <*mask*> parameter of the **ip address** *ip-address mask* **[secondary]** command restricts the network or subnet to which routing updates are broadcast or received. While net masks in theory can be anything, they are in practice usually required to be contiguous (i.e. all ones from the left). For example, the mask `255.7.0.0`, when converted to binary, equates to `1111 1111 0000 0111 0000 0000 0000 0000` and thus violates this convention.

Non-utilized Rule in Access List

The **access-list** *access-list-number* **{deny|permit}**commands specify a collection of rules which can be utilized for route filtering (e.g. distribution lists) and packet filtering (e.g. access-groups on interfaces). The rules are evaluated sequentially looking for a match. If a match is found, the packet or routing update is either denied/permitted according to the rule. The matching criteria for an extended access list are source address and mask, destination address and mask, destination TCP/UDP port, and protocol. A danger when creating long access lists is that preceding rules may subsume (be more general than) subsequent rules. If this happens, the later rules are never utilized. The prior rule shadows them. If `access-list 80 deny 198.0.0.0 0.255.255.255` preceded the rule `access-list 80 permit 198.65.0.0 0.0.255.255`, the second rule would never be exercised. If the access list action differs between the two rules, as in this case, a (declarative) inconsistency exists which is only resolved by virtue of the sequential (i.e. procedural) evaluation of the rules. This is classified as a high severity violation. If, on the other hand, the access list actions agree, the second rule is unnecessary and is identified as a warning. The potential for subsumption exists also in the protocol restriction (`ip` subsumes `tcp`, `udp`, `icmp`, `igrp`) and in the port restriction (`gt 0` subsumes `gt 1024`). This check performs a pair-wise comparison between every rule in the same access list.

OSPF Area Does Not Border Area Zero

An OSPF network should be designed such that all areas border area zero. If a connection to the backbone is lost, it can be repaired by establishing a virtual link. Virtual links are defined using the **virtual link** option of the **area** router subcommand. Version 1.1 of the NETSYS Connectivity Tools does *not* model the **virtual link** option, therefore, if you are using this command, you can ignore this check. Use the suppress option in the Diagnostic Report window to suppress entries related to this check from being displayed in the Diagnostic Report.

Overlapping IP Subnets

Subnets denote a range of host IDs. Subnet host-id ranges should be mutually exclusive. Host addresses must map uniquely to one, and only one, subnet. In general, an encompassing subnet range overlaps with many others. For example, a misconfigured subnet, IP address `131.108.1.2 255.255.0.0`, will encompass all legal subnets on this class B network. In the interest of reducing "chattiness", only the first overlap is flagged.

RIP/IGRP Using Variable Length Subnet Masking

The RIP and IGRP routing protocols do not support variable-length subnet mask (VLSM). You should not assign different subnet masks to the same major net.

Static Route Next Hop is a Shutdown Interface

The **ip route** *network* **[***mask***] {***address*|*interface***} [***distance***]** command has an option to specify a forwarding interface. This check identifies static route definitions that forward to a non-existent or shutdown interface.

Static Route Next Hop is Indirectly Connected

The **ip route** *network* **[***mask***] {***address*|*interface***} [***distance***]** command has an option to specify the next hop address of a *directly* connected router. A directly connected router is one hop away. This check identifies static route definitions that forward to what appears to be, an indirectly connected router interface. The problem may stem from an incompleteness in the baseline topology. For example, missing serial links (as discussed in "") may be the source of the problem.

(Sub)Net Mask Creates Hostid of Zero

The <*mask*> parameter of the **ip address** *ip-address mask* **[secondary]** command restricts the network or subnet to which routing updates are broadcast or received. By convention, the mask should not create a hostid of zero. For example, the command `ip address 158.131.67.17 255.255.255.240` describes an interface with a hostid 1 on subnet `158.131.67.16`. The command `ip address 158.131.67.16 255.255.255.240` breaks this convention, since it results in a hostid of 0. The Connectivity Solver disallows the attachment of this interface to the implied (sub)net.

Undefined Access List Referenced

The **ip access-group and distribute-list** commands require a reference to an access list. This check identifies references to undefined access lists. Depending on the IOS version, an undefined access list has implicit behavior of denying or allowing all access. In 9.x IOS releases, the behavior is *deny* all. In 10.x IOS releases, the behavior is *allow* all.

IP Warning-Level Reports

The following IP integrity checks are performed by the Connectivity Tools. If a potential problem is uncovered while performing these checks, an IP Warning-Level problem report is generated and placed in the Diagnostic Report.

Bad Default Network Specification

The **ip default-network** *network-number* command allows the specification of a catch-all forwarding vehicle to destinations unknown to the local routing processes. This check identifies network addresses not compatible with the class of the network. For example, a default network of `128.0.0.0` is illegal, since this is a reserved class B network. However, a default network address of `90.0.0.0` is legal, since it is a class A network and thus only requires the first octet.

Bad Network Address Specified in Routing Process

The **network** *network-number* router sub-command restricts the networks from which routing updates are acquired and to which local routing updates are advertised. This check identifies network addresses not compatible with the class of the network. For example, a network address of `192.0.0.0` is illegal, since this is a reserved class C network. However, a network address of `90.0.0.0` is legal, since it is a class A network and thus only requires the first octet. Also identified are host or subnetted addresses, as a major network address is required.

Bad Target in Static Route Definition

The **ip route** *network* **[***mask***] {address|interface}** command requires a target IP address and mask for the static route definition. The mask must be compatible with the class (e.g. A, B, or C) of the target IP address. This check identifies erroneous address/mask combinations. For example, a target address of `198.23.56.0` and a mask of `255.255.0.0` are not compatible since a class B mask is provided for a class C address.

Connected IP (Sub)Net not advertised by RIP/IGRP/EIGRP/OSPF

The **network** *network* router sub-command specifies the networks upon which a routing process advertises and accepts routing updates. If a connected network is not mentioned in the **network** command of a defined routing process (e.g. RIP, IGRP, EIGRP, OSPF), routing updates are not advertised to the interfaces of those networks. This check identifies all connected networks and subnets that are not mentioned by any of the local routing processes.

Dead-End Serial Interface

A topology is created by piecing together the information resident in a specified collection of router configuration files. A topology is a collection of “nodes” and “edges” inter-connecting the nodes. The nodes in the topology are comprised from the observed routers and the serial links and LAN segments denoted by the **interface** *type* command, where *type* is one of `ethernet`, `tokenring`, `fddi`, `serial`, `hssi`, or `bri`. A LAN segment is created for every observed network or subnet, as defined by the masking information in the **ip address interface** sub-command for interfaces of type `ethernet`, `fddi`, or `tokenring`. A serial link is created to connect pairs of `serial`, `hssi`, or `bri` interfaces which *uniquely* map to the same network. Uniquely means only two router interfaces map to the net or subnet denoted by the link. If more than two serial interfaces map to the same subnet, a check is made to determine if two of them have consecutive IP addresses (e.g. `199.35.121.22` and `199.35.121.23`). If they do, a link is created to connect the two serial interfaces. If they do not, a link is *not* created. Thus, after topology generation, dangling serial link connections may exist. They may also exist due to any inherent incompleteness in the specified collection of router configuration files, which may only represent a subset of the routers. Dead-end serial links are flagged to allow the missing pieces to be filled in.

Fast Switching Low Speed Serial Interface

Fast switching is configured by default. Routers generally offer better packet transfer performance when fast switching is enabled, with the following exception. On networks using slow serial links (64K or less), disabling fast-switching's per-destination load-balancing behavior and enabling per-packet load-sharing is usually the best choice. This can be accomplished via the no **ip route-cache** command.

IGRP Metric Mismatch Between Connected Interfaces

IGRP (Interior Gateway Routing Protocol) and EIGRP (Extended Interior Gateway Routing Protocol) factor several interface metrics (e.g. bandwidth and delay) into their computation of routing cost. If a metric value is not explicitly stated, default values are used based upon the interface type (e.g. `serial`, `ethernet`, `tokenring`, etc.) For example, T1 characteristics are assumed for a `serial` interface. The reality may be that the interface is connected up to a 56 Kbps line, meaning traffic may be non-optimally routed across the link. This check examines all interface interconnections (via both serial links and LAN media types) and checks the bandwidth and delay metric specified or assumed on both sides. If the metrics differ, a warning is issued. A common occurrence is that a serial link metric is diligently recorded on one end, but not the other. In the case of a 56 Kbps line, the link looks attractive in one direction and is avoided, if possible, in the other direction (leading to asynchronous paths). This may be intentional, but is not the typical case.

Missed Opportunity for a Passive Interface

IP routing algorithms can selectively advertise on interfaces. If all connected routers on an interface are *not* running one of the algorithms on the current router, that interface should be made passive. This way, the bandwidth of the attached media will not be consumed with routing updates which are not of interest to the attached parties. This check identifies this situation on point to point serial interfaces, where wasted bandwidth is especially critical.

Non-utilized Rule in Access List

The **access-list** *access-list-number* **{deny|permit}** commands specify a collection of rules which can be utilized for route filtering (e.g. distribution lists) and packet filtering (e.g. access-groups on interfaces). The rules are evaluated sequentially looking for a match. If a match is found, the packet or routing update is either denied/permitted according to the rule. The matching criteria for an extended access list are source address and mask, destination address and mask, destination TCP/UDP port, and protocol. A danger when creating long access lists is that preceding rules may subsume (be more general than) subsequent rules. If this happens, the later rules are never utilized. The prior rule shadows them. If `access-list 80 deny 198.0.0.0 0.255.255.255` preceded the rule `access-list 80 permit 198.65.0.0 0.0.255.255`, the second rule would never be exercised. If the access list action differs between the two rules, as in this case, a (declarative) inconsistency exists which is only resolved by virtue of the sequential (i.e. procedural) evaluation of the rules. This is classified as a high severity violation. If, on the other hand, the access list actions agree, the second rule is unnecessary and is identified as a warning. The potential for subsumption exists also in the protocol restriction (`ip` subsumes `tcp`, `udp`, `icmp`, `igrp`) and in the port restriction (`gt 0` subsumes `gt 1024`). This check performs a pair-wise comparison between every rule and subsequent rules in the same access list.

Opportunity for Autonomous Switching

Autonomous switching provides the router much faster packet processing abilities in most cases. This is configured on interfaces via the **ip route-cache cbus** command. It works only in Cisco 7000 series and AGS+ systems. This check identifies router interfaces not configured as such for autonomous switching and which appear to be from the appropriate router series. Currently, this can only be checked by noticing interfaces with port/unit parameters (e.g `Ethernet0/1`), as this feature was introduced with the 7000 series.

OSPF Cost Mismatch Between Connected Interfaces

Like the IGRP metric mismatch problem, an OSPF cost mismatch among connected interfaces can lead to asynchronous paths. If a cost value is not explicitly stated, default values are derived based upon configured bandwidth.

Redistribution: Metric Value Missing Where No Default

Metrics for different protocols cannot, in general, be directly compared. A default metric helps solve the problem of redistributing routes with incompatible metrics. Whenever metrics do not convert, use of a default provides a reasonable substitute and enables the redistribution to proceed. IGRP/EIGRP can automatically redistribute IGRP/EIGRP/static, but they do not have a metric conversion for RIP. RIP needs a metric conversion to redistribute IGRP, EIGRP, or OSPF. OSPF always produces a metric conversion when it redistributes, but it is best to provide an explicit metric.

Primary and Secondary Addresses Map to Same Subnet

The **network** *network-number* router sub-command specifies the networks upon which a routing process advertises and accepts routing updates. A little known restriction is that major network routing updates are only broadcast from the primary address of each interface. Thus routers inter-connected via a secondary address (i.e. secondary net or subnet) only receive updates about their subnet and not about the other major networks connected to the other router. Implicit route filtering occurs on interfaces through the secondary address. A good policy is to interconnect routers via primary addresses and to only use secondary addresses to make a network contiguous. Thus, primary addresses should map to (sub)nets accessible via a primary address on other routers. Secondary addresses should map to (sub)nets accessible via a secondary address on other routers. In general, problems occur when primary addresses map to (sub)nets only accessible via secondary addresses on other routers. This check identifies all router interconnections via (sub)nets where the first router is logically connected via a primary address and the second router is logically connected via a secondary address.

Static Route Next Hop is an Unresolved Address

The **ip route** *network* **[***mask***] {address|interface}** command has an option to specify the next hop address of a connected router. This check identifies static route definitions that forward to an unknown router interface address. Either an erroneous forwarding address was specified or the problem stems from an incompleteness in the baseline router configuration files. The latter case could occur if only a subset of the router configuration files in an internetwork were included in the baseline.

Unconnected Net in network Command of RIP/IGRP/EIGRP

The **network** *network* router sub-command specifies the *connected* networks upon which a routing process advertises and accepts routing updates. If an unconnected network is mentioned in the **network** command of a defined routing process (e.g. RIP, IGRP, EIGRP), the command is essentially ignored. It is not uncommon to see this mistakenly used in an (ineffective) attempt to filter routes from indirectly connected networks. Therefore, check your assumptions.

Unused distance Command

The **distance** *weight ip-source-address ip-mask* commands are sequentially evaluated to assign an administrative distance to the routes received from another router. The first matching distance determines the weight assigned to a learned route. If a preceding command is more general than a subsequent command, the latter command will never be exercised. This may not be what you want.

Unused eigrp summary-address Command

The **ip summary-address eigrp** *autonomous-system* commands on an interface are sequentially evaluated to determine summarization of advertised EIGRP routes. The first rule which matches with respect to an autonomous system is used. If a preceding command is more general than another command, the subsequent command will never be exercised. This may not be what you want.

Unused OSPF area range Command

The **area range** OSPF commands are sequentially evaluated to determine how to summarize between areas. The first rule which matches with respect to an area is the one used. If a preceding command is more general than another command, the subsequent command will never be exercised. This may not be what you want.

Unused OSPF network area Command

The **network area** OSPF commands are sequentially evaluated to determine which area an interface is to be assigned. The first rule which matches determines the assigned area. If a preceding command is more general than a subsequent command, the latter command will never be exercised. This may not be what you want.

IPX Integrity Checks

This section describes the IPX integrity checks performed by the Connectivity Tools. The checks are organized into High Priority and Warning-Level groupings and are listed alphabetically within each of the Report groupings.

IPX High Priority Reports

The following IPX integrity checks are performed by the Connectivity Tools. If a potential problem is uncovered while performing these checks, an IPX High Priority problem report is generated and placed in the Diagnostic Report.

Duplicate Address Check

All IPX network/host address pairs assigned to routers are checked for duplication. Each router running IPX, is assigned a host address via the Novell **ipx routing** *node* command. Interfaces on these routers connect to IPX networks via the Novell **ipx network** *number* command. This check verifies that the IPX network/host-address pairs are unique across all routers.

IP and IPX Logical Topologies are Out of Synch

If two router interfaces are logically connected via IP but do not share any IPX networks in common, the IPX and IP topologies are inconsistent. Check your addresses. The IP and IPX views should overlay consistently upon one another.

Non-utilized Rule in Access List

The **access-list** *access-list-number* **{deny|permit}** commands specify a collection of rules which can be utilized for IPX routing table, SAP, and generic output filtering. The rules are evaluated sequentially looking for the *first* match. If a match is found, the packet's SAP/routing update is either denied/permitted according to the rule. The use of wild-cards (e.g. -1 network values, 0 socket/protocol values, and masking) enables the creation of very general rules. This check locates all preceding rules in access lists which shadow (or are more general than) subsequent rules. If the actions differ (e.g. permit vs. deny) between the two, the intent for the latter rule is not met (since the rule is never exercised.) This is classified as a high severity problem. If the action flag is the same, a simple redundancy situation exists. This is classified as a warning-level problem.

Undefined Access List Referenced

The Novell **ipx access-group**, **ipx input-network-filter**, **ipx output-network-filter**, **ipx input-sap-filter**, and **ipx output-sap-filter** commands require a reference to an access list. This check identifies references to undefined access lists. An empty access list has the implicit behavior of denying all access.

IPX Warning-Level Reports

The following IPX integrity check is performed by the Connectivity Tools. If a potential problem is uncovered while performing this check, an IPX Warning-Level problem report is generated and placed in the Diagnostic Report.

Non-utilized Rule in Access List

The **access-list** *access-list-number* **{deny|permit}** commands specify a collection of rules which can be utilized for IPX Routing Table, SAP, and generic output filtering. The rules are evaluated sequentially looking for the *first* match. If a match is found, the packet, SAP update, or routing update is either denied/permitted according to the rule. The use of wild-cards (e.g. `-1` network values, `0` socket/protocol values, and masking) enables the creation of very general rules. This check locates all preceding rules in access lists which shadow (or are more general than) subsequent rules. If the actions differ (e.g. permit vs. deny) between the two, the intent for the latter rule is not met (since the rule is never exercised.) This is classified as a high severity problem. If the action flag is the same, a simple redundancy situation exists. This is classified as a warning-level problem.

Remote Source Route Bridging Integrity Checks

This section describes the Remote Source Route Bridging (SRB) integrity checks performed by the Connectivity Tools. The checks are organized into High Priority and Warning-Level groupings and are listed alphabetically within each of the Report groupings.

SRB High Priority Reports

The following SRB integrity checks are performed by the Connectivity Tools. If a potential problem is uncovered while performing these checks, an SRB High Priority problem report is generated and placed in the Diagnostic Report.

Local SRB Peer Definition Problem

Each router in a remote source-bridged ring group must specify a local interface address as its stop on the virtual ring. For Fast Sequenced Transport (FST) encapsulation, the **source-bridge fst-peername** *local-interface-address* command must be used to define the local address. For Transport Control Protocol (TCP) encapsulation, a **source-bridge remote-peer** *ring-group* **tcp** *ip-address* command must exist to define the local address.

Multiple SRB Remote Peer References to a Router

The **source-bridge remote-peer** *ring-group* **{fst|tcp}** *ip-address* [version *n*] command requires a reference to a router interface address which denotes a stop on the virtual ring. Only one remote peer relationship, per ring group, per router, is allowed.

Referenced Remote Peer is not the Local Peer

The **source-bridge remote-peer** *ring-group* **{fst|tcp}** *ip-address* [version *n*] command requires a reference to a router interface address which denotes a stop on the virtual ring. If the specified address resolves to a known router interface address, but that address is *not* the local peer on that router, a problem exists. Note that this became a requirement for IOS 10.x releases, but is not a problem for IOS 9.x releases.

SRB Remote Peers Encapsulation Mismatch

The **source-bridge remote-peer** *ring-group* **{fst|tcp}** *ip-address* [version *n*] command requires an encapsulation type (`fst` or `tcp`) and optionally a version tag. If the specified address resolves to a known router interface address, but the remote peer does not agree on the encapsulation attribute, this constitutes a potential high severity problem.

Unbalanced SRB Remote Peers

The **source-bridge remote-peer** *ring-group* **{fst|tcp}** *ip-address* [version *n*] command requires a reference to a router interface address which denotes a stop on the virtual ring. If the specified address resolves to a known router interface address, but the remote router does not have a balancing **source-bridge remote-peer** command pointing back to the first router's peer address, an unbalanced remote peer relationship exists.

SRB Warning-Level Reports

The following SRB integrity check is performed by the Connectivity Tools. If a potential problem is uncovered while performing this check, an SRB Warning-Level problem report is generated and placed in the Diagnostic Report.

Unresolved SRB Remote Peer Address Referenced

The **source-bridge remote-peer** *ring-group* **{fst|tcp}** *ip-address* [version *n*] command requires a reference to a router interface address which denotes a stop on the virtual ring. If the specified address does *not* resolve to a known router interface address, this warning is issued.

SNA STUN Integrity Checks

This section describes the SNA STUN integrity checks performed by the Connectivity Tools. The checks are organized into High Priority and Warning-Level groupings and are listed alphabetically within each Report group.

SNA STUN High Priority Reports

The following SNA STUN integrity checks are performed by the Connectivity Tools. If a potential problem is uncovered while performing these checks, an SNA STUN High Priority problem report is generated and placed in the Diagnostic Report.

STUN Route Does Not Reference STUN Peername

If a STUN connection uses TCP/IP encapsulation, the IP addresses of the **stun route address** and **stun route all** interface configuration commands should match the IP addresses of the complementary **stun peer-name** global configuration commands.

STUN Needs Local Peername

The **stun peer-name** command should reference one of the IP addresses on the router.

SNA STUN Warning-Level Reports

The following SNA STUN integrity check is performed by the Connectivity Tools. If a potential problem is uncovered while performing this check, an SNA STUN Warning-Level problem report is generated and placed in the Diagnostic Report.

STUN Route References Unknown Address

The peer IP address mentioned in a **stun route address** or **stun route all** interface configuration command does not match any known IP address.

AppleTalk Integrity Checks

This section describes the AppleTalk integrity checks performed by the Connectivity Tools. The checks are listed alphabetically within the Report group.

AppleTalk High Priority Reports

The following AppleTalk integrity checks are performed by the Connectivity Tools. If a potential problem is uncovered while performing these checks, an AppleTalk High Priority problem report is generated and placed in the Diagnostic Report.

IP and AppleTalk Logical Topologies Out of Synch

If two router interfaces are logically connected via IP but do not have common AppleTalk cable-ranges, the AppleTalk and IP topologies are inconsistent. Check your addresses. The IP and AppleTalk views should overlay consistently upon one another.

Overlapping AppleTalk Cable Ranges

A cable range denotes a range of AppleTalk nets connected to an interface. The cable ranges assigned to two different interfaces should match exactly and not overlap.