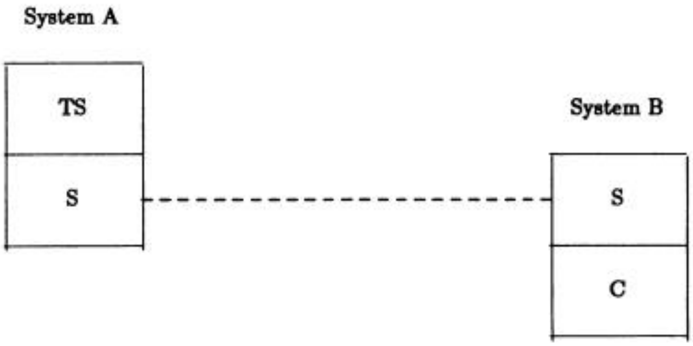
C.3.2. The Cascading Problem

One of the problems that the interconnection rule does not address is the *cascading problem*. The cascading problem exists when a penetrator can take advantage of network connections to compromise information across a range of security levels that is greater than the accreditation range of any of the component systems he must defeat to do so. Cascading is possible in any connected network that processes a greater range of security levels than any one of its component systems is accredited to handle, and it is possible in others as well.

As an example of the cascading problem, consider two systems, each of which is accredited to handle two adjacent classifications of information, as shown in Figure C2. System A processes SECRET and TOP SECRET information, and all users are cleared to at least the SECRET level. System B processes CONFIDENTIAL and SECRET information, and all users are cleared to at least the CONFIDENTIAL level.

While the risk of compromise in each of these systems is small enough to justify their use with two levels of information, the system as a whole has three levels of information, increasing the potential harm that could be caused by a compromise. When they are connected so that SECRET information can pass from one to the other, a penetrator able to defeat the protection mechanisms in these systems can make TOP SECRET information available at the CONFIDENTIAL level.

Figure C2. Cascade Problem, Illustration 1



Consider this chain of events: a penetrator (1) overcomes the protection mechanism in System A to downgrade some TOP SECRET information to SECRET; (2) causes this information to be sent over the network to System B; and (3) overcomes the protection mechanism in System B to downgrade that same information to CONFIDENTIAL. This is the cascading problem.

C.3.2.1. Problem Identification

There are various approaches, with different degrees of complexity and precision, for recognizing a potential cascading problem. Two of these approaches will be addressed in this Appendix. The first is a fairly simple test that can ensure that a network does *not* have a cascading problem: the *nesting condition*. The second, discussed in Section C.4, is a less conservative but much more complex heuristic that takes into account the connectivity of the network and the evaluation classes of the component AIS.

The nesting condition is satisfied if the accreditation ranges of every two AISs are either disjoint (have no level in common) or nested, i.e., one is included within the other. In most cases, the nesting condition is enough to determine whether there is a cascading problem. However, this is a somewhat conservative test; there are cases where the nesting condition fails, but there is actually no cascading problem.

Example 1: Consider the situation illustrated in Figure C1. The accreditation range of Component A is nested within that of Component B (i.e., C-S is completely contained within C-TS). Therefore, the nesting condition is satisfied, and there is no cascading problem.

Example 2: Consider the situation illustrated in Figure C2. The accreditation ranges of System A and System B are not disjoint; neither is one completely contained within the other. Therefore, the nesting condition fails, and a cascading condition is indicated.

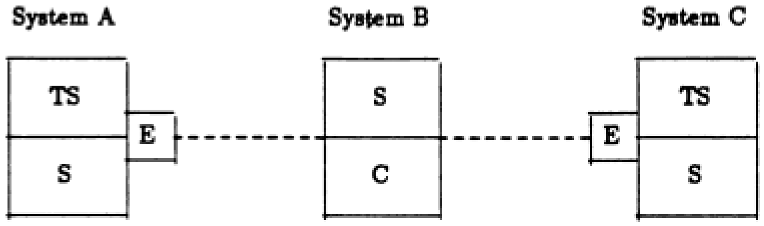
Example 3: Consider the situation illustrated in Figure C3. Again, the nesting condition does not hold, because the accreditation range of System B is neither disjoint from nor contained in that of Systems A and C. A cascading condition is thus indicated. However, it will be shown below in this Appendix that Figure C3 actually does not contain a cascading condition, due to the presence of the end-to-end encryption devices.

C.3.2.2. Solutions

When a cascading problem is to be addressed, there are several ways to do so. One solution is to use a more trusted system at appropriate nodes in the network, so that a penetrator will be forced to overcome a protection mechanism commensurate with the seriousness of the potential compromise. In the example depicted in Figure C2, If either system A or system B is evaluated at class B3, which is sufficient according to the Environmental Guidelines for a range of TOP SECRET to CONFIDENTIAL, then the penetrator is presented with an acceptable level of difficulty.

Another possible solution is to eliminate certain network connections, either physically or by means of end-to-end encryption. End-to-end encryption allows hosts that need to communicate to do so, while eliminating additional unnecessary cascading risk on the path from one to the other.

Figure C3. Cascade Problem, Illustration 2 (End-to-End Encryption)



In Figure C3, suppose that System A needs only to communicate with System C, and B is just an intermediate node. The possible cascade from TOP SECRET in A to CONFIDENTIAL in B can be avoided by applying end-to-end encryption from A to C, since encrypted data from A can be released at the CONFIDENTIAL level in B without compromise.

Note that end-to-end encryption is of no help in the Figure C2 example, since the systems participating in the cascade were required to communicate.

In some situations where the potential for a cascading problem exists, the risk of its occurrence is actually not significant. A penetration making use of network connections, as described above, generally requires coordination between attacks on two connected systems. It may be possible to determine, in individual cases, that opportunities for this kind of coordination, in the form of common software or common users, are unlikely. One might then disregard cascading over the connections in question.

On a more global scale, one might divide the network into communities, with respect to the possibility of cascading. If connections between one community and another were believed not to support a cascade threat, then a cascading analysis would be performed only within each community.

C.3.2.3. Networks of Evaluated Systems

If the systems to be interconnected can be assigned evaluation classes, the ratings of these systems can be used as input to analysis procedures for detecting the cascade problem and testing proposed solutions. The first step in developing analysis procedures is to define formally the conditions necessary for the absence or presence of a cascade problem.

An assertion called the *Cascade Condition* will be defined below. A network satisfying the Cascade Condition does not have a cascade problem. This condition is stated in terms of the evaluation ratings of the interconnected systems and the direction and sensitivity level of connections between them.

Some definitions are needed in order to state the cascade condition formally. The terminology given below is meant to be used only in the context of this section.

A *protection region* is a pair (h,s) such that h is a network component and s is a sensitivity level processed by component h.

A *step* is an ordered pair of protection regions (h_1,s_1), (h_2,s_2) such that either

1. $s_1 = s_2$ and h_1 sends to h_2 at level s_1 (a network link), or
2. $h_1 = h_2$ (an information flow within a component).

A *path* is a sequence of protection regions such that each consecutive pair of regions is a step.

A path is a sequence of protection regions that may be traversed, step by step, by data. A step along a network link is possible either when there is a direct communications link from one component to the other carrying information at a given level, or when there is an indirect, end-to-end encrypted connection in which intermediate components are unable to read the data. A step between two regions in the same component may be (but does not have to be) a covert channel.

Given a host h, let $L(h)$ be the minimum clearance of users of h. Given a sensitivity level s, one can use the Environmental Guidelines to determine the minimum evaluation class $C(s,h)$ required for a system with the associated risk index. The requirement for open environments should be used unless all systems on the path are closed. Note that $C(s,h)$ will be at least B1 if the risk index associated with s and $L(h)$ is greater than zero.

With these definitions, we can now state the *Cascade Condition*:

For any path $(h_1,s_1),\ldots,(h_n,s_n)$ such that $s_n = L(h_n)$ and $C(s_1,h_n)$ is at least B1, there must exist at least one step $(h_i,s_i),(h_{i+1},s_{i+1})$ such that $h_i = h_{i+1}$, the evaluation class of h_i is at least $C(s_1,h_n)$, and s_i is not dominated by s_{i+1}.

This condition can be paraphrased by saying that every path that might compromise data of level s_1 by making it available to an insufficiently cleared user of h_n must overcome the protection mechanism in a component of class at least $C(s_1,h_n)$.

C.4. EXAMPLE: An Heuristic Procedure for Determining if an Interconnection Should Be Allowed

There should be some way of determining whether a system has a risk index that is too great for its evaluation rating (and the evaluation ratings of its components). Given the goal of not allowing a greater risk than is recommended by the Environmental Guidelines, the following is an heuristic algorithm that has been developed to examine systems and determine if they fall within the bounds prescribed by the Environmental Guidelines. (In formal terms, this algorithm is an approximate test for the Cascade Condition, described above.) It should be noted that this algorithm is not intended to be prescriptive: it is merely one way of examining the problem. There are doubtless many other ways that are just as valid.

Furthermore, as any heuristic algorithm, this cannot be derived from first principles. It has been derived largely through trial and error; it produces reasonable results (e.g., it disallows systems when it seems prudent; it recommends levels of security that are consistent with the Environmental Guidelines).

This algorithm should not be taken to be anything more than intended; it does not magically solve all network security problems. It does, however, provide useful guidance and recommendations as to the prudence of interconnecting various systems.

The following describes an algorithm for determining whether or not a given network, composed of evaluated components, meets the risk categories of the Environmental Guidelines. The algorithm is based on the idea of dividing up a network into groups (where a group is defined to be a group of components that can potentially exchange information i.e., send and receive data at a common sensitivity level, and have an evaluation Class at or below a given level).

The risk presented by any given group can be compared to the maximum allowed risk, as defined by the Yellow Book for a system at the given evaluation class, to determine if any community presents an unacceptable risk.

1. Create a Network Table listing all components within the network. This table, illustrated in Table C1, should include for each component the following information: Component ID, Evaluation Class, Range of Security Classifications at which the component sends data to the network, List of Security Classifications at which the component receives data from the network, Maximum of (highest level of data received from network and highest level of data processed by component), Minimum of (clearance of the user with the lowest clearance of the users with direct access to the component and lowest level of data sent to the network from the component).

Table C1. Example Entry:

Component ID	Eval.	Send	Receive	Maximum	Minimum
Node A	B2	TS S	TS S	TS	S
Node B	A1	S C	S C	TS	FOUO

2. Produce a Network Table Evaluation Class, a Network Table Maximum and a Network Table Minimum. The Network Table Evaluation Class will be the highest evaluation class of any component listed in the table. (In Table C1, this is A1.) The Network Table Maximum will be the maximum of the Maximums associated with all the components listed in the table which send data to the network. (This is determined by taking the highest entry in the "Maximum" column; in Table C1, it is TS.) The Network Table Minimum will be the minimum of the Minimums associated with all the components listed in the table which receive data from the network. (This is determined by taking the lowest entry in the "Minimum" column; in Table C1, this is FOUO.)

3. If the Network Table Evaluation Class is greater than B1, (i.e, A1, B3, or B2) then tables for each evaluation class lower than the Class of the Network Table, must be produced, down to table(s) for the C1 class. These tables will be produced for each evaluation class by first listing any one component whose evaluation class is less than or equal to the evaluation class for the table. Then, add to the table all components that meet all of the following conditions:

a) They have an evaluation class less than or equal to the class of the table.

b) They receive data from the network at a level that is being sent by a component who is already in the table.

c) They send data to the network at a level that is equal to or less than any node already in the table.

4. After all the tables have been constructed then the Network Table Evaluation Class of each table is compared to the Maximum and Minimum for the Table with regard to the rules specified by the Environmental Guidelines.

5. If all Tables satisfy the assurance requirements for the Environmental Guidelines then the Network passes the assurance requirements. If any of the Tables provide a greater risk index than is permitted by the Environmental Guidelines then the Network provides a high level of risk, and should not be connected as currently designed.

Table C2. B2 TABLE 1

ID	EVAL.	SND	RCV	MAX	MIN
A	B2	S	S	S	S

Table C3. B2 TABLE 1, EXTENDED

ID	EVAL	SND	RCV	MAX	MIN
A	B2	S	S	S	S
B	B2	C-S	C-S	S	C

Table C4:

Table C4(a). B2 TABLE 1

ID	EVAL.	SND	RCV	MAX	MIN
A	B2	S	S	S	S
B	B2	C-S	C-S	S	C

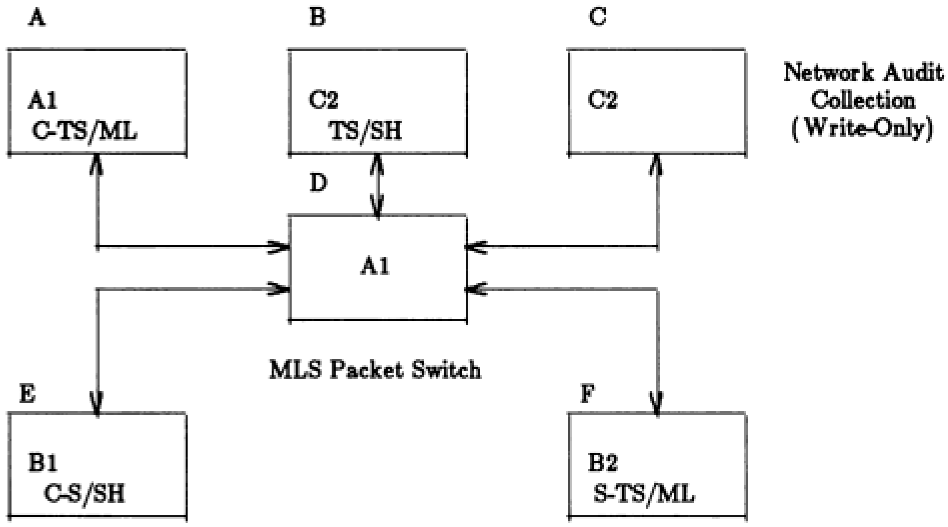
Table C4(b). B2 TABLE 2

ID	EVAL.	SND	RCV	MAX	MIN
C	B2	TS	S,TS	TS	S

C.4.1. Example B2 Table

As an example consider Table C2. This represents a B2 table under construction with a single entry. If in the network there existed another node which was evaluated at B2 and could receive and send at C-S, this node would be added to the table, producing Table C3. In contrast if there existed in the network a B2 node that could receive at S-TS but could only send at TS, this node would not be added to Table C3 but could be used to start a second B2 table. There would then be a set of two tables, represented in Table C4.

Figure C4. A Sample Network



Component ID	Permitted Operations
A	Send and Receive data from C through TS
B	Send and Receive TS-only
C	Can Receive only audit records, all of which are treated as TS
D	Can Send and Receive C through TS
E	Can Send and Receive S-only
F	Send and Receive S and TS data

C.4.2. Sample Network and Tables

A sample network is illustrated in Figure C4. The tables that are produced for it are given in Tables C5(a) through C5(h).

Table C5(a). NETWORK TABLE

NETWORK TABLE EVAL CLASS = A1
NETWORK TABLE MAXIMUM = TS
NETWORK TABLE MINIMUM = C
ENVIRONMENTAL GUIDELINES RULING = OK

ID	EVAL CLASS	SND	RCV	MAX	MIN
A	A1	C-TS	C-TS	TS	C
B	C2	TS	S-TS	TS	TS
C	C2	-	C-TS	TS	TS
D	A1	C-TS	C-TS	TS	C
E	B1	S	S	S	S
F	B2	S-TS	S-TS	TS	S

(Note: since there are no B3 components, the B3 tables are identical to the B2 tables and are therefore not reproduced here.)

Notice at the B2 level the network is represented by two tables, C5(b) and C5(c). This is due to the fact that one of the components (Component C) is a receive-only component. Such components will always end up in a table by themselves due to the fact that they never can affect the security of other nodes on the network. (The only security consideration to make with such components is whether they are receiving data at a level dominated by the approved maximum processing level for the component.)

Notice at the B1 level the components are each in a table by themselves (Tables C5(d), C5(e), and C5(f)). This is due to the fact that although Component B may receive data from Component E, it never sends data to the network at a level lower than that sent by E (i.e., B can only send TS, never Confidential or Unclassified). Thus there is no "added" risk in having B receive data from E. If it were the case the B could send data at a level lower than (or equal to) E then they would be included in the same table since they present an added (or equal) risk.