Naval Research Laboratory
Washington, DC 20375-5320

NRL/FR/5542--92-9528

(2)

The ECA Critical Requirements Model

CHARLES N. PAYNE, JR.
DAVID M. MIHELCIC
ANDREW P. MOORE

Center for Secure Information Technology
Information Technology Division

KENNETH J. HAYMAN

Trusted Computer Systems Group
Electronics Research Laboratory
Defence Science & Technology Organisation
Salisbury, South Australia

December 28, 1992

12.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (*Leave Blank*)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	December 28, 1992	Final

4. TITLE AND SUBTITLE

The ECA Critical Requirements Model

5. FUNDING NUMBERS

PE - CSP 03051670
WU - 555328400

6. AUTHOR(S)

Charles N. Payne, Jr., David M. Mihelcic, Andrew P. Moore,
and Kenneth J. Hayman*

7. PERFORMING ORGANIZATION NAME(S) and ADDRESS(ES)

Naval Research Laboratory
Washington, DC 20375-5320

8. PERFORMING ORGANIZATION REPORT NUMBER

NRL/FR/5542--92-9528

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Space and Naval Warfare Systems Command
Arlington, VA

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

*Trusted Computer Systems Group, Electronics Research Laboratory
Defence Science & Technology Organisation, Salisbury, South Australia

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (*Maximum 200 words*)

The ECA is an embedded computing device that processes message traffic for a network that must enforce end-to-end user message confidentiality. The ECA uses a commercial, off-the-shelf cryptographic device to transform sensitive data from the Red Domain of the network so that it can be transmitted over the untrusted communication links of the Black Domain. For transmission purposes, certain parts of a message, namely the message header, must be bypassed around the cryptographic device. The primary critical requirement for the ECA, "Restricted Red-to-Black Flow" (RRTBF), requires that the bypassed portion of each message must satisfy certain format restrictions, and that the rate of bypass must be constrained. In this report, we present an informal model of the ECA's critical requirements together with the assumptions under which the model was constructed. We than formalize this model by using the CSP Trace Model of computation.

14. SUBJECT TERMS

Formal methods
Formal models
Security

15. NUMBER OF PAGES

13

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std 239-18
298-102

CONTENTS

THE ECA CRITICAL REQUIREMENTS MODEL

INTRODUCTION

The External Communications Security (COMSEC) Adapter (ECA) is an embedded computing device that processes message traffic for a network. Its functional requirements [1] are summarized briefly below in Overview. We assume that the network in which the ECA resides enforces a simple security policy of message data confidentiality. From this policy, we can derive critical requirements for the ECA. A *critical requirement* is a constraint on a system that, if not satisfied, may result in the system engaging in catastrophic behavior. This report[1] presents a formal model of the ECA's critical requirements. First, we develop an informal model of the requirements. Then we formalize that model by using the Trace Model of CSP developed by Hoare [2, 3]. Our exposition of the ECA formal model is patterned after the Secure Military Message System model [4].

In the Overview below, we identify the ECA's critical requirements. Next we present the informal model and then the formal model. The Glossary contains definitions of ECA–specific terms; henceforth, these terms appear in SMALL CAPITAL LETTERS.

OVERVIEW

The ECA partitions the network in which it resides into a RED DOMAIN for processing sensitive information and a BLACK DOMAIN for processing nonsensitive information. Information is nonsensitive if its classification level does not exceed the classification level to which the BLACK DOMAIN is trusted; it is sensitive otherwise. The ECA has four external interfaces: a RED INTERFACE for communicating MESSAGEs with the RED DOMAIN, a BLACK INTERFACE for communicating MESSAGEs with the BLACK DOMAIN, a CRYPTOGRAPHIC INTERFACE for loading a KEY, and a TIME INTERFACE for accepting TIME signals. The CRYPTOGRAPHIC INTERFACE and the TIME INTERFACE reside in the RED DOMAIN.

The ECA must satisfy two important functional requirements: the ECA shall use a cryptographic function, and it shall use a bypass around that function. Encryption makes the sensitive portion of a MESSAGE nonsensitive so that the MESSAGE can be transmitted over an untrusted medium. A MESSAGE can be partitioned into a CRYPTO DATA portion, which contains sensitive text supplied by the user, and a BYPASS DATA portion, which contains transmission protocol information. The CRYPTO DATA must be encrypted in the BLACK DOMAIN. The BYPASS DATA cannot be encrypted there, because the network routing function resides in the BLACK DOMAIN. The ECA must divert the BYPASS DATA around the cryptographic function.

Figure 1 illustrates a typical scenario for using the ECA in a network. A MESSAGE is transmitted from some DEVICE A to another DEVICE B. An ECA that is local to A splits the MESSAGE into BYPASS DATA and CRYPTO DATA, encrypts the CRYPTO DATA by using the cryptographic function E_K, bypasses the corresponding BYPASS DATA, and transmits the encrypted MESSAGE over the network to a remote ECA (that is local to B). The remote ECA decrypts the CRYPTO DATA with the cryptographic function D_K. The dashed box in Fig. 1 represents the division of the RED DOMAIN and the BLACK DOMAIN. Everything outside of the dashed box resides in the RED DOMAIN.

CRITICAL REQUIREMENTS

The network must enforce the confidentiality of information: users shall not obtain information for which they are not authorized. This is partially achieved by the distribution of the cryptographic KEY. A user's local ECA receives a KEY that is appropriate for decrypting the information that the user is authorized to obtain.

Manuscript approved June 25, 1992.

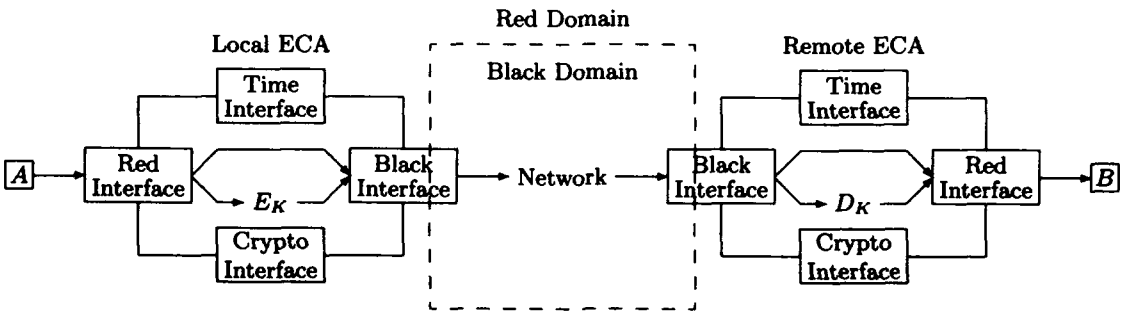

Figure 1: The ECA in a network

However, KEY distribution alone is not sufficient to enforce data confidentiality. The ECA must ensure that sensitive information does not enter the unprotected BLACK DOMAIN. Since the ECA communicates with DEVICEs in the BLACK DOMAIN over its BLACK INTERFACE, we have the following critical requirement for the ECA:

> **Restricted Red–To–Black Flow.** Sensitive information shall not be transmitted over the BLACK INTERFACE to the BLACK DOMAIN.

The obvious response to satisfying **Restricted Red–To–Black Flow** is to encrypt the CRYPTO DATA of every MESSAGE before the MESSAGE is transmitted over the BLACK INTERFACE. However, sensitive information can also reside in the BYPASS DATA; a DEVICE could inadvertently or maliciously encode sensitive information in the BYPASS DATA. For example, a MESSAGE can be ill-formed so that its "BYPASS DATA" actually includes some CRYPTO DATA. The ECA must determine what belongs in the BYPASS DATA of a MESSAGE and what does not. This is accomplished through FORMAT CHECKs on the BYPASS DATA. However, although the FORMAT CHECKs are thorough, they do not ensure perfect confidentiality. The ECA must also constrain the BYPASS RATE of the BYPASS DATA so that, even if sensitive information is released, the bandwidth is small. Finally, the ECA must ensure that all MESSAGEs (that were not generated internally) are transmitted over the BLACK INTERFACE in the order they were received, and each MESSAGE must be transmitted only once. This last constraint is designed to restrict covert signaling initiated within the ECA, e.g., by a Trojan horse; it is not intended to address signaling initiated by the ECA's environment, i.e., external to the ECA. In general, we are concerned primarily with the obvious covert storage channels.

The issues discussed above suggest some derived critical requirements for the ECA. These requirements, together with important assumptions about the ECA's operating environment, are identified in the informal model described below.

INFORMAL MODEL

The functional requirements described in Ref. 1 do not need to be modeled to argue that an implementation enforces **Restricted Red–To–Black Flow**. Instead, we model only those critical requirements that, if not enforced, could compromise **Restricted Red–To–Black Flow**. This section presents an informal model of those critical requirements.

While the functional requirements discuss the operating states of the ECA, the informal model ignores them: the ECA must preserve **Restricted Red–To–Black Flow** regardless of its operating state. The informal model constrains the ECA's behavior only when the ECA is attached to a network; otherwise, **Restricted Red–To–Black Flow** has little meaning. For example, a system administrator loads the cryptographic KEY, the FORMAT CHECK parameters, and the BYPASS RATE parameters during the ECA's system configuration, but this action occurs while the ECA is disconnected from the network.

Each of the critical requirements identified in Informal Assertions suggests a "mechanical check" of a MESSAGE before it is transmitted over the BLACK INTERFACE. All MESSAGEs must satisfy the intent of these requirements, but because of operational constraints, not all MESSAGEs will undergo the mechanical check. The ECA may exempt certain MESSAGEs from these checks with the understanding that these MESSAGEs would otherwise satisfy the constraint.

User's View of Operation

The ECA is an embedded system. It has no human users, so a "user's" view of its operation must be interpreted for the DEVICEs to which it connects.

A DEVICE communicates with the ECA over one interface only. A DEVICE in the RED DOMAIN engaged in transmitting and receiving MESSAGEs communicates over the RED INTERFACE; similarly for a DEVICE in the BLACK DOMAIN. The DEVICE communicates with the ECA by using an established protocol. Progress of a transmitted MESSAGE can be relayed to the originating DEVICE if the notification does not violate the critical requirements.

The TIME INTERFACE and the CRYPTO INTERFACE affect communications over the RED INTERFACE and the BLACK INTERFACE but are not accessible to the latter interfaces. The CRYPTO INTERFACE is accessed only during system configuration, when the ECA is disconnected from the network. We rely on administrative procedures to ensure the KEY is not loaded while the ECA is connected. No facilities to load a KEY remotely are provided.

Assumptions

To enforce **Restricted Red–To–Black Flow**, the environment in which the ECA operates must obey certain restrictions. Because the ECA cannot control its environment, these restrictions represent assumptions on the proper operation of the ECA that must be validated before the ECA is used.

1. **Physically Secure** – the ECA operates in a physical environment appropriate for the data it processes, i.e., it is physically secure.
2. **Valid Formats** – the FORMAT CHECK parameters are installed properly (while the ECA is disconnected from the network) and are appropriate for the MESSAGE SET and the network's data confidentiality policy.
3. **Valid Bypass Rates** – the BYPASS RATE parameters are installed properly (while the ECA is disconnected from the network) and are appropriate for the MESSAGE SET, the central processing unit (CPU) used by the ECA, and the network's data confidentiality policy.
4. **Valid Crypto Algorithm** – the ECA is loaded with a cryptographic algorithm and protocol (while the ECA is disconnected from the network) that is appropriate for the MESSAGE SET being processed and the network's data confidentiality policy.
5. **Authentication** – DEVICEs gain access to the services provided by the ECA only after being authenticated.
6. **Key Distribution** – the KEY distribution procedures for the network are appropriate for the network's data confidentiality policy.
7. **Fixed Key** – the KEY that is used to encrypt MESSAGEs does not change while the ECA is connected to the network.[2]

[2]We include this assumption because it simplifies our formal exposition of the critical requirements.

8. **Valid Clock** – the CLOCK used by the ECA communicates TIME to the ECA in a monotonically increasing, linear fashion.

9. **Valid Exemptions** – a MESSAGE that is exempt from one or more of the requirements in Informal Assertions satisfies the intent of the requirement(s) from which it is exempt.

Informal Assertions

The following critical requirements shall be enforced by the ECA. MESSAGEs that are exempt from one or more of these requirements shall be identified prior to the installation of the ECA at a site.

1. **Correct Encryption** – the CRYPTO DATA, if any, of every MESSAGE transmitted over the BLACK INTERFACE shall be encrypted before transmission.

2. **Correct Format** – a MESSAGE shall be transmitted over the BLACK INTERFACE only if the MESSAGE satisfies the FORMAT CHECK restriction: the value of each FIELD of the BYPASS DATA must be within a predetermined range; the length of each FIELD must match a predetermined length for that FIELD; and the overall length of the BYPASS DATA, as specified by a FIELD within the BYPASS DATA, must equal the sum of the lengths of the FIELDs of the BYPASS DATA.

3. **Correct Bypass Rate** – the ACTUAL BYPASS RATE for BYPASS DATA around the cryptographic function, from the RED DOMAIN to the BLACK DOMAIN, shall not exceed the ALLOWED BYPASS RATE. The ACTUAL BYPASS RATE is the amount of BYPASS DATA actually diverted divided by the TIME elapsed. The ALLOWED BYPASS RATE is the amount of BYPASS DATA that *could have been* diverted divided by the TIME elapsed. The ALLOWED BYPASS RATE is bounded from above by a prespecified constant rate.

4. **Correct Order** – every MESSAGE that is not generated internally and that is transmitted over the BLACK INTERFACE shall be transmitted in the same order in which it was received by the RED INTERFACE, and it shall be transmitted only once.

FORMAL MODEL

In this section, we offer a formal statement of the structure and assertions of the informal model. The assumptions of the informal model are still valid, but they are not repeated here. The CSP Trace Model from Refs. 2 and 3 is the computational paradigm for the formal model. It permits the specification of correct behavior in terms of a system's external inputs and outputs. More importantly, this paradigm is the foundation for our proposed decomposition method [5].

The critical requirements from Informal Assertions are formalized in terms of the ECA's six external communication CHANNELs. Our previous illustration of the ECA (Fig. 1) is refined in Fig. 2 to include the CHANNEL set introduced below. In general, if a MESSAGE enters the ECA over RI and satisfies all of the restrictions defined in Informal Assertions, it will exit over BO. Similarly, if a MESSAGE enters the ECA over BI, it will exit over RO. The TIME is input over TI. The KEY enters the ECA over CI only while the ECA is disconnected from the network.

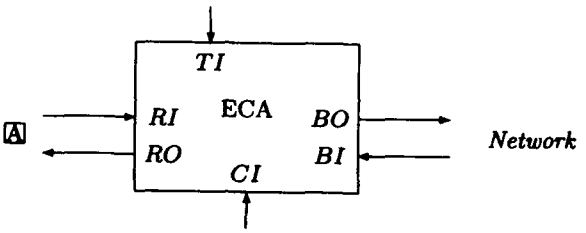

Figure 2: ECA refined view

Definitions

Sequences and traces are fundamental to the model. A sequence $S = \langle a_1, a_2, a_3, \ldots, a_n \rangle$ is an ordered list that is defined under reflexivity, antisymmetry, and transitivity over a precedence operator $\prec$ such that $a_1 \prec a_2 \prec a_3 \ldots \prec a_n$. The sequence is composed of elements, e.g., a_1, from some set A. The length of S is denoted $\#S$. The ith element of S is accessed by $S[i]$.[3] The last element can be accessed by $S[\#S]$, which for simplicity shall be denoted S_{last}. All but the last element can be accessed by $S_{nonlast}$. An empty sequence is denoted $\langle\rangle$.

A trace $t = \langle e_1, e_2, e_3, \ldots, e_n \rangle$ of a process P is a sequence of communication events $e_i \in \alpha P$, where αP is the alphabet of allowed events for process P, in which P has engaged at some point in time [2]. An event is of the form $ch.v$, where ch is the CHANNEL over which the communication occurred and v is the value communicated. The operator $\leq$ denotes that one trace is a prefix of another. For example, $s \leq t$, where s and t are both traces, indicates that s is a prefix of t. The expression $t \downarrow ch$ denotes the sequence of communications over CHANNEL ch recorded in trace t.

In the definitions below, **N** denotes the natural numbers, **I** the integers, and **Q** the rational numbers. *Unit* is an unspecified primitive entity. For example, a *Unit* is the smallest component of a MESSAGE. A MESSAGE is a finite sequence of *Units*.[4] The following data types, constants, and functions are defined for the formal model:

M is the set of MESSAGEs that can be processed by the ECA, where each MESSAGE is a finite sequence of *Units*. Four subsets of MESSAGEs are identified: $M_{EF} \subseteq M$ that is exempt from format restrictions; $M_{EB} \subseteq M$ that is exempt from bypass rate restrictions; $M_{EC} \subseteq M$ that is exempt from encryption, e.g., all-bypass MESSAGEs; and $M_{IG} \subseteq M$ represents MESSAGEs that originate within the ECA.

F is a set of FIELDs, where each FIELD is a finite sequence of *Units* and represents a value. The function *valueF*: $F \rightarrow \mathbf{I}$ returns the value.

B is a set of FIELD sequences, where each FIELD sequence (representing the BYPASS DATA of a MESSAGE) is a finite sequence of elements from F. The function *lengthB*: $B \rightarrow \mathbf{N}$ returns the declared length of the BYPASS DATA. The declared length is specified by a FIELD in the sequence. The function *Byp*: $M \rightarrow B$ extracts the BYPASS DATA for a particular MESSAGE.

R is a set of restrictions, where each restriction has a length value, a lower bound value, and an upper bound value. The function *lengthR*: $R \rightarrow \mathbf{N}$ returns the length value. The function *lwrbndR*: $R \rightarrow \mathbf{I}$ returns the lower bound value. The function *uprbndR*: $R \rightarrow \mathbf{I}$ returns the upper bound value.

RS is the set of restriction sequences that specify the criteria for the FORMAT CHECKS. Each restriction sequence is a finite sequence of elements from R.

C is the set of CRYPTO DATA. Each CRYPTO DATA is a finite sequence of *Units*. The function *Crp*: $(M - M_{EC}) \rightarrow C$ extracts the CRYPTO DATA for a particular MESSAGE.

[3] This definition of indexing is slightly different from Hoare's description on page 20 of Ref. 2. Hoare's traces are indexed from 0, but we prefer to index from 1.

[4] A *Unit* can be thought of as a single *bit*, i.e., a Message is a finite sequence of *bits*. However, *Unit* can also represent a *byte*. We decided that it was unnecessary to specify the underlying representation here.

Z is the set of TIME values, and $Z \subseteq \mathbf{N}$. $Z0$ represents the initial TIME value received by the ECA.

Ch is the set of external communication CHANNELS $Ch = \{RI, RO, BI, BO, CI, TI\}$ for the ECA. See Fig. 2.

ECA is a process. The alphabet of ECA, αECA, is $\{RI.m, RO.m, BI.m, BO.m, CI.k, TI.z \mid m \in M \wedge k \in K \wedge z \in Z\}$ where K is the set of cryptographic KEYS.

T is the universe of traces, i.e., the union of trace sets of all imaginable processes. Formally, $T \equiv traces(CHAOS_U)$ where U is the universe of events and $CHAOS$ is a process that can engage in any event at any time. (See Ref. 2, p. 126.)

$ECAEncrypt$ is the cryptographic encryption transform $ECAEncrypt{:}M \rightarrow C$ that is applied by the ECA to CRYPTO DATA. $ECAEncypt$ is subject to NSA Type I cryptographic constraints.

δ is the ALLOWED BYPASS RATE, and $\delta \in \mathbf{Q}$.

σ is the initial number of *Units* permitted to bypass the cryptographic function of the ECA, and $\sigma \in \mathbf{N}$.

$\mathcal{F}$ is the transformation function $\mathcal{F} : M \rightarrow M$ that the ECA applies to a MESSAGE; $Crp(\mathcal{F}(m)) = ECAEncrypt(m)$ where $m \in M$.

Formal Assertions

In the following assertions, $t1 \in T$.

1. ECA sat CorrectEncryption

CorrectEncryption($t1$)
$$\equiv \forall t2 \in T, \forall m1 \in M :$$
$$((t2 \leq t1 \wedge t2 \neq \langle\rangle \wedge t2_{last} = BO.m1 \wedge m1 \notin M_{EC})$$
$$\Rightarrow (\exists t3 \in T, \exists m2 \in (M - M_{EC}) :$$
$$t3 \leq t2 \wedge t3 \neq \langle\rangle$$
$$\wedge (t3_{last} = RI.m2 \vee m2 \in M_{IG})$$
$$\wedge Crp(m1) \leq ECAEncrypt(m2)))$$

A MESSAGE that is subject to encryption and is leaving the BLACK INTERFACE of the ECA must be the encrypted transformation, specifically $ECAEncrypt$, of some other MESSAGE that was received previously at the RED INTERFACE or was generated internally by the ECA.

2. ECA sat CorrectFormat

CorrectFormat($t1$)
$$\equiv \forall t2 \in T, \forall m1 \in M :$$
$$((t2 \leq t1 \wedge t2 \neq \langle\rangle \wedge t2_{last} = BO.m1 \wedge m1 \notin M_{EF})$$
$$\Rightarrow (lengthB(Byp(m1)) = \textstyle\sum_{i=1}^{\#(Byp(m1))} \#(Byp(m1)[i])$$
$$\wedge (\exists rs1 \in RS, \forall i \in \mathbf{N} :$$
$$i \geq 1 \wedge i \leq \#rs1$$
$$\wedge \#(Byp(m1)) = \#rs1$$
$$\wedge valueF(Byp(m1)[i]) \geq lwrbndR(rs1[i])$$
$$\wedge valueF(Byp(m1)[i]) \leq uprbndR(rs1[i])$$
$$\wedge \#(Byp(m1)[i]) = lengthR(rs1[i]))))$$

All MESSAGEs transmitted from the BLACK INTERFACE of the *ECA* must satisfy the FORMAT CHECK: the declared length of the BYPASS DATA must equal the actual length; the value of each FIELD must be within range; and the length of the FIELD must satisfy the restriction.

3. *ECA* sat CorrectBypassRate

$$\begin{aligned}&\text{CorrectBypassRate}(t1)\\&\quad\equiv \forall t2 \in T, \forall m1 \in M :\\&\qquad((t2 \leq t1 \wedge t2 \neq \langle\rangle \wedge t2_{last} = BO.m1 \wedge m1 \notin M_{EB})\\&\qquad\Rightarrow \exists z1 \in Z :\\&\qquad\quad(TI.z1 \text{ in } t2\\&\qquad\quad\wedge \mathit{TotalBypass}(t2) < \sigma + \delta \times (z1 - Z0)))\end{aligned}$$

$$\begin{aligned}\mathit{TotalBypass}(t) \equiv\ &\text{if } t = \langle\rangle \vee t_{last} = TI.Z0 \text{ then } 0\\&\text{elseif } t_{last} = BO.m \wedge m \notin M_{EB} \text{ then } \mathit{lengthB}(\mathit{Byp}(m)) + \mathit{TotalBypass}(t_{nonlast})\\&\text{else } \mathit{TotalBypass}(t_{nonlast})\end{aligned}$$

The amount of BYPASS DATA that can exit the BLACK INTERFACE is determined by ALLOWED BYPASS RATE, the TIME that has elapsed and the amount of BYPASS DATA already diverted around the cryptographic function.

4. *ECA* sat CorrectOrder

$$\begin{aligned}&\text{CorrectOrder}(t1)\\&\quad\equiv (t1 \downarrow BO) \trianglelefteq (t1 \downarrow RI)\end{aligned}$$

$$\begin{aligned}s1 \trianglelefteq s2 \equiv\ & s1 = \langle\rangle\\&\vee(s2 \neq \langle\rangle\\&\quad\wedge((s1_{last} = \mathcal{F}(s2_{last})\\&\qquad\wedge s1_{nonlast} \trianglelefteq s2_{nonlast})\\&\quad\vee(s1_{last} \neq \mathcal{F}(s2_{last})\\&\qquad\wedge(s1 \trianglelefteq s2_{nonlast}\\&\qquad\quad\vee(\exists m \in M_{IG} : s1_{last} = \mathcal{F}(m)\\&\qquad\qquad\wedge s1_{nonlast} \trianglelefteq s2)))))\end{aligned}$$

The number of MESSAGEs transmitted over the BLACK INTERFACE (ignoring internally generated MESSAGEs) must not be greater than the number of MESSAGEs received over the RED INTERFACE, and each MESSAGE must have been transmitted only once and in the order it was received.

INFORMAL MODEL CORRESPONDENCE

The assertions of the Informal Model correspond, one to one, with those of the Formal Model; however, the Formal Model fails to restate completely the critical requirements of the Informal Model. Namely, the formal assertion *CorrectBypassRate* restates only partially the informal assertion *Correct Bypass Rate*.

The informal assertion includes the constraint: "The ALLOWED BYPASS RATE is bounded from above by a prespecified constant rate." We decided that the benefits of specifying this constraint formally were outweighed by the complexity of the result. The formal specification was unwieldy and difficult to comprehend. We felt that it inhibited our ability to reason effectively about the critical requirement *Correct Bypass*

Rate as a whole. We decided that other means would have to be explored for gaining assurance that this constraint is enforced.

ADDITIONAL CLARIFICATIONS

This section clarifies certain aspects of the formal model to facilitate the interpretation of the model.

1. While the sets M_{EF}, M_{EB}, M_{EC}, and M_{IG} are all subsets of M, their intersection is not empty necessarily.
2. For some MESSAGE SETs, the "declared length" returned by the function *lengthB* may not represent the entire BYPASS DATA but only a portion of it. For such MESSAGEs, *lengthB* must add the length of the remainder to its returned value. Since this added value should be constant for many MESSAGE SETs, the effort to represent it in the model did not seem justified.
3. The lower bound $lwrbndR(r)$ of every restriction $\forall r \in R$ should be less than or equal to the upper bound $uprbndR(r)$. If the lower bound is strictly greater than the upper bound, then the consequent of *CorrectFormat* is always false, and no useful system can satisfy the assertion. Although this behavior is secure, it is probably not desirable.
4. The ALLOWED BYPASS RATE δ should be positive. A negative value for δ will prevent any bypass after a period of time, since the right–hand side of the consequent of *CorrectBypassRate* will become negative (and the left–hand side never can be). Although this behavior is secure, it is probably not desirable.

ACKNOWLEDGMENTS

The authors are grateful to Carl Landwehr and John McLean of NRL for their valuable assistance during the preparation of this report.

REFERENCES

1. Center for Secure Information Technology, "Overview and Required Capabilities of the ECA/KG-84A," NRL Technical Memorandum 5540-104A:cp, Naval Research Laboratory, Washington, D.C., June 1991.
2. C. A. R. Hoare. *Communicating Sequential Processes* (Prentice-Hall International, UK, LtD., London, 1985).
3. E.R. Olderog and C.A.R. Hoare. "Specification Oriented Semantics for Communicating Sequential Processes," *Acta Inform.* **23**, 9–66 (1986).
4. C. Landwehr, C. Heitmeyer, and J. McLean. "A Security Model for Military Message Systems," *ACM Trans. Comput. Sys.* **2**(3), 198–222 (1984).
5. Andrew P. Moore, "The Specification and Verified Decomposition of System Requirements Using CSP," *IEEE Trans. Software Eng.* **16**(9), 932–948 (1990).

GLOSSARY

The following terms have specific meaning for the ECA.

ACTUAL BYPASS RATE – the amount of BYPASS DATA actually diverted around the cryptographic function divided by the TIME elapsed.

ALLOWED BYPASS RATE – the amount of BYPASS DATA that *could have been* diverted around the cryptographic function, divided by the TIME elapsed.

BLACK DOMAIN – a region for processing nonsensitive information, i.e., for processing format- and rate–checked BYPASS DATA and encrypted CRYPTO DATA.

BLACK INTERFACE – the set of external CHANNELs for communicating with the BLACK DOMAIN.

BYPASS DATA – that part of a MESSAGE that is diverted around the cryptographic function.

BYPASS RATE – the rate of the diversion of BYPASS DATA around the cryptographic function, as measured in relative terms.

CHANNEL – a communication link.

CLOCK – a source for TIME.

CRYPTO DATA – that part of a MESSAGE targeted for encryption/decryption.

CRYPTO INTERFACE – an external CHANNEL from the RED DOMAIN for loading the KEY.

DEVICE – hardware capable of requesting ECA services.

FIELD – an identifiable subsequence of the BYPASS DATA.

FORMAT CHECK – a test that determines whether the BYPASS DATA of a MESSAGE is suitable for bypass through the ECA.

KEY – a seed for a cryptographic device.

MESSAGE – a block of data processed by the ECA.

MESSAGE SET – all possible MESSAGEs that can be transmitted across the network.

RED DOMAIN – a region for processing sensitive data, i.e., for processing BYPASS DATA and unencrypted CRYPTO DATA.

RED INTERFACE – the set of external CHANNELs for communicating with the RED DOMAIN.

TIME – a discrete value.

TIME INTERFACE – an external CHANNEL from the RED DOMAIN for inputting TIME.