STATEMENT OF WORK

HIS/MIT

For the Period 1 July 1976
Through 30 September 1977

INTRODUCTION

The intent of the effort described herein is to:

1) Complete work in progress on concepts to reduce Multics hardcore to permit certification.

2) Address new security issues relevant to the Guardian program, as mutually agreed upon by the subcontractor and contractor on or before 31 December 1976.

3) Complete all technical tasks and prepare a comprehensive final report covering all subcontractor effort from 1 January 1974 through 30 September 1977.

but the report is due on July 15, 1977!

Should be June 30, 1977

Section 1.0 Reduction of Hardcore to Permit Certification

The contractor will accomplish the following specific tasks during the course of this contract. The following paragraphs describe several on-going specific tasks which have been identified for the restructuring of Multics. Several of the tasks identified below involve modifications to the current (MR 4.0) Multics system. For each of these tasks, a method of measurement of progress in restructuring the Multics supervisor toward the project's goals of performance, compatibility and security must be identified.

In total, this task is a collection of many research subtasks, directed at a reduction in the size and complexity of the present Multics hardcore supervisor (Ring zero). Many of the areas currently under investigation continue to generate design approaches which could be useful in the long-term kernel design and development.

Task 1.1 Restructuring of Traffic Control

This subtask has developed the design for a restructured traffic control that partitions its functions into two levels. The lower level multiplexes the hardware processors of the system among a fixed number of virtual processors. By establishing in advance the number of these virtual processors, this low level processor multiplexer need make no use of the Multics virtual memory facilities. As such, there can be a strict isolation and ordering between the multiplexer and the virtual memory. The higher level multiplexer assigns and coordinates some of the virtual processors among all of the currently operating real Multics processes. This higher level scheduler then uses all of the available facilities of the Multics virtual memory, since those facilities are implemented at a lower level.

The technical note which describes the theoretical advantages of the structure has been completed. The next step in this task is to complete the detailed design and implementation study. An implementation schedule will be established when this determination is made. Following the implementation, a technical note will be prepared which discusses the conclusions that were drawn from this implementation. These technical notes will include discussion of the issues such as performance, compatibility and security impact.

Task 1.2 Separation of Page Control and Segment Control Functions within the Active Segment Table (AST).

A detailed investigation into the structure and behavior of the Multics Directory Control subsystem and the Active Segment Table has been completed.

Currently, the Active Segment Table, a central data base in Multics, is not specifically organized for functional modularity, in that it contains two distinct types of information. The AST contains page table words, needed by Page Control, and other variables which describe per segment attributes rather than attributes of individual pages. These latter variables are used by both segment control and page control and causes somewhat needless interaction between the two subsystems.

This subtask will show that it is possible to partition these variables in such a way that, in every case, they are used by either page control or segment control, but not both. This separation will lead to a significant simplification in the algorithm supported by page control. In particular, it is possible that a thorough separation of page control and segment control in this way would lead to a further simplification in the locking strategies required during the handling of a page fault and a segment fault.

Nearly all the modes of interaction currently existing between Segment Control and Page Control have been identified. Using the understanding and insight derived from this study, a proposal for the correct type of interaction will be prepared. The result of this study will be a specification for the interface between segment control and page control. This specification will provide the specific implementation that is appropriate to demonstrate the validity of the specification. This specification will be revised if experimentation shows that additional modes of the page control/segment control interaction still exist. This task will then be completed with the preparation of a final technical note.

Task 1.3 Study of Multics System Initialization

This task is defining a methodology for the initialization of the Multics system which is simpler, and more structured, than the current strategy. The development of this methodology is considered very important, not only to help systematize a complex area of Multics, but also to identify a technique which validates the secure initial state of Multics.

With the design of this task now complete, trial implementations of a hardware reconfiguration will be performed to demonstrate the validity of some of the proposed designs made in this study. Preliminary coding of other implementations will be undertaken to demonstrate the gains made by this new strategy over the one currently used. This task will conclude with a technical note which identifies both the design issues and the experimentation which was undertaken.

Task 1.4 Provision of Breakproof Environment for User Programming

As various parts of the operating environment are removed from the Multics hardcore supervisor, the question arises as to where they should be relocated. If they are placed in the same ring as the user's executing programs, then they are subject to inadvertent destruction by a user's programming error. It is very desirable that removal of software from the Multics supervisor does not lead to a reduced robustness of the programming environment. A consistent collection of user support programs that can be protected from destruction by casual error of the user is now being defined. This collection might include, for example, the linker and the linker's data bases: the linkage sections and the Linkage Offset Table (LOT). This task will provide a practical solution to the problem and will further demonstrate the utility of the Multics ring structure for self protection of user software.

The implementation of the selected environment is now underway and will be completed. When completed and evaluated by experimentation, the techniques selected by this study can be verified. A final technical note describing the study, the design, implementation and experimentation will be prepared.

Task 1.5 Page Control Restructure

This research task is utilizing several asynchronous parallel processes to perform the various functions of page control. In particular, separate processes were used to remove pages from Main Memory and from the paging device so that a minimum free storage pool would always exist to be available for servicing page faults. This task has demonstrated that the use of parallel processes in this context provides an intrinsic simplification to the algorithm.

The theoretical design and trial implementation has been documented in a technical report. The remainder of this task will be the preparation of a technical note which discusses the performance implications of this multi-process page control. This technical note will be prepared when the corresponding experimentation, currently in progress, is completed.

Task 1.6 Support of User Defined Object Types

This task deals with the modular decomposition of Multics as a technique to simplify certification. In particular, this study focuses on the intermodule relationships which may exist. It is presently believed that just a few relationships, each with a simple explanation, are sufficient to express intermodule dependencies in a large number of cases. To support this claim,

a virtual memory subsystem -- similar to a Multics virtual memory enhanced to support type extention -- is considered as a case study. The intermodule relationships will be specified and a modularization of the Multics virtual memory will be identified.

The technical note describing this study is now being prepared. The completion of this note will complete this task.

Task 1.7 Study of System Reliability and Recovery from System Errors

This study is developing a model of the occurrence and handling of system errors in Multics. In terms of this model, the missing-segment fault path is being traced to develop a characterization of the structure of Multics with respect to errors.

A technical note which identifies the issued studied to date will be written.

Section 2.0 Networking Interface Studies

Task 2.1 Study Definition

The purpose of this area of study is to identify basic issues, develop design concepts and determine the impact of connecting a secure Multics and its Secure Front End Processor (SFEP) into an operational network. The output of this task will be a definition of specific tasks to be accomplished by the subcontractor. Candidate tasks which will be considered during the development of the study plan are contained in Attachment C. Other tasks consistent with the overall objectives of Project Guardian networking and as agreed upon by Honeywell should be considered.

The approved plan resulting from this task will form the basis for subcontract extension beyond 31 December 1976.

Section 3.0 Technology Transfer Tasks

Task 3.1 Continuing Support Activity

This task covers a collection of activities to communicate the insights and understanding gained over the course of the project to Honeywell and to other organizations creating certifiably secure systems. It involves writing of internal memoranda and

publishable papers discussing results from different points of view. Publishable paper requires Honeywell approval. It also includes attendance at meetings and discussion and review of ways in which the results of the research projects can be applied to the Multics and SFEP kernel designs.

Section 4.0 Project Management

Task 4.1 Technical Reviews

The subcontractor shall hold periodic technical design reviews as requested and agreed upon between the subcontractor and the contractor. There will be at least one such technical review per year. These design detailed discussions will provide a detailed discussion of the technical progress of each subcontractor task.

Task 4.2 Subcontractor Task Schedule

The subcontractor shall perform the listed tasks defined in this Statement of Work in conjunction with the mutually agreed upon Subcontractor Task Schedule identified in Attachment A. If the tasks cannot be performed according to this schedule, then the subcontractor shall notify the contractor immediately so that impact can be assessed and a revised schedule can be agreed upon.

Task 4.3 Data Items

The subcontractor shall deliver those data items listed in the Subcontractor Data Requirement List (SDRL) as shown in Attachment B.