Poseidon House
Castle Park
Cambridge CB3 0RD
United Kingdom

TELEPHONE: **Cambridge (01223) 515010**
INTERNATIONAL: **+44 1223 515010**
FAX: **+44 1223 359779**
E-MAIL: **apm@ansa.co.uk**

APM

Information Systems Security; ARPA BAA 95-15 proposal

Owen Rees

Abstract

The Advanced Research Projects Agency (ARPA) is soliciting proposals for research in various aspects of computer and network security, to create and integrate advanced security technologies for the DII, NII, National Challenge problems, and defense uses. The reference number for this solicitation is BAA 95-15. This document is a proposal in response to that solicitation. This is a joint proposal between APM, MIT and OSF Research Institute.

The technical problem created by that business problem is set out in the ARPA Commerce Business Daily (CBD) announcement, in three topic areas. For infrastructure protection, proposals are sought for seamlessly integrated security using modular services. For Protection of End-systems, ARPA is seeking technology to allow geographically separated parts of an organization to interact as if they shared a common security perimeter. For Assurance,: proposals are sought for prototype experimental system structuring languages, analysis methods, and systems development tools and development environment to express the structure of information systems, reason about their security and other properties, and allow efficient and secure implementations.

The solution being offered is based on the principles set out in APM.1006.01 “A Framework for Federating Secure Systems”, APM.1007.01 “Security Services”, and subsequent work done at APM, MIT, and elsewhere.

APM.1454.00.01 **Draft** 7th April 1995
Marketing and Contracts

Distribution:
Supersedes:
Superseded by:

Proposal for BAA 95-15
Information System Security Without Global Trust

Topic areas:

Infrastructure Protection; Protection of End-systems; Assurance

Technical Contact:

Note: Technical point ofcontact including: name, telephone number, electronic mail (if available), fax (if available) and mailing address,

Primary contact, and for MIT-LCS: Dr Karen R. Sollins, Massachusetts Institute of Technology, Laboratory for Computer Science, 545 Technology Square, Cambridge, Massachusetts 02139, USA; phone +1 617 253 6006; fax +1 617 253 2673; email <sollins@lcs.mit.edu>.

For APM: Mr Owen Rees, APM, Poseidon House, Castle Park, Cambridge CB3 0RD, UK.phone +44 1223 568926 fax +44 1223 359779 email <rtor@ansa.co.uk>.

For OSF-RI: Ms Mary Ellen Zurko,OSF Research Institute,11 Cambridge Center,Cambridge, MA 02142, USA; phone +1 617-621-7231; fax +1 617-621-8696; email <zurko@osf.org>.

Administrative Contact:

Note: Administrative point of contactincluding: name, telephone number, electronic mail (if available), fax (if available) and mailing address.

MIT-LCS

APM: Mr Rob van der Linden, APM, Poseidon House, Castle Park, Cambridge CB3 0RD, UK.phone +44 1223 568928 fax +44 1223 359779 email <rvdl@ansa.co.uk>.

OSF-RI

Summary of the schedule:

Note: Summary of the schedule and milestones for the proposed research, including total base cost, estimates of base cost in each year of the effort, estimates of itemized options in each year of the effort, total cost (including options), and cost sharing if relevant.

Contractor's type of business:

Note: Contractor's type of business, selected among the following categories: "LARGE BUSINESS," "SMALL DISADVANTAGED BUSINESS," "OTHER SMALL BUSINESS," "HBCU," "MI," "OTHER EDUCATIONAL," or "OTHER NONPROFIT."

MIT-LCS: OTHER EDUCATIONAL

APM: OTHER SMALL BUSINESS

OSF-RI: OTHER NONPROFIT

Note: This section provides an overview of the proposed work as well as an introduction to the associated technical and management issues.

A Innovative claims

Note: A. {1} Innovative claims for the proposed research. This page is the centerpiece of the proposal and should succinctly describe the unique proposed contribution.

The research will develop a practical strategy for establishing secure interaction between systems that are currently islands of security, without requiring either the construction of a global trusted infrastructure or any other form of global agreement.

The focus will be on developing a strategy that scales well, and demonstrating its practicality by implementing a prototype that shows how the elements fit together. The major theme will be to establish only that level of trust that is necessary for the required interaction. The complexity of both the model and the system will be determined only by the number of parties that need to be trusted in a particular interaction, and not by the overall size of the distributed system within which the interaction takes place.

This strategy exploits the object-based approach to distributed systems of the ISO Reference Model for Open Distributed Processing (ISO 10746). In particular, the strategy is based upon the principle that the interacting objects are autonomous, and are responsible for their own security. This is in contrast to current approaches that depend upon an external infrastructure in which each layer must be more trusted that the layer it supports, and the trust must be absolute rather than limited to a specific function.

The object-based model is well matched to a world in which resources are owned and controlled by many independent organisations that need to protect their own interests while interacting with other organisations. The research will develop the models of transfer of limited authority, and of delegation that are necessary to support authorisation and access control in this environment.

B Deliverables

Note: B. {2} Deliverables associated with the proposed research. Include in this section all proprietary claims to results, prototypes, or systems supporting and/or necessary for the use of the research, results, and/or prototype. If there are no proprietary claims, this should be stated.

B.1 Outputs from the research

The output from the programme will be a set of 6 integrated reports, and a prototype to demonstrate the concepts.

1. Report: The self-defence model;
2. Report: Authority transfer protocols;
3. Report: Key management and federation;
4. Report: The applications toolkit;
5. Prototype: Demonstrator
6. Report: Life cycle support for security in distributed systems
7. Report: Accreditation and conformance

All documentation and source code produced under this contract will be available to project partners for research and evaluation purposes. Commercial licenses to all source code developed during the project will also be available on a fair and equitable basis in order to encourage COTS products from system vendors, system integrators, and software suppliers.

Proprietary rights to ANSAware as set out in the ANSAware Source Licence Agreement will remain with APM. APM does not impose any restictions on publications by project partners resulting from experience gained from the use of ANSAware. Applications or other code written during this project for use with ANSAware will be treated as project results.

C Statement of Work

Note: C. {3} Statement of Work (SOW) written in plain English, outlining the scope of the effort and citing specific tasks to be performed and specific contractor requirements.

Note: We need separate SOW for MIT.

The proposed programme of work consists of six tasks which will produce reports, and a seventh task which will produce a demonstrator prototype.

0. There will be an initial start-up phase in which the participants will establish coordination between the teams, and establish the technical baseline for the work. This process will also refine the plan, and in particular, will specify how the work on the early tasks is split between the participants.

1. Develop a business model of security based largely on human interaction, and the concepts of transfer of authority, permission, responsibility, etc. This model would be consistent with the Enterprise and Information viewpoints of the ISO Reference Model for Open Distributed Processing (RM-ODP), which is the target framework for OMG.

2. Develop a technical model of transfer of authority, etc, within systems, based on Access Certificates and associated signatures and seals. These protocols would be analysed rigorously using present techniques for analysis of belief in authentication protocols.

3. Develop a non-hierarchical model of key management, and integrate this with emerging models of federation for interaction between security domains.

4. Specify a modular toolkit of application components that could be integrated into applications by language and tools systems, so allowing them to provide for their own security. This would behave in the same way as selective transparencies already specified in RM-ODP (although strictly not a transparency of quite the same form as the others).

5. Produce a security demonstrator, based on ANSAware (or some other suitable platform) of the concepts derived above. (By request from ARPA, this section needs to be expanded, with a proportionate increase in costs).

6. Specify rigorously, and possibly formally, the minimum needed in security kernels for enforcement of encapsulation, taking note of the need for protection in virtual memory, and on back up media, as well as in real memory address spaces. Also specify the model for shared secrets to be set up through parent-child associations, particularly in bootstrap situations. Consider this first in a single domain, and then extend it to multiple domains and integrate it with the model of federation introduced in 3 above.

7. Consider the accreditation and conformance issues that derive from the above model, and whether accreditation certificates could be integrated with the model for transfer between systems using the same protocols as for transfer of authority, etc. This component is more speculative than the above, and will depend on earlier work.

D Description of the results

Note: D. {5} Description of the results, products, transferable technology, and expected technology transfer path.

D.1 Self defence in distributed systems

Most present security proposals assume central infrastructure control. This requires that all systems implement the same security infrastructure, that the specification they agree upon is flawless, that all systems be trusted properly to adopt it, and that all implement to the same quality. This is unrealistic. Whereas such a system could be specified technically, it is not practical or feasible to impose such a rigid technical standard on all participants. Security in distributed systems can only be based upon a principle of self defence, which implies that applications must be responsible for themselves, even if they choose to transfer or delegate this responsibility. A model will be presented that shows how applications can take primary responsibility for their own security, with only minimum support needed from the infrastructure.

D.2 Transferring authority, delegating responsibility and passing permission in distributed systems

Most present security models assume that all security related actions can be traced back to individual users, and assume that administrators can limit the actions of those users through system directives to the infrastructure. This is unrealistic in a distributed system of truly global scope, since there can be no ubiquitous security infrastructure, and no central point of control. Transfer of authority, delegation of responsibility, or granting of permission, has to be possible from one user to another, just as is true in human interaction. This also has to be possible from users to applications, so that applications can act autonomously. But, at present, applications do not have the means to take responsibility for their own security. A model of transfer of authority will be presented that is more consistent with normal human practice, and would allow transfer of authority to and between applications. It will be expressed in protocols that could be system implemented.

D.3 Non-hierarchical key management in distributed systems

Most present security models rely on a hierarchy of administrative authority, and a hence a hierarchy of key management. This takes the form of authentication or certification servers that derive authority from similar servers of higher authority and wider scope. Ultimately, one would have to place trust in national or international authorities. In a large scale distributed system of global scope, it is unrealistic to expect such trust in single points of failure. Taking a general model of federation in distributed systems as a baseline, it will be shown how key management can be handled on a peer to peer basis, without the need for a hierarchy. The federation model would apply to the security services themselves, so that they become part of the system, and do not somehow exist outside it (such as in a separate infrastructure). Also, from a small scale perspective, applications currently have to trust the infrastructure to handle keys, thus depriving them of the

means to protect themselves. It will also be shown how minimum (and easily accredited) infrastucture support for object encapsulation would enable applications to manage their own keys.

D.4 A security toolkit for distributed applications

At present, security control is largely built into infrastructures. But with minimum infrastructure support for object encapsulation, applications could take responsibility for their own security. To support this, application writers would need to delegate responsibility to system provided services that support a suitable security policy, and/or acquire services from a modular software library. These could be requested through language declaratives, or be bound dynamically into applications in response to system events. The process could further be automated by application construction tools that insert library routines, thereby supporting security independent application programs. In ODP terms, the effect would be of a selective security transparency, with system default security provided implicitly by library additions, except where applications explicitly request otherwise. This theme will be developed further, both through a discussion of the engineering needed to support the computational model, and a discussion of the implied transfer of responsibility for security from the infrastructure to the software construction tools.

D.5 Demonstrator

Enhancements to an existing distributed system platform to demonstrate how the strategy can be implemented, and how the mechanisms interact. This will be in the form of a prototype that can be made available to organisations intending to develop products that fit within the framework. The prototype will demonstrate that the results described in the reports form part of a coherent overall approach to the problem.

D.6 Life cycle support for security in distributed systems

Systems enter and leave networks in real time. It is not possible to re-boot everything to permit additions to the security infrastructure or the authority hierarchy. Within domains, authority has to be transferred in parent-child procreation such that child objects could build shared secrets with peer objects. Between domains, peer relationships have to be established between trusting partners by administrators taking on parental roles, rather than a dictatorial ones. A model will be presented that shows how security can be maintained in applications during bootstrap processes, and during reconfiguration of dynamically evolving networks. In particular, it will be argued that object encapsulation is the necessary and sufficient minimum to support the model. Only very limited system accreditation would be needed before systems could be invited to join the network community.

D.7 Accreditation and conformance

Although the requirement for establishing trust has been reduced, it is still necessary for the prospective parties to an interaction to establish the level of trust that each requires. Each party to an interaction must be able to determine not only that the other parties are authorised to participate, but also that they are are constructed and are operating at an appropriate quality level. Systems that have accreditation and conformance certificates should be able to demonstrate this to the parties that require such assurance. The issues involved in adapting the authority transfer model to convey accreditation and quality assurance will be presented.

E Cost, schedule, and milestones

Note: E. {1} Cost, schedule, and milestones for the proposed research, including estimates of cost for each task in each year of the effort and total cost.

The total cost is estimated at $740K, as set out in the tables below. They show days per task and allocation to participants, and the overall budget for effort, equipment and travel.

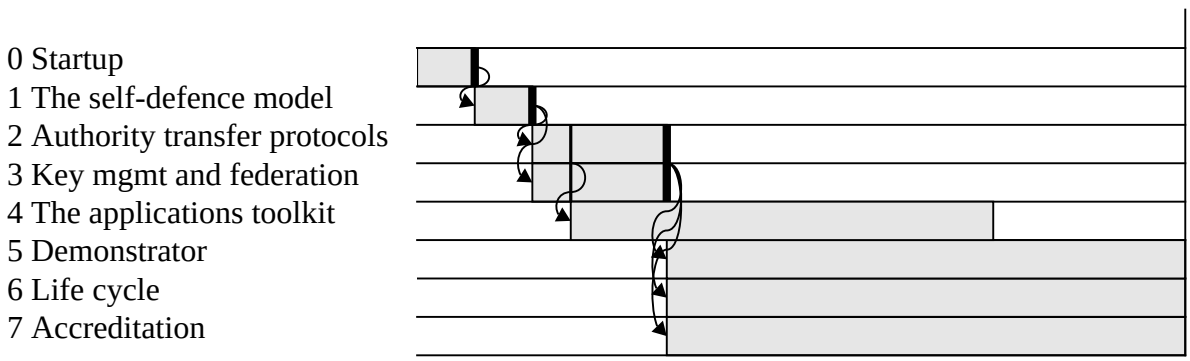
E.1 Total cost and effort

Table 1: Totals

		MIT-LCS		APM		OSF-RI		All	
		days	cost $	days	cost $	days	cost $	days	cost $
Effort totals		305	192150	290	237800	305	250100	900	680050
Equipment			10000		20000		0		30000
Travel			5000		20000		5000		30000
Total			207150		277800		255100		740050

The figure below shows the schedure for the tasks, and indicates dependencies between tasks.

Figure 1: Dependencies and schedule of tasks



E.2 Year 1 milestones

0. Startup

1. Report: The self-defence model;

2. Report: Authority transfer protocols;

3. Report: Key management and federation;

Table 2: Year 1

		MIT-LCS		APM		OSF-RI		All	
		days	cost $	days	cost $	days	cost $	days	cost $
Year 1 tasks		75	47250	150	123000	75	61500	300	231750
Year 1 meetings/QA		33	20790	33	27060	33	27060	99	74910
Year 1 equipment			10000		20000		0		30000
Year 1 Travel			2500		10000		2500		15000
Year 1 total		108	80540	183	180060	108	91060	399	351660

E.3 Year 2 milestones

4. Report: The applications toolkit;

5. Prototype: Demonstrator

6. Report: Life cycle support for security in distributed systems

7. Report: Accreditation and conformance

Table 3: Year 2

		MIT-LCS		APM		OSF-RI		All	
		days	cost $	days	cost $	days	cost $	days	cost $
Year 2 tasks		150	94500	60	49200	150	123000	360	266700
Year 2 meetings/QA		47	29610	47	38540	47	38540	141	106690
Year 2 travel			2500		10000		2500		15000
Year 2 total		197	126610	107	97740	197	164040	501	388390

F Technical rationale

Note: F. {12} Technical rationale, technical approach and constructive plan for accomplishment of technical goals in support of innovative claims and deliverable production.

F.1 The problem

All present proposals for Open Systems Security, such as those submitted to OMG, assume central infrastructure control. This requires that all systems implement the same security infrastructure, that the Publicly Available Specification agreed upon is flawless, that all systems may be trusted properly to adopt it, and that all implement to the same quality. This is unrealistic outside a closed community. Whereas if all caveats were met the present proposals would be technically sound, they are not politically or practically sound. They are neither economically sound, since not everyone has the same security concerns and would be prepared to make the same investment to protect themselves.

With controls in the infrastructure, present proposals also assume that all security related actions can be traced back to individual users, and that administrators can manipulate all the controls through system directives. This again is unrealistic. There must be provision for transfer of authority, delegation of responsibility, or granting of permission, from one user to another, just as is true in normal business practice. Furthermore, these must be transferable to systems to allow them to exhibit user behaviour without administrator control. At present, applications cannot participate in their own security, and have to rely on the manipulation of controls by central services, which because of their complexity are often mis-applied; most security failures result from user or administrator errors, failures or deliberate rule bending to improve ease if use, rather than from direct attack on security protocols.

Infrastructure control leads to a form of hierarchical key management, typically through certifications servers of wider and wider scope. Between domains, this requires users to place high levels of trust in global authorities and risk a single point failure. Reluctance of users to take this risk, particularly across national boundaries, would limit business activity. Also the hierarchical structures would have to be created in advance of potential business transactions, thus limiting the scope for ad-hoc system contracts. Within domains, applications would have to trust the infrastructure to handle keys, yet it is just as important for the applications to be protected from the infrastructure, as the infrastructure from the applications.

Systems enter and leave networks in real time. It is not possible to re-boot the world to add to the security infrastructure or the authority hierarchy. Within domains, authority has to be transferred in parent-child procreation such that child objects can build shared secrets with peer objects. Parent objects need to assist child objects to establish shared secrets with their peers, but without the parents being able to discover the secrets themselves. Between domains, provision is needed for peer relationships between trusting partners to be created by administrators taking a parental role, rather than a dictatorial one.

One system will only trust another that has a certificate of accreditation. The more function that is placed in the infrastructure, the more difficult it will become to accredit a system, particularly when the system configuration could vary, and the system will evolve through maintenance or enhancement. With present infrastructure based proposals, particularly where trust has to spread between domains, system accreditation will become impractical.

F.2 The solution

In opens systems, security must be based on a principle of self defence. Applications must be responsible for themselves, even if they choose to transfer or delegate this responsibility. They may choose to place trust in local security services, but not necessarily in remote ones. A model is required whereby a service may protect itself from any client that attempts to invoke a service it offers. It can only do this if it is responsible for managing its own keys, and if it is protected from the infrastructure.

Where applications are responsible for defending themselves, they can migrate from one environment to another, without them necessarily having to place more than minimal trust in the host infrastructure.

All key management must be handled on a peer to peer basis, without the need for a hierarchy. This requires integrating security with a model of federation. It also requires that the federation model should apply to the security services themselves, so that they become part of the system, and do not somehow exist outside it (such as in a separate infrastructure).

Where security responsibility rests with the applications, they need a modular security toolkit to offer them security services. These would be consolidated into applications in response to language declaratives, or be bound dynamically into applications in response to system events. Tools systems would support application construction, inserting library routines where necessary, thereby allowing security independent application programs. This provides a form of selective security transparency, allowing security to be provided implicitly by library additions, or explicitly by applications, but independent of the infrastructure.

Support from the infrastructure to allow applications to defend themselves must be the absolute minimum necessary. This requires encapsulation of software objects, to prevent any one object interfering with any another (other than through the service interfaces offered by the object); provision for objects to set up shared secrets with peer objects or security services; and provision for objects to transfer authority, responsibility and/or permission to peer objects. Within these provisions, encapsulation is paramount, so that the infrastructure may provide the means for secrets to be shared, and the means for applications to store and protect keys, without itself being able to know them. This can only be guaranteed by a process of code inspection and accreditation, but of a security kernel that is very small and easily inspected. Thus, although fraud and abuse are always possible, the degree of trust that has to be placed in the systems of business partners can be minimised.

F.3 Technical approach

The research will start with the set of principles set out below, and will refine and augment these principles into a set that can be used in practice. The research goal is to demostrate that these principles can be use in practice, and that they lead to systems which are demonstrably secure, and which scale.

Underlying principles:

- that object encapsulation be rigorously enforced — no back doors;
- that each object is responsible for enforcement of its own security policy;
- that any transfer of privilege into or out of an object is at the discretion of the object;
- that each platform infrastructure provides encapsulated object creation services;
- that encapsulated objects are physically protected from direct interference from all other objects (including operating system objects). Any modification of internal object state is done by the object, at its discretion, in response to invocations of operations within interfaces that it provides;
- that the infrastructure provides for interception of all system calls for physical resources, and allows monitoring of them by the platform infrastructure. In effect, the platform infrastructure encapsulates all physical resources within its encapsulation boundary;
- that when an object is granted a physical resource, that resource is presented in a clean state, devoid of any remnants of past ownership;
- that when an object releases a physical resource, that resource is returned in a clean state, devoid of remnants of past ownership;
- that for practical engineering of object-based security on existing platforms, the physical unit of encapsulation will be a capsule, where capsules contain sets of objects with like security requirements;
- that objects are primarily responsible for their own security policy and for enforcement of that policy. System designers may choose to have objects pass this responsibility to common security services, but ultimate responsibility rests with the object;
- that an object address cannot be secret. An address can always be manipulated by transparency mechanisms. Even if measures were attempted to prevent release of an address, a spoof which might address an object could always be constructed and tried;
- that the platform infrastructure must rely on the platform engineering for hardware monitoring of access to system resources and, in particular, must rely on hardware enforcement of security boundaries;
- that it is recognised that the security of the platform infrastructure can be no better than the platform engineering will allow;

- that for purposes of accreditation, the security services offered by the platform infrastructure will be limited to those that are essential, and that the platform infrastructure code that offers them will be small, self contained, and easy to inspect;
- that security may be improved by minimising sharing, or by limiting access paths to shared resources. Security concerns arise when services or resources are shared;
- that all objects may be deemed to have been created by the platform infrastructure, and the platform infrastructure, when it creates an object, decides and records the rights of access of the object to its interfaces;
- that access rights of an object to platform infrastructure interfaces need only be decided and recorded at the time of object creation (that is, that there is no need to change the position dynamically at some later time, even were this to be feasible with integrity maintained);
- that the platform infrastructure must control access to its interfaces using whatever facilities may be engineered from the platform operating system and hardware facilities. This mechanism for platform infrastructure access control may be distinct from access control provided by servers that are not part of the platform infrastructure.

Other principles relate to specific choices of access control model; that is, whether Access Control Lists (ACL's) or Access Certificates (AC's), or some mixture of both, are to be used.

The following assume an Access Certificate model:

- that an access certificate behaves like a capability created and issued by a server but with some essential differences:
- when presented along with a service request, the server can authenticate itself as the originator of the certificate;
- in the first instance, the access certificate is authorised to a specified recipient, who may then transfer it to a subsequent recipient. The server will decide from whom it will accept an Access Certificate for purposes of delivering a service, subject to whatever authentication and other checks the server may wish to make;
- nested signatures can be used subsequently to trace any transfers of authority. The server can then check accumulated transfers of authority against an authority transfer policy;
- that secret keys, used for purposes of authentication, are propagated by the platform infrastructure during the process of object creation;
- that given shared secrets for purposes of authentication between X and Y, and between Y and Z, it is possible, with the co-operation of Y, to set up new shared secret between X and Z which may be unknown to Y;

- that authentication between objects not associated by common ancestry, such as between systems, require some common authority, such as co-operating administrators, physically to place keys in those systems;
- that certification servers may be used to distribute keys to systems (perhaps using public key encryption) but ultimately their authority to do this derives from the agreement and co-operation of administrators.

F.4 Plan

For each of the tasks, one of the participants will take the lead role, and be responsible for the progress of the task. The task leader will be responsible for keeping all participants informed of the progress of the task, and the technical issues being worked on.

F.4.1 Startup

Leader: APM

In the startup phase of the project, the participants will:

1. establish how the work will be coordinated
   (i) ensure that the participants have the means to exchange documents including, but not limited to the mail reports
   (ii) ensure that all the participants have the means to exchange prototype code, both for experimentation and for incorporation into the demonstrator
   (iii) establish the progress monitoring and reporting process
2. establish the technical baseline to be used as a starting point
   (i) identify relevant technical papers and research notes
   (ii) review the principles listed above to establish common understanding
3. produce a more detailed breakdown of the tasks and specify how the individual work items are to be assigned to the participants
4. Obtain and install the equipment, both hardware and software, required for the project.

F.4.2 The self-defence model

Leader: APM

This task is to produce a report...

F.4.3 Authority transfer protocols

Leader: APM

This task is to produce a report...

F.4.4 Key management and federation

Leader: APM

This task is to produce a report...

F.4.5 The applications toolkit

Leader: OSF-RI

This task is to produce a report...

F.4.6 Demonstrator

Leader: MIT-LCS

This task is to implement a prototype that demonstrates...

F.4.7 Life cycle support for security in distributed systems

Leader: MIT-LCS

This task is to produce a report...

F.4.8 Report: Accreditation and conformance

Leader: OSF-RI

This task is to produce a report...

G Comparison with other ongoing research

G. {3} Comparison with other ongoing research indicating advantages and disadvantages of the proposed effort.

H Key personnel

Note: H. {2} List of key personnel and concise summary of their qualifications along with the amount of effort to be expended by each person during each contract year and other (current and proposed) major sources of support for them.

H.1 Qualifications

Dr. David D. Clark graduated from Swarthmore College in 1966, and received his Ph.D. from M.I.T. in 1973. He has worked since then at the M.I.T. Laboratory for Computer Science, where he is currently a Senior Research Scientist. His research interests include networks, network protocols, operating systems, distributed systems and computer and communications security.

After receiving his Ph.D., he worked on the early stages of the ARPAnet, and managed the development of one of the first host implementations of the ARPA network protocols. Following this effort, he worked on local area network technology, and was one of the developers of the token ring LAN. This effort directly led to current commercial products, and helped stimulate the IEEE 802.5 token ring standard.

Since the mid 70s, Dr. Clark has been involved in the development of the Internet protocol suite. From 1981-1989 he acted as Chief Protocol Architect in this development, and chaired the Internet Activities Board.

As a part of this work in protocols, Dr. Clark has made an extensive study of protocol efficiency. He guided the design and implementation of the SWIFT operating system at M.I.T., which demonstrated that a major impediment to effective data throughput is the internal structure of existing operating systems. His current research interests are protocol architectures for very large and very high speed networks. He has proposed new principles for protocols that will better support these networks of tomorrow. He is developing the methods and architecture for network that can support real-time services, the ISPN.

In the security area, Dr. Clark participated in the early development of the multi-level secure Multics operating system. He consulted on the development of SDNS, a secure version of the Internet architecture. He developed an information security model derived from commercial practices, a model which stresses integrity of data rather than disclosure control. He chaired a study committee of the National Academy of Science on computer and communications security.

In 1990 he received the ACM SigComm award for his work in Internet, and he was recently recognized by Federal Computer Week in its Federal 100 award.

Dr. Karen R. Sollins received a B.A. in Mathematics from Swarthmore College, and an M.S.E.E. and Ph.D. from M.I.T. in Computer Science. She has been a Research Scientist at the M.I.T. Laboratory for Computer Science since 1985 and during that time has both participated in research and supervised student thesis work. Her research has been under the umbrellas of two projects, first the Mercury Project, and more recently the Information Mesh. Mercury was a system to provide distributed system composition, including both a communications substrate and

tools for composing distributed applications from a set of components running on a heterogeneous set of remote hosts. The Information Mesh addresses the problems of scope, both topological in networks of information and in time, by supporting as a central goal the accessibility of information for 100 years, by providing a kernel for network based information to enable the ability to build long-lived relatiohships among identifiable nodes of information.

The emphasis of her work has been in naming and security, with particular emphasis on application requirements. In the area of naming, her doctoral thesis and related publication were on distributed name management, and she has authored a RFC proposing a white pages service for the internet. Much of the core of the Information Mesh is addressing problems of naming. She has supervised several student theses in related naming projects. Her work on security includes both a protocol for cascading authentication within the context of the Mercury communications substrate and supervision of several student theses on authentication and one on the use of security mechanisms for a mercantile protocol. Her cascaded authentication model was subsequently adopted by ANSA as the basis for a distributed security architecture and has led to further collaboration with ANSA. She has been the project leader on the Information Mesh project including supervising 5 students and a staff member. She continues to be an active participant in the research and standards community, through publications, active participant in IETF working groups, membership on program committees, and as a member of a review committee of the National Academy of Science on the HPCC.

Mr John A. Bull ...

Mr R. T. Owen Rees, MA (cantab), is a former member of BSI panel preparing UK contribution to the OSI security architecture. He worked on a contract for a UK government department analysing approaches to distributed system security, with particular emphasis on multi-level secure systems, authentication and key distribution, and requirements for enforcing separation both between and within hosts.

Working on the ANSA distributes system architecture since 1985, at first seconded from Racal, and since 1989 with APM. His special interests are security, atomicity, the computational model, the information model, and tool support for high level programming constructs. He is the author of ANSA reports on the computational model, and on the use of path expressions for concurrency control. He is also co-author of ANSA reports on the framework for federating secure systems, the atomicty model and infrastructure, and the enterprise and information models.

He is currently working on the Information Services Framework area of the ANSA Phase III programme, supported by ANSA sponsors.

Ms. Mary Ellen Zurko is a member of the two-person DCE-Web team, which is using the DCE infrastructure, including authorization and location-independent naming, to solve some of the current problems on the World-Wide Web. Before joining the RI, she pioneered GUI-based, usable tools for viewing and changing DCE information, such as ACLs, at Digital Equipment Corporation. She also led the development of the user interfaces for a privacy-enhanced mail (PEM) prototype and

an A1-target operating system. Her Master's thesis at MIT was on user attributes in distributed systems, including the support of least privilege and inter-domain trust.

H.2 Effort level and other support

Dr Clark will not charge any of his time to this proposal. He will be available as needed on a consulting basis.

Dr Sollins expects to allocate 20% of her time to the proposal. The remaining percentage is allocated to another project with complementary research objectives.

Mr Bull expects to allocate 25% of his time to the proposal. The remaining percentage is allocated to another security research project.

Mr Rees expects to allocate 50% of his time to this proposal. The remaining percentage will be allocated to the ANSA Phase 3 programme funded by the ANSA sponsors.

Ms Zurko expects to allocate …

I Proposers previous accomplishments

Note: I. {3} Discussion of proposer's previous accomplishments and work in this or closely related research areas.

This proposal is submitted by a consortium comprising:

- Architecture Projects Management Limited (APM)
- OSF Research Institute (OSF-RI)
- The MIT Laboratory for Computer Science (MIT-LCS)

The participants APM and MIT-LCS have a proven track record of collaboration, having presented a paper at ESORICS '92, "Towards Security in an Open System Federation". The work presented in this proposal builds on the earlier collaboration to take it forward into a solid technology that could form the basis of future Open System Security standards. Since the ESORICS paper, other researches have developed the ideas further in various papers and theses, and these would be considered within the overall work programme.

I.1 APM:

APM, Poseidon House, Castle Park, Cambridge CB3 0RD, UK; phone +44 1223 515010, fax +44 1223 359779; email <apm@ansa.co.uk>

APM is the company responsible for managing the ANSA research programme on behalf of the ANSA consortium members. APM/ANSA is a recognised world leader in distributed system technology, with ANSA being the major contributor to ODP. APM has a good relationship with OMG, and provides chairmanship of the CORBA 2 task force.

The ANSA consortium members presently comprise:

- GEC — Marconi
- GPT
- Hewlett Packard
- ICL
- British Telecom
- France Telecom
- Telefonica (Spanish Telecom)
- Bell Communications Research (on behalf of the Bell Regional Operating Companies)
- The UK Defence Research Agency (DRA)
- Northern Telecom (BNR — originally Bell Northern Research)
- Barclays Bank
- Iona Technologies
- Prism Telnologies

I.2 OSF

To be supplied

I.3 The MIT Laboratory for Computer Science

To be supplied

J Description of the facilities

Note: J. {1} Description of the facilities that would be used for the proposed effort.

The participants will use their existing accomodation and communication facilities.

The participants will use some existing computing equipment, plus the additional equipment acquired for the project as set out below.

J.1 Hardware

MIT-LCS will require a workstation for the graduate student to be assigned to the project. This must be compatible with the existing MIT-LCS infrastructure, and capable of running the distributed system platform which is to be used for the research. The cost breakdown in section K includes an Alpha workstation from Digital Equipment Corporation which satisfies these conditions.

APM will require ...what?

OSF-RI do not require any additional equipment.

J.2 Software for prototype

APM will supply the ANSAware distributed computing platform under its normal project license conditions, but, as a participant, will waive the project license fee.

J.3 Documentation facilities

What do we use?

K Cost breakdown

Note: K. {5} Cost breakdown to the level of major tasks and equipment for the entire contract and for each contract year. Where the effort consists of multiple portions which could reasonably be partitioned for purposes of funding, these should be identified as contract options with separate cost estimates for each. Details of any cost sharing should also be included.

- The cost breakdown by tasks and by participants is shown in the table below

Table 4:

		MIT-LCS		APM		OSF-RI		All	
		days	630	days	820	days	820	days	$total
Start up	0	20	12600	40	32800	20	16400	80	61800
Self defense	1	15	9450	30	24600	15	12300	60	46350
Auth txfr	2	20	12600	40	32800	20	16400	80	61800
Key mgmt	3	20	12600	40	32800	20	16400	80	61800
App tools	4	0	0	20	16400	90	73800	110	90200
Demonstrator	5	90	56700	20	16400	0	0	110	73100
Life cycle	6	50	31500	10	8200	10	8200	70	47900
Accreditation	7	10	6300	10	8200	50	41000	70	55500
Meetings	8	50	31500	50	41000	50	41000	150	113500
QA	9	30	18900	30	24600	30	24600	90	68100
Effort totals		305	192150	290	237800	305	250100	900	680050
Equipment			10000		20000		0		30000
Travel			5000		20000		5000		30000
Total			207150		277800		255100		740050

- MIT effort will include a graduate student assumed to be 50% of the cost and 50% of the effectiveness of a qualified engineer or research scientist, the effort and cost figures show qualified engineer equivalent days.
- Each of the six major reports will be subject to a quality review by each participant; five days per report per participant have been allowed for quality assurance review and revision.

K.1 Equipment

K.1.1 Hardware and system software

MIT-LCS require one additional workstation for use by the graduate student to be assigned to the project.

A DEC Alpha workstation will be compatible with the existing MIT-LCS infrastructure, and will run the ANSAware distributed system platform that is to be used for the prototype.

Note: Details required

APM will require ...what?

OSF-RI do not require any additional equipment.

K.1.2 Additional software for prototype

APM will supply the ANSAware distributed computing platform under its normal project license conditions, but, as a participant, will waive the project license fee.

K.1.3 Documentation facilities

Note: What do we use?

K.2 Travel costs

Each person from the UK attending a meeting in the USA (or vice-versa) is expected to require 6 days and $2500 for transatlantic air travel, accomodation and subsistence.

Note: [suggestion from Karen Sollins re number of trips] I don't know what you had in mind for travel, but I have a suggestion to make. We should plan to get together twice during each year. I would expect there would be 3 Americans (in Cambridge), and 2-3 British (also in Cambridge). In addition, there should be travel money for each of us to go to one conference during the contract period. For us in the US, getting money from ARPA, it is easier to specify foreign travel ahead of time. We don't know which conference, so that's more difficult, but for the collaboration travel that should be clear. We need to justify travel costs. Normally, we do this by using historical information doing averages. But since we haven't gone to Cambridge (UK) in a while, I don't know how to do this. John, I assume that ANSA can do this for coming to Boston. We might assume Oakland for the conference. We can certainly get an estimate of airfare. I know that we are currently paying about $1200 to go to Stockholm in July, by staying over Sat. night. If we assume that we don't have to do this travel during peek vacation time, but could do it in spring or fall (or even winter), it would cost less. The other bits can be learned - hotel and miscellaneous (food, car).

Cost figures allow for 8 person-meetings in the USA by APM staff.

Note: Assumptions about meetings for MIT-LCS and OSF-RI needed here

Where participants are attending relevant conferences, either to present the results of the research or for any other purpose connected with the research, every effort will be made to schedule project meetings so that transport and accommodation costs can be shared between the conference and the project meeting.

I. Additional Information

Note: A bibliography of relevant technical papers and research notes (published and unpublished) which document the technical ideas upon which the proposal is based. Copies of not more than 3 relevant papers can be included in the submission, one set for each of the four copies.

ESORICS paper. “Towards Security in an Open System Federation”

ANSA AR.008 “A Framework for Federating Secure Systems”

ANSA AR.009 “ANSA Security Services”