

SECTION I: Administrative

PROPOSAL BAA Number: 99-16

Organization/Company:	Trusted Information Systems, Inc. (A wholly-owned subsidiary of Network Associates, Inc.)
Title:	A Communications Security Architecture and Cryptographic Mechanisms for Distributed Sensor Networks

Technical Area:	Nanocryptography
------------------------	------------------

Technical Contact:	David W. Carman
Address:	TIS Labs at Network Associates 3060 Washington Road (Rt. 97) Glenwood, MD 21738
Telephone:	(443) 259-2374
Fax:	(301) 854-4731
E-mail:	dwcarman@tis.com

Administrative Contact:	Timothy J. Samples
Address:	TIS Labs at Network Associates 3060 Washington Road (Rt. 97) Glenwood, MD 21738
Telephone:	(443) 259-2313
Fax:	(301) 854-4731
E-mail:	tim@tis.com

Type of Business:	Large Business
--------------------------	----------------

SECTION II: Detailed Proposal Information

A. Innovative Claims

Distributed sensor networks will employ communications among large numbers of sensors remotely deployed in irregular patterns to form ad hoc distributed processing networks that can produce high-quality information with minimized resource consumption. The security properties of these networks will be of profound importance. We propose to develop and demonstrate a layered communications security architecture for distributed resource-limited sensor networks. The architecture will incorporate cryptographic security mechanisms that efficiently support the provision of integrity, authentication, and confidentiality security services, as well as survivability and robustness characteristics. Distributed sensor networks will include many small, low-power nodes distributed in a sparse and irregular network across remote and often hostile locations. Sensor networks may also include more powerful “super” nodes. To reliably support coordinated control, management, and reporting functions, sensor networks must be self-organizing with both decentralized control and autonomous sensor behavior, resulting in a sophisticated processing capability. Sensor networks must be robust and survivable despite individual node failures and/or intermittent connectivity.

We will research and develop a comprehensive communication security architecture organized into three layers: sensor data; network communications; and wireless link. The security architecture will encompass the essential communications security services (i.e., authentication, integrity, and confidentiality), and underlying security mechanisms at each of the layers. The architecture will also specify needed security support services, including an efficient sensor network equivalent to a public key infrastructure (PKI).

We will research and develop new cryptographic security mechanisms that provide security and security support services, satisfy unique sensor network requirements, and operate efficiently under the resource limitations of distributed sensor networks. These mechanisms will include: (1) the use of public key cryptography in a manner that takes advantage of its asymmetric nature to minimize power consumption; (2) the use of secret key (symmetric) cryptography in a manner that efficiently emulates public key functionality (i.e., key notarization and symmetric-key certificates); (3) the use of special-purpose hardware to accelerate selected cryptographic operations; and (4) efficient key management techniques, including adaptive selection and use of group keying, and ripple-keyed cryptography, a novel concept whereby protected keys ripple, like waves created by throwing pebbles in a pond, node-by-node, across a sensor network.

The comprehensive communications security architecture and underlying cryptographic mechanisms will be developed in two concurrent tasks: (1) study sensor environment, communications, security requirements and constraints, and develop an appropriate communications security architecture comprised of selected cryptographic mechanisms; and (2) develop, demonstrate, and analyze a prototype software implementation of the architecture to assess its ability to provide essential security services for protecting distributed sensor networks despite resource limitations.

B. Technical Rationale, Technical Approach, and Constructive Plan

B.1 Technical Rationale

The primary goal of the proposed effort is to combine new and existing cryptographic mechanisms into a secure communications architecture suitable for use by the SenseIT program. The envisioned distributed sensor network environment applies numerous constraints that impact the sensor designer's technology choices. These limitations include such factors as how sensors are deployed, how long they remain on station, and their desired cost. Since many cryptographic operations are computationally intensive, these factors directly impact the resources available for performing communications security operations.

The SenseIT program is pioneering new network-based approaches that can expect to significantly enhance the capabilities of future sensor systems. Unlike today's sensors, future deployments may include hundreds or thousands of small, low-power, low-cost sensor nodes distributed in a sparse and irregular fashion across an entire area. To reliably support control, management and reporting functions in this scattered, random-looking network, sensor networks must be self-organizing and capable of cooperative processing. Survivability, through redundancy and rapid adaptation to a dynamic environment, is of paramount importance.

Examples of fielded sensor technology include Remote Battlefield Sensor System (REMBASS), Improved REMBASS (I-REMBASS), and REMBASS II. REMBASS and its improvements provide an unattended ground sensor system that responds to seismic-acoustic energy, infrared energy, and magnetic field changes to detect enemy activities. These anti-intrusion sensors process the data and provide detection of classification information which is incorporated into digital messages and transmitted through short burst transmissions to the system sensor monitor.

Various versions of REMBASS incorporate radio repeaters that are used to extend the broadcast range of radio messages from anti-intrusion sensor to the monitoring set. Several repeaters may be used in a station-to-station chain, one sending to another, to relay messages over a long distance. The messages are demodulated, decoded, displayed, and recorded to provide a time-phased record of enemy activity. REMBASS uses remotely monitored sensors placed along likely enemy avenues of approach.

Figure 1 depicts a simple, block diagram of a typical sensor node or platform. Each sensor platform will have a reconfigurable set of at least three types of sensors. A Global Positioning System (GPS) receiver must be one of the supported sensors. Each platform will have wireless communications with multiple channels, a rechargeable battery power supply, and a CPU and memory. The CPU will be remotely programmable. Each sensor platform will include a communications subsystem and will be able to communicate with other sensor nodes and exterior control, management, and reporting nodes. Each sensor will include an execution environment for loading and executing code to control and configure the operation of the platform.

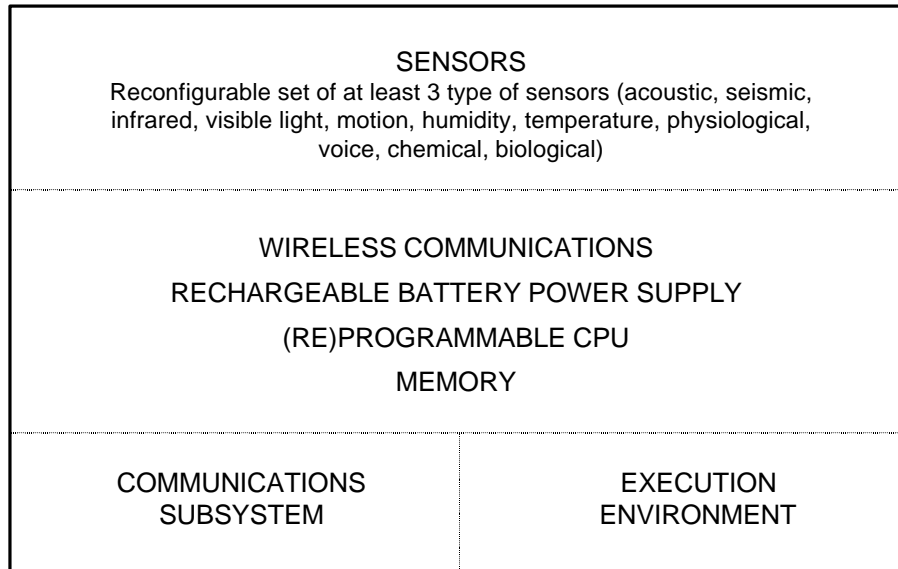


Figure 1: Typical sensor platform

Future sensors may be deployed via a variety of methods, including light infantry or airborne means for battlefield deployments, imposing severe size and weight limitations on sensor packages. Size and weight restrictions directly limit battery power capacity, which directly bounds the amount of power available for RF transmission and processor computations. To provide communications despite severe power limitations, researchers at the Berkeley Sensor and Actuator Center at the University of California at Berkeley are examining low-power communications logic (see *Ultra-Low Power Communication Logic Circuits for Distributed Sensor Networks*, Brett Warneke and Carl Chang, UC Berkeley, <http://www-bsac.eecs.berkeley.edu/~bwarneke/ee241/>). To progress the feasibility of a sensor concept called SmartDust, researchers have designed and demonstrated communications logic for a SmartDust sensor that draws only nanowatts worth of power.

The meager amount of power available at the sensor also influences the choice of the sensor's main processor. A sensor's CPU power requirements are most directly related to the type of CPU and the processor clock frequency. Running on a small battery or other weak power source, the sensor designer will be under pressure to choose a CPU that draws minimal power. As a result, sensor designer's will be more likely to chose a microcontroller such as the 20 MHz 8051 which draws 20 mA at 5 V, and can transition to a "power down" state to draw less than 50 μ A. For comparison, more modern processors such as the 450 MHz Pentium II Xeon draw 14 A at 1.5 V (not including the L2 second level cache). However, choosing a less capable processor that operates at lower frequencies increases the processing time for cryptographic operations, as will be discussed briefly.

Cost may also influence sensor CPU and memory design choices, since size of the CPU and the amount of memory affects a chip's die size, which translates directly into cost. For instance, a 10,000-gate 8051 microcontroller that can be fabricated for less than one dollar per unit is more likely to be selected for a low-cost sensor package, than a several million-gate Pentium II CPU

core that costs several hundred dollars. Similarly, RAM is fairly expensive in chip acreage terms. Therefore, any cryptographic algorithm or protocol designed for use in sensor networks must function with relatively little memory. Non-volatile memory such as EEPROM, Flash, or battery-backed RAM may also be at a premium, hence long term storage of information such as digital certificates may prove difficult.

However, conventional cryptographic algorithms and protocols tailored for modern super-scalar CPUs may not provide sufficient performance when implemented on low-gate count microcontrollers. For instance, a 1024-bit modular exponentiation, such as might be performed in the RSA or Diffie-Hellman algorithms, may take more than one minute on a 20 MHz 8051 microcontroller. The same 1024-bit modular exponentiation takes less than forty milliseconds on Intel's latest Pentium II Xeon 450 MHz CPU.

Despite the constraints of this environment, sensors offer some characteristics that give the security designer more options. For instance, sensor packaging is often tamper-resistant, thus making any sensitive information or keys stored within the package difficult for an adversary to compromise. Tamper resistance allows sensor packages to securely store keys for long periods of time, thus allowing unconventional cryptographic techniques to be used.

B.2 Technical Approach and Constructive Plan

Our basic technical approach is to develop a comprehensive security architecture comprised of an underlying set of cryptographic mechanisms suitable for distributed sensor network communications. The main concepts of this approach include:

- allocating security functions based on differing capabilities of various component types in a multi-tiered sensor network architecture,
- providing computationally efficient public key cryptography,
- emulating public key functionality via symmetric cryptography,
- analyzing the relative merits of performing selected cryptographic operations in a hardware co-processor, and
- providing efficient key management via novel techniques, including group keying and ripple-keyed cryptography.

B.2.1 Multi-tiered sensor communications architecture

Components of a distributed sensor network will likely possess vastly different computational and communications capabilities. For instance, low-power sensor nodes may have relatively little computational and communications capabilities. These limitations may significantly restrict the cryptographic operations that will be performed when attempting to provide network security. More capable sensor nodes, similar to conventional radio repeaters, can perform reasonable amounts of computation, and may support clusters of sensors to help securely route

communications to appropriate recipients. These "super" nodes may function as "security enhancers" by providing additional network security services to sensor data before it is transmitted to control management and reporting points. Conventional battlefield sensor systems, such as REMBASS, currently employ radio repeaters to extend the broadcast range of radio messages from sensors. As depicted in Figure 2, we envision that future sensor networks will similarly require and use a multi-tiered communications architecture that allows for at least two tiers of components; a standard, resource-limited sensor node; and a "super" sensor node that offers many more capabilities beyond simply blind repeating of sensor data. Our communications security architecture will examine and develop cryptographic mechanisms that benefit from this multi-tiered arrangement.

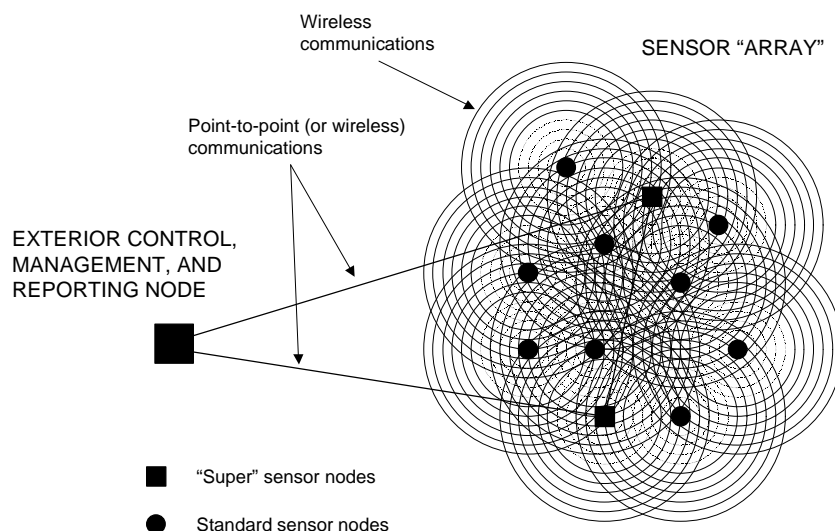


Figure 2: Multi-tiered sensor communications architecture

Verification that sensor data is legitimate may often be the most important use of cryptography in the distributed sensor network. Super nodes may provide the first, and in some cases the only, verification that data originated at a legitimate sensor. Verification makes use of various methods including verification of a sensor's ID, verification of a secure timestamp or verification of a sensor's location. Digital signatures, coupled with a public key infrastructure, may be used to give each sensor a verifiable "identity"— although we expect many sensors will lack the computational resources to provide this functionality using conventional algorithms. An important element of this research is to develop alternate techniques for providing acceptable security that are computationally efficient enough to be performed by low power sensors.

Instead of or in addition to sensor source authentication, the sensor may provide timestamping on all reported data to help prove its legitimacy. Sensors are especially well suited to providing and

verifying timestamping, since their inherent tamper-resistance provides the necessary protection to maintain a self-contained trusted time source.

Similar to timestamping, super nodes may also perform verification using location dependent cryptography or *location-stamping*. In some instances, it is more important to know where a transmission has originated rather than who originated it. In the case of distributed sensor networks, it may not be important which sensor is reporting events, but where the event is occurring. Thus, location may be more important to the intended receiver than the ID of the sensor that detected the event.

A GPS receiver embedded in a sensor platform enables location-stamping to provide a cryptographic binding between a sensor's location and the sensor data that was reported. In addition to miniaturization of GPS receivers, the sensor package's tamper resistance also facilitates this functionality since it prevents an adversary from modifying the raw location data associated with a set of sensor data. Location-stamping offers an alternative to binding sensor identity to data. The location-stamping alternative may prove valuable in instances where mass production prohibits embedding a unique identification into each sensor unit. Similar to timestamps, the location-stamp could remain associated with the sensor data throughout its lifetime as a type of digital watermark.

Once the sensor data has been verified, the super node may fuse or analyze data from multiple sensors, possibly including its own, before forwarding the processed information to destination recipients. The super node may apply strong confidentiality, integrity and source authentication to the forwarded information, services that the resource-limited sensor may not have been able to provide.

B.2.2 Layered communications security architecture

Our technical solution will be organized as a layered communications security architecture suitable for distributed networks of resource-limited sensor nodes. The layers of the sensor communications architecture include the sensor data, the network communications, and the wireless link layers. The sensor data layer includes capturing and formatting the data obtained by the sensor and the control of the data being captured. The network communications layer includes all the control structure needed to transfer and control the transfer of data throughout the distributed sensor network. The wireless link layer includes the electronic formatting and transmission of bits and handling errors when the bits did not arrive at their intended destination correctly.

The security services needed in the distributed sensor network include confidentiality, authentication, and integrity. The security support services include key management and public key infrastructure (PKI) functionality. These terms have the same definitions as those used in a general-purpose data communications network. In the communication security architecture to be designed, what is being protected changes a little, but how it is protected changes significantly from a conventional security architecture (e.g., ISO/IEC 7498-2: Information Technology – Open Systems Interconnection – Basic Reference Model – Part 2: Security Architecture).

The security mechanisms available in the distributed sensor network environment include:

- cryptography (encryption, codes),
- cryptographic authentication (hashes, message authentication codes, digital signatures),
- source authentication (device unique keys, device class keys),
- traffic flow security (message padding, false messages),
- transmission obfuscation (spread spectrum, frequency shifting),
- redundancy (multiple paths for a transmission, multiple encoding of a message), and
- geographical location stamping (high and low precision GPS positions, high precision timing, and synchronization of communications).

Security and security support services will be provided by security mechanisms selected for appropriate layers in the communications architecture. Security mechanisms will be optimized for the special security requirements in the specialized communications environment of the distributed sensor network. For instance, a resource-limited sensor node might use security services at the sensor layer while a super node might use security services at the network communications layer.

Mappings among communication layers, security services, and security mechanisms will be specified as part of the communications security architecture after analyzing the specialized security requirements of the distributed sensor network. For instance, since resource-limited sensors will apply security services at the sensor data layer, the selection of computationally efficient cryptographic mechanisms will be paramount.

B.2.3 Computationally efficient public key cryptography

Public key cryptography is naturally asymmetric, and often requires the sender and receiver to perform different amounts of computation. Since we envision a network containing nodes with different computational capabilities, we will explore techniques for taking advantage of the natural asymmetry of public key cryptography. Conventional RSA “public key” operations, such as encryption and signature verification, use a short public key exponent that allows computations to be performed up to a hundred times faster than “private key” operations. We expect that the computationally limited sensor node is capable of performing the RSA encryption and verification operations, while the super node is capable of performing the corresponding RSA decryption and signature operations. When performing RSA operations in the opposite direction, the sensor node would use a relatively short, but cryptographically secure, private exponent, while the super node would perform public operations using the resulting long public exponent. Thus, by matching the different capabilities of sensor nodes with the corresponding public key operations, we can dramatically reduce the processing required on computationally limited nodes while maintaining strong security.

Furthermore, we will explore other performance-enhancing public key cryptography techniques, such as constructing an RSA modulus in the form $n=p^kq$ vice the conventional form $n=pq$. This notion, based on work presented by Tsuyoshi Takagi at Crypto '98, is predicated on the assumption that existing factoring algorithms, such as number field sieve and the elliptic curve method, are no more effective against $n=p^kq$ than $n=pq$, for appropriate sizes for p and q .

Beyond traditional algorithms such as RSA, elliptic curve cryptosystems (ECC) may offer superior performance for providing some security services on computationally limited sensor processors. ECC has often been proposed for use in smart cards, which have similar implementation constraints as sensor processors. ECC's unique properties make it especially well suited to applications where processing capability is severely limited, since it provides the highest strength per bit of any cryptosystem known today. We plan to benchmark the performance of our enhanced RSA algorithms with those of ECC to evaluate the relative merits of these alternatives on our target sensor platforms.

Precomputing certain quantities before the actual public key operation takes place can accelerate most public key algorithms. For instance, most of the computation time involved in performing the Digital Signature Algorithm (DSA) may be computed prior to even preparing the message to be signed. The computation of the value r , the half of the DSA signature computed via the equation $r = (g^k \text{ mod } p) \text{ mod } q$, may be performed at any time convenient to the sensor platform. Significant portions of both the signature and key exchange operations of the El Gamal and ECC algorithms may also be precomputed. This precomputation may be performed much earlier than when the signature is needed, such as at a time when speed is not as critical. Furthermore, if sufficient time is available to perform the precomputation, the sensor may additionally be able to slow its processor (or math co-processor) clock frequency to conserve power during the calculation.

B.2.4 Emulating public key functionality using symmetric cryptography

Despite various methods of enhancing the performance of public key cryptography, such as speeding up RSA or using ECC, public key operations may be too slow for computationally limited sensors. This may be especially true when low power sensor nodes communicate among themselves, since we may not be able to take advantage of the asymmetric public key operations described in the previous section. In these situations, fast symmetric cryptography operations can be used to emulate public key functions by taking advantage of the tamper-resistant nature of sensors. Two techniques for achieving such emulation are key notarization and symmetric-key certificates.

Historically, cryptographic technology has used symmetric, sometimes commutative, algorithms to protect communications between pairs or groups of communicating entities (e.g., people, devices, and surrogates). It was sufficient to assure that only the authorized group of entities had the cryptographic key variables (algorithms, initial setting, stepping sequences, combining functions, parameters, limits) needed to encrypt (encipher, encode) and decrypt (decipher, decode) the information. The security perimeter included all the authorized entities and little concern was assigned to who was the sender or who was the receiver. Asymmetric cryptography

was developed to provide differentiation between the encryptor (i.e., producer) of data and the decryptor (i.e., consumer) of the data. Information was hidden in the encrypted form of the data. The decrypting key was related to, but not the same as, the encrypting key. Thus, the producer of data could be differentiated from the consumer and held accountable for its accuracy and integrity. The consumer could not create false data and claim it came from the producer. Accountability to the individual person or transmitting device can thereby be assured. Integrity was assured by redundancy of data by including a cryptographic function of the value and order of all the bits of a message in the message. Errors can thus be detected, but not corrected. Asymmetric (or public key) cryptography thus held advantages in many applications over symmetric (or secret key) cryptography.

Key notarization

Key notarization is a technique for employing high-speed secret key cryptography while providing the authentication and accountability of public key cryptography (see Smid, M.E., A Key Notarization System for Computer Networks, NBS Special Publication 500-54, October 1979). It requires explicit specification of the identities of the parties involved in a keying relationship.

In particular, given a session key s used to encrypt data between a sender i and a recipient j , a “notarized” session key ns is formed by first combining a key-encrypting key k with the identities i and j , in that order, and then encrypting the session key, i.e., $ns = E_{kA(i||j)}(s)$. Given the notarized session key ns , the correct identities must be specified, in the same order, to recover the session key s . Key notarization requires either a trusted third party (notary) or protected hardware to enforce the creation of the notarized session key and to check the identities. In this manner, secret key cryptography can produce some of the benefits of public key cryptography in the area of authentication and accountability by differentiating between the sender and intended recipient of encrypted data in an otherwise symmetric relationship. This technique is directly applicable to sensor communication, since the tamper-resistant nature of sensor nodes allows the use of notarized keys to be enforced, and the limited computational capability of sensor nodes makes frequent public key computations undesirable.

Symmetric-key certificates

The use of symmetric-key certificates is another technique for achieving the high speed of secret key cryptography while providing the authentication and accountability of public key cryptography. Like a public-key certificate, a symmetric-key certificate cryptographically binds a key with the identity of its owner. In particular, a symmetric-key certificate c_i cryptographically binds a party’s data encryption key k with its identify i by encrypting the two quantities under a secret master key mk , i.e., $c_i = E_{mk}(k || i)$. In a distributed sensor network, the master key mk would be a secret key securely embedded in each sensor node. The symmetric-key certificate could be self-generated by a sensor node and exchanged with other sensor nodes, or it could be generated and distributed by a trusted third party, similar to a certification authority. The data encryption key symmetric-key certificate can be used by other systems for confidentiality and data origin authentication purposes with assurance of the identity of its owner, namely, the identity contained in the certificate. Techniques such as this are directly applicable to sensor communications since the tamper-resistant nature of sensor nodes allows

master keys to be securely maintained, and their limited computational capability makes frequent public key computations unattractive.

B.2.5 Hardware acceleration of cryptographic computations

In addition to the algorithmic methods of achieving greater computationally efficiency to provide the processing for security and security support services, we will examine the relative merits of providing hardware co-processing capabilities within the sensor. There are two major benefits of providing hardware co-processing of cryptographic algorithms on a sensor platform: (1) the hardware co-processor may be able to complete the computation task in much less time than the main general-purpose microprocessor or microcontroller; and (2) while computations are performed by the hardware co-processor, the main processor is free to perform other sensor tasks. The hardware co-processor may be contained on a separate chip, or it may be integrated as a module on a sensor ASIC.

Many conventional cryptographic algorithms can be accelerated dramatically by being implemented as a separate hardware co-processor. Certain bit-oriented algorithms, such as the Data Encryption Standard (DES), are especially amenable to hardware since general-purpose microprocessors aren't usually capable of performing multiple single-bit operations in a single processor cycle. For instance, the S-Box look-ups, expansion, and permutation operations of DES can be sped up significantly by performing these operations in simple hardware modules that execute in parallel.

Even algorithms that are designed to execute on general-purpose microprocessors, such as SHA-1 and MD5, can significantly benefit from the parallelism that may be attained through hardware implementation. This property is especially relevant when the sensor's main processor is less capable than the target class of processors that the cryptographic algorithm was originally designed for. For instance, SHA-1 and MD5 were designed for microprocessors with 32-bit operations and several 32-bit registers, and thus do not perform very well on 8-bit microcontrollers with a limited number of registers.

Hardware co-processing often proves most valuable when used to speed up public key computations. Until the past few years, sub-second public key operations were not commonly available on general-purpose microprocessors. Recently, with the advent of Pentium-class processors capable of high-frequency performance and low clock cycle multiply operations, many public key operations can be completed in less than 100 milliseconds. Even today, however, computationally-limited platforms such as PC (formerly PCMCIA) cards and smart cards generally lack the ability to perform public key operations quickly enough with their main processor. Thus, when these platforms are required to support public key operations, they commonly use hardware acceleration of the public key operations.

An example of hardware acceleration on a computationally limited platform is the U.S. Government-developed Fortezza PC card, which contains acceleration for a symmetric cipher (SKIPJACK), a hashing algorithm (SHA-1), and for modular exponentiation (DSA and KEA). Commercially, companies such as SpyruS and Chrysalis also offer PC and smart cards that

contain hardware acceleration for cryptographic operations. Furthermore, many smart cards that implement the public-key operation-intensive Secure Electronic Transaction (SET) standard contain modular exponentiation acceleration.

Hardware acceleration of public key operations may occur in various forms, depending on the public key algorithms being used, and the level of control provided by the main processor. For operations over $GF(p)$, a hardware acceleration module might perform an entire modular exponentiation without any interaction with the main processor. Alternately, a hardware acceleration module may perform only modular multiplication or multiplication, which would require significantly more interaction with the main processor at each step. For ECC, a hardware module that provides acceleration of elliptic curve computations over $GF(2^m)$ may similarly provide significant performance improvements.

B.2.6 Efficient key management

Conventional key management and public key infrastructure techniques used for network security may not be appropriate to some or all distributed sensor networks due to the large amounts of computation that must be performed, and the large numbers of messages that must be exchanged. Group keying methods such as Authenticated Group Diffie-Hellman and Burmester-Desmedt can be computation and message efficient for relatively small groups and allow for decentralized control. Other methods, such as Key Distribution Centers, can also be efficient for relatively small groups. Hierarchical combinations of the two types of methods may provide a good basis for secure, efficient, robust communications throughout sensor networks. Other schemes such as Logical Key Hierarchy (LKH) and One-way Function Tree (OFT), which was used in the DARPA-funded Dynamic Cryptographic Context Management (DCCM) project, are superior for larger group sizes.

However, each sensor deployment may offer an entirely different environment, with varying number of sensor nodes, varying connectivity, varying numbers of control management and reporting points, and varying security requirements. It is unlikely that a single method of key management will be best for all situations. Sensor nodes must be capable of adapting to these different missions by providing a group key management scheme appropriate to the given scenario. Thus, we will examine techniques and protocols for selecting an appropriate group key management scheme among a group of distributed sensor nodes.

Furthermore, we will look at key management techniques such as a novel scheme based on preliminary research at TIS Labs called ripple-keyed cryptography, which requires little-to-no infrastructure. Ripple keying is a concept of operation for establishing cryptographic keys among distributed (and mobile) communicating entities. Protected communications of keys ripple, like waves created by throwing a handful of pebbles in a pond, node-by-node, throughout a distributed sensor network. While first conceived for active networks, ripple keying may be more applicable to sensor networks. The concept of active networks included asking some network entity to perform some service on behalf of the originator of data. However, neither the identity nor the location of the entity could be determined prior to data transmission. Ripple keying includes the concept of distributing keys based on determined (or deterministic)

capabilities rather than identities or location by continually rippling data through the network that would allow transformation to be made in the entities with the correct capabilities that could then be utilized later. A transmitter could simply request that any entity with capability X perform service Y on the data while the data is passing through the entity. While necessary and sufficient criteria and protocol have not been developed for active networks, the concept is more amenable to distributed sensor networks that have the needed capability of having protected information and information processing capability. This capability was not believed to be available in general-purpose active networks. It is proposed that ripple keying technology be researched and developed for distributed sensor networks.

B.2.7 Prototype implementation and demonstration

We plan to implement a simulation of a distributed sensor network by designing sensor objects that mimic the wireless communications environment of distributed sensor networks. Sensor objects will be composed of two types: resource-limited sensors; and “super” sensor nodes. Sensor objects will contain prototype implementations of the cryptographic mechanisms that provide the security, and security support services that comprise the communications security architecture.

Each sensor object, possibly executing in its own thread, will contain environmental and position information that represents a deployed battlefield sensor node. The environmental information will be “observed” by the sensor object, periodically resulting in the sensor object attempting to securely communicate information to other nodes. The position information of sender and receiver sensor nodes will be used to simulate each communications link by creating a probability that the receiver node will receive each message. When the simulation is executed, each sensor object will probabilistically observe its environment, send and receive data. This simulation will allow various scenarios and methodologies to be executed. The results of these simulations will provide data concerning the relative merits of different mechanisms.

Although not offered in this proposal, successful prototyping and simulation efforts could lead to a more sophisticated demonstration where the communications security architecture is implemented in a testbed containing multiple sensor simulations executing on candidate sensor microprocessors. Each simulated sensor could contain the candidate microprocessor, volatile memory, and non-volatile memory chips. Hardware co-processors may also be included within the sensor simulation.

Ultimately, prototyping efforts could make use of the low-cost prototyping kits solicited in the SenseIT BAA. If these kits are not available in adequate time, sensor simulations could initially be prototyped on breadboards and wire-wrap boards; eventually resulting in fabrication of prototype sensor board circuitry on printed circuit boards after hardware debugging is complete. Simulation of the sensor environment and communications between simulation sensors may be managed via a controlling personal computer. These hardware-intensive alternatives would provide a more realistic timing and power consumption picture of the sensor environment than a pure software demonstration.

The prototype implementation and demonstration experience will be reflected in the final specification of the communications security architecture and underlying cryptographic mechanisms, which will include revisions to the initial specification. The analysis of the communications security architecture and underlying cryptographic mechanisms via the prototype implementation and demonstration will be incorporated into a final report.

C. Deliverable Products

The deliverables of this project will be composed of the following reports, prototype software implementation, and demonstration of the communications security architecture and underlying cryptographic mechanisms for distributed resource-limited sensor networks:

- **Study of Distributed Sensor Network Requirements and Constraints**

Report on the unique communications environment and requirements of distributed sensor networks, the resource limitations of sensor nodes, and the security and security support requirements of distributed sensor networks. (REPORT; MONTH 6)

- **Preliminary Design and Specification of the Communications Security Architecture and Underlying Cryptographic Mechanisms for Distributed Sensor Networks**

Preliminary specification of the communications security architecture and underlying cryptographic mechanisms for distributed sensor networks that provide security and security support services needed for protected communication of control, management, and application data. (REPORT; MONTH 18)

- **Prototype Software Toolkit Implementation of the Communications Security Architecture and Underlying Cryptographic Mechanisms for Distributed Sensor Networks**

Prototype software implementation of the communications security architecture, consisting of software modules that provide security and security support services using the selected cryptographic mechanisms, key management schemes, and public key infrastructure functionality. (SOFTWARE; MONTH 24)

- **Demonstration of the Communications Security Architecture and Underlying Cryptographic Mechanisms for Distributed Sensor Networks**

Demonstration to DARPA of the prototype software implementation integrated into a simulated distributed sensor network. (DEMO; MONTH 28)

- **Final Design and Specification of the Communications Security Architecture and Underlying Cryptographic Mechanisms for Distributed Sensor Networks**

Final specification of the communications security architecture and underlying cryptographic mechanisms for distributed sensor networks incorporating revisions based on the prototype development and demonstration experience. (REPORT; MONTH 28)

- **Final Report on the Communications Security Architecture and Underlying Cryptographic Mechanisms for Distributed Sensor Networks**

Final report summarizing project goals, objectives, scope, approach, specifications, and software development and demonstration activities, assessing the communications security architecture and underlying cryptographic mechanisms for effectiveness in satisfying the requirements and constraints of resource-limited distributed sensor networks. (REPORT; MONTH 30)

Proprietary Claims

DARPA-funded research has maximum impact if it is widely available, both as research results and incorporated into commercial products. For each of the areas of proposed research that is not based on proprietary technology, TIS Labs will provide prototype software to Government organizations, research groups, and DARPA-designated industry working groups. TIS Labs is interested, nevertheless, in acquiring exclusive commercial rights for further development of some aspects of the technology proposed herein. Our technology transfer plan, described in Section F, is intended to support DARPA and the research community's interests and lead to commercialization of important technologies.

D. Statement of Work

D.1 Objectives

This research has three primary objectives: to identify practical cryptographic mechanisms and protocols that can be selectively employed by resource-limited sensor nodes; to design a communications security architecture suitable for use by distributed networks of resource-limited sensor nodes; and to implement a prototype system and simulation that can be used to demonstrate efficient and practical communications security for distributed sensor networks in a variety of environments and scenarios.

D.2 Scope

The scope of the contract includes: investigating the requirements of distributed sensor networks; exploring the feasibility of applying ultra-efficient cryptographic mechanisms to resource-limited sensor nodes; designing a communications security architecture that efficiently combines cryptographic mechanisms with varying performance and computational characteristics; developing a prototype implementation of the architecture and simulation of a distributed sensor network environment; and demonstrating the effectiveness of the architecture and cryptographic mechanisms in satisfying the requirements of distributed networks of resource-limited sensor nodes in a variety of deployment scenarios.

The primary problems to be addressed include: accurately modeling the resources that will likely be available to proposed SenseIT sensor networks; selecting appropriate cryptographic mechanisms and protocols to be used as components in a computationally-efficient communications security architecture; mapping candidate mechanisms to appropriate layers of the communications security architecture; and optimizing tradeoffs between security, functionality, implementation complexity, and computational efficiency.

D.3 Tasks/Technical Requirements

Over the course of this contract, the contractor shall: (1) study sensor environment, communications, security requirements and constraints, and develop an appropriate communications security architecture comprised of selected cryptographic mechanisms; and (2) develop, demonstrate, and analyze a prototype software implementation of the architecture to assess its ability to provide essential security services for protecting distributed sensor networks despite resource limitations.

The effort shall be composed of two concurrent tasks:

Task 1: Architecture and Mechanisms Study and Specification

The contractor shall study the unique communications environment and requirements of distributed sensor networks; identify the resource limitations of sensor nodes; and determine the security and security support requirements for the distributed sensor network. The contractor

shall design and specify a comprehensive communications security architecture for distributed sensor networks that: provides security and security support services needed for protected communication of control, management, and application data; allows selection of cryptographic mechanisms based on tradeoffs between security, functionality, implementation complexity, and computational efficiency; and incorporates efficient sensor network key management schemes for generating and distributing appropriate cryptographic parameters. The contractor shall research appropriate cryptographic mechanisms including: the efficient use of public key cryptography; the use of secret key (symmetric) cryptography in a manner that efficiently emulates public key functionality; the use of special-purpose cryptographic hardware accelerators; and efficient key management techniques, including ripple-keyed cryptography, and adaptive group keying methods. The contractor shall also address the control interface for managing keys and identifiers using limited memory, low power, and without interactive human input (e.g., without passphrases).

Deliverables:

- Study of Distributed Sensor Network Requirements and Constraints (REPORT; MONTH 06)
- Preliminary Design and Specification of the Communications Security Architecture and Underlying Cryptographic Mechanisms for Distributed Sensor Networks (REPORT; MONTH 18)
- Final Design and Specification of the Communications Security Architecture and Underlying Cryptographic Mechanisms for Distributed Sensor Networks (REPORT; MONTH 28)

Task 2: Prototype Development, Demonstration, and Analysis

The contractor shall: develop a prototype implementation of the communications security architecture, consisting of software modules that provide security and security support services using the selected cryptographic mechanisms and key management and public key infrastructure functionality; develop a simulation of a distributed sensor network environment; demonstrate the prototype implementation within the simulated distributed sensor network; and analyze the effectiveness of the architecture and underlying cryptographic mechanisms in satisfying the requirements and constraints of distributed sensor networks. The contractor shall prepare a final report that: summarizes project goals, objectives, scope, approach, specifications, and software development and demonstration activities; assesses the communications security architecture and underlying cryptographic mechanisms for effectiveness in satisfying the requirements of distributed networks of resource-limited sensor nodes; and discusses the lessons learned from developing and demonstrating the prototype implementation.

Deliverables:

- Prototype Software Toolkit Implementation of the Communications Security Architecture and Underlying Cryptographic Mechanisms for Distributed Sensor Networks (SOFTWARE; MONTH 24)
- Demonstration of the Communications Security Architecture and Underlying Cryptographic Mechanisms for Distributed Sensor Networks (DEMO; MONTH 28)
- Final Report on the Communications Security Architecture and Underlying Cryptographic Mechanisms for Distributed Sensor Networks (REPORT; MONTH 30)

E. Schedule of Milestones

Over a period of thirty months, the contractor will deliver the following:

Task 1: Architecture and Mechanisms Study and Specification

- Study of Distributed Sensor Network Requirements and Constraints (REPORT) MONTH 06
- Preliminary Design and Specification of the Communications Security Architecture and Underlying Cryptographic Mechanisms for Distributed Sensor Networks (REPORT) MONTH 18
- Final Design and Specification of the Communications Security Architecture and Underlying Cryptographic Mechanisms for Distributed Sensor Networks (REPORT) MONTH 28

Task 2: Prototype Development, Demonstration, and Analysis

- Prototype Software Toolkit Implementation of the Communications Security Architecture and Underlying Cryptographic Mechanisms for Distributed Sensor Networks (SOFTWARE) MONTH 24
- Demonstration of the Communications Security Architecture and Underlying Cryptographic Mechanisms for Distributed Sensor Networks (DEMO) MONTH 28
- Final Report on the Communications Security Architecture and Underlying Cryptographic Mechanisms for Distributed Sensor Networks (REPORT) MONTH 30

F. Technology Transfer

F.1 Results/Products

The proposed effort will provide DARPA and the SenseIT research communities with a comprehensive communications security architecture, and a set of efficient cryptographic mechanisms suitable for distributed networks of resource-limited sensors. The architecture and mechanisms will be provided in technical specifications, a prototype software implementation, and a demonstration.

The full set of results and products produced by the proposed effort will include:

- Analysis of the communications security environment and requirements found in existing sensor platforms and networks of resource-limited sensor nodes, including recommendations for further sensor platform and network development.
- Design and specification of efficient cryptographic mechanisms suitable for distributed networks of resource-limited sensor nodes, including computationally efficient public key cryptography, methods for emulating public key cryptography, hardware-acceleration, and efficient key management.
- Analysis of tradeoffs among security, functionality, implementation complexity, and computational efficiency of cryptographic mechanisms suitable for distributed networks of resource-limited sensor nodes.
- Design and specification of comprehensive communications security architecture suitable for distributed networks of resource-limited sensors. The architecture will specify mappings among cryptographic mechanisms, security services, and communications layers. The architecture will also specify supporting key management mechanisms and infrastructure components.
- Prototype software implementation and demonstration of the communications security architecture and underlying cryptographic mechanisms.

F.2 Technology Transfer

Our technology transfer plan for the work described above is modeled after the approach we have taken with other DARPA-sponsored TIS Labs projects, including the Firewall Toolkit (fwtk), the DNS Security (DNSSEC) reference implementation, Privacy Enhanced Mail (PEM) and MIME Object Security Services (MOSS) reference implementations, and our work on the Domain and Type Enforcement (DTE) system and Generic Software Wrappers. We will distribute all reports in Microsoft Word, Latex, PostScript, and/or PDF format. We will produce the prototype software implementation using Microsoft Windows and/or a commonly available, POSIX-compliant platform to facilitate widespread use. To the extent that the prototype components are not based on proprietary technology or products and are compliant with export control of cryptography, we will make associated source code and linkable object code available via the Internet. (The fwtk has been retrieved over 50,000 times in this manner). We will also

provide limited documentation of the system to describe configuration and use of the software. TIS Labs will make the prototype software and documentation available for non-commercial purposes.

Making the reports, software, and documentation available in this fashion will provide other SenseIT research teams with an available experimental toolkit to examine, experiment with, use and modify as they choose. Other research teams will benefit by access to a prototype of the communications security architecture and underlying cryptographic mechanisms. TIS Labs will benefit by receipt of valuable feedback from the community on functionality and efficiency. This method of distribution to the community supports DARPA's need to stimulate security research in the distributed sensor network area as well as technology transfer to the private sector.

Besides making the prototype available to the community, TIS Labs will undertake a variety of other technology transfer activities to make other research groups and organizations aware of our results and help evaluate their applicability. These activities will include the following:

- We will interact directly with other government research teams to ensure that their designs provide for, or at least do not hamper, the inclusion of our communications security architecture and underlying cryptographic mechanisms, and also that our architecture and mechanisms are adequate for the state-of-the-art in sensor research. By working closely with other contractors, we will be able to closely track related research and identify technology transfer opportunities in other researchers' results and products.
- We will publish our communications security architecture, cryptographic mechanisms, and other results for distributed sensor networks in hard-copy reports and electronically via the TIS Labs World Wide Web (WWW) page.
- We will participate in appropriate computer and network security conferences in order to solicit community feedback and foster community acceptance of our results.
- We will participate with other DARPA researchers in PI meetings and cooperative workshops. As a result, we will ensure the applicability of our results and ease the near-term transfer of our results into other research efforts.
- We will work closely with the TIS Labs Internal Research and Development group to transition technologies, as appropriate, especially efficient cryptographic mechanisms, into Network Associates' products, thus making them available as COTS to government customers. We will present research results at TIS Labs' semi-annual technology review to provide visibility to Network Associates' engineering and senior management.

Together, these results, products, and technology transfer activities will ensure that our research contributions are effectively channeled to the SenseIT research community and to government and industry, thereby maximizing the overall payoff and impact sought by DARPA.

G. Comparison with Ongoing Research

There are several recent and ongoing research projects related to distributed sensor networks:

The existing military **REMBASS, IREMBASS, and REMBASS2** systems are good examples of the current state-of-the-art in remote battlefield sensor technology. However, these systems employ sensors designed for limited, manual, perimeter-based deployment, and require a central monitoring system. Another good example of current sensor technology is **System Innovations, Inc.'s OmniSense**, a remote intrusion sensor system that incorporates a number of sensor units along with field processing and display units. Like the REMBASS systems, OmniSense is designed for limited, manual deployment, and requires a central monitoring system. Both the REMBASS systems and OmniSense generally do not employ cryptographic mechanisms for protection of sensitive communications, rather they rely on physical protection (e.g., a wire between two communicating devices), or electronic protection that hides the fact of communication (e.g., spread spectrum). The SenseIT program will explore new revolutionary systems that support complex communications and sophisticated processing among 10 to 10,000 sensors/nodes, and have the goal of being easily deployed, of being self-organizing, and of allowing decentralized control. Automated means of protecting sensitive communications will be essential in these new systems.

The **DARPA/ITO Global Mobile (GloMo) Information Technology** program is developing and integrating technologies and techniques at applications, networking, and wireless link levels to enable a full range of wireless communications capabilities. Within GloMo, SRI International's Advanced Secure Wireless Integrated Networks (ASWIN) project is implementing cryptographic techniques, including multicast key management, to secure IP header routing information to prevent traffic analysis in a wireless network. Similarly, the Internet Engineering Task Force (IETF) Mobile IP Working Group is developing architectures to support basic mobility, including security, within the Internet. The SenseIT program involves a special case of wireless communications that can borrow from the GloMo program, SRI International's ASWIN project, and the IETF Mobile IP WG, but the SenseIT program also has its own unique requirements for data distribution and aggregation, network survivability, robustness, and self-configuration.

At the **Berkeley Sensor and Actuator Center at the University of California at Berkeley** researchers are examining low-power communications logic. To progress the feasibility of their **SmartDust** sensor concept, Brett Warneke and Carl Chang have designed and demonstrated communications logic for a SmartDust sensor that draws only nanowatts worth of power.

ISO 7498/2, Basic Reference Model – Part 2: Security Architecture, specifies a standard security architecture for use with the basic reference model for network communications. The security architecture specifies the relationship between security services and mechanisms, and the placement of security services and mechanisms at the layers of the basic reference model. Our communications security architecture for distributed sensor networks will follow a similar approach as the ISO 7498/2, with mappings between cryptographic mechanisms, security services, and communication layers in the distributed sensor network.

In the paper "**Estimating the Computational Requirements of a Software GSM Base Station**" by **Thierry Turletti and David Tennenhouse**, the authors examine software performance on GSM (the Global System for Mobile communications) base stations. The paper estimates computational performance for many of the communications and cryptographic functions that sensor CPUs may perform, including convolutional coding, interleaving, and ciphering. We will explore whether this model and the paper's results for estimating computational performance of GSM base stations are applicable to SenseIT sensor nodes.

TIS Labs currently has another proposal submitted to DARPA in response to BAA 99-16. **Sensor Net Execution Environment for Cooperating Hardware (SNEECH)** proposes to develop a prototype execution environment designed to improve power efficiency when the sensor network node interacts with the external world. SNEECH also proposes to develop and implement a node security architecture that focuses on restricting acceptance of network traffic to authenticated users, controlling access to, and sharing of, resources including power, and providing the task isolation required by multi-tasking. The SNEECH node security architecture and the communications security architecture proposed here will build upon and complement each other.