Integrating End-to-End Encryption and Authentication Technology into Broadband Networks

Lyndon G. Pierson
Sandia National Laboratories
Albuquerque, NM

[1]BISDN services will involve the integration of high speed data, voice, and video functionality delivered via technology similar to Asynchronous Transfer Mode (ATM) switching and SONET[2] optical transmission systems. Customers of BISDN[3] services may need a variety of data authenticity and privacy assurances. via Asynchronous Transfer Mode (ATM) services Cryptographic methods can be used to assure authenticity and privacy, but are hard to scale for implementation at high speed. The incorporation of these methods into computer networks can severely impact functionality, reliability, and performance.

To improve functionality of high speed computer networks, a wide interoperability of applications, network hardware, and network software is desired. This is facilitated by adherence to standards and extending those standards as appropriate. The signalling required to establish customer selected security assurances for traffic in these networks should conform to the evolving ATM and SONET standards.

High end-to-end reliability (availability) requires even higher reliability of intermediate components and automatic fail-over to redundant components where feasible. Cryptosystems which are improperly keyed or which have lost synchronization can ruin the availability of an otherwise reliable communication system. Therefore, key management and synchronization must be carefully implemented so that the availability of encrypted services is as high as the availability of unencrypted services.

End-to-end throughput performance depends on the communication processing rates and signalling rates of network components and also on the delay and error rate to which the traffic is subjected. Clearly, encryption/decryption processes should subject network traffic to as little additional delay and error rate as possible.

Data authenticity assurance will likely be based on public key authentication methods negotiated during the signalling for setup of virtual circuits. Methods for authenticating called and calling party as well as called and calling station have been proposed.[4 5 6]

Data privacy assurance will likely be based on a hybrid encryption method in which symmetric session keys are negotiated via a public key method during or at the time of signalling for setup of virtual circuits.

While there are many design issues associated with the serving of public keys for authenticated signalling and for establishment of session cryptovariables, this paper is concerned with the impact of encryption itself on such communications once the signalling and setup have been completed.

1. This work was supported by the United States Department of Energy under contract DE-AC04-94AL85000
2. Synchronous Optical Network
3. Broadband Integrated Services Data Network

4. L. G. Pierson and Tom D. Tarman. Requirements for security signalling. ATM Forum/95-0137, Denver, Co., April 1995. ATM Forum Technical Committee Meeting.
5. "A Proposed Generic Authentication Information Element", ATM Forum/95-0460
6. "Proposed DSS-Specific Fields for the Generic Authentication Information Element", ATM Forum/95-0461

DISCLAIMER

Portions of this document may be illegible electronic image products. Images are produced from the best available original document.

Network security protections should be carefully matched to the threats against which protection is desired. Even after eliminating unnecessary protections, the remaining customer-required network security protections can impose severe performance penalties. These penalties (further discussed below) usually involve increased communication processing for authentication or encryption, increased error rate, increased communication delay, and decreased reliability/ availability. Protection measures involving encryption should be carefully engineered so as to impose the least performance, reliability, and functionality penalties, while achieving the required security protection.

To study these trade-offs, a prototype encryptor/ decryptor was developed. This effort demonstrated the viability of implementing certain encryption techniques in high speed networks. The research prototype processes ATM cells in a SONET OC-3 payload. This paper describes the functionality, reliability, security, and performance design trade-offs investigated with the prototype.

End-to-End Encryption vs. Link Encryption

Link encryptors typically encrypt each and every bit of a synchronous communication line at one end of a leased or private circuit, and decrypt each and every bit at the other end. End-to-end encryption can be thought of as occurring at a higher layer of communication protocol, and involves identifying and passing the control information associated with each data packet, while encrypting the payloads of selected data packets. Since the control information is not encrypted, this allows the processing of such packets at intermediate equipment without decryption. End-to-end encryption does not protect against traffic analysis of header information [1]. Since many secure applications do not require protection from traffic analysis, the promise of less equipment and correspondingly smaller key management difficulty makes end-to-end encryption more attractive than link encryption. In ATM switched virtual circuits, encryption must take the form of end-to-end encryption unless the intermediate ATM switches are physically secured and link encryptor/decryptor pairs are to be installed on each segment of the path along the switched virtual circuit.

The encryption and decryption processes are usually simpler (and faster) for link encryption than for end-to-end encryption. Link encryptors have fewer decisions to make, and need not identify the beginning, end, or control information of each data packet.

The encryption and decryption processes are intended to be computationally infeasible unless one holds the cryptographic "key". This usually causes the computations to become intensive even on behalf of the authorized "key" holder.

Encryption is computationally intensive, and end-to-end encryption is not only computationally intensive, but also communication processing intensive. This tends to cause the commercial availability of high speed link encryption and end-to-end encryption equipment to lag the availability of high speed communication equipment and service offerings.

High speed communication equipment and service offerings become available as higher speed switching components (faster transistors) become available. In order to keep up with data rates available through high speed communication equipment, encryptors typically require a great many of the more expensive, faster switching components. Since the consumer market for encryption has been relatively small, faster encryptors typically become available only when faster transistors become "affordable".

This research is applying parallel processing techniques to scale the speed of encryption processing without requiring the availability of higher speed components. This approach may still require higher speed components to perform parallel-

to-serial and serial-to-parallel conversions for transmission, but will require fewer higher speed components throughout the encryption processing function.

Security

A "dictionary lookup" crypto-analytic attack involves matching the cyphertext of a message against recurring identical cyphertext messages (thereby indicating a subsequent identical plaintext was transmitted). A "playback" attack involves "re-playing" a given cyphertext in order to spoof the repetition of a message. To prevent "dictionary lookup" and/or "playback" attacks, cryptosystems employ various feedback methods. The feedback causes repeated plaintext messages to encrypt into different cyphertexts. Implementation of these feedback methods complicate the scaling of the encryption/decryption process. Functionality

End-to-end encryption can provide greater granularity of protection by encrypting simultaneous communication streams with separate sets of cryptovariables. This requires hardware which can quickly switch "cryptovariable contexts" to encrypt cells from multiple concurrent virtual circuits.

For some security purposes, the closer encryption is implemented to the user's application (in terms of communication protocol layers), the better. Other security purposes may demand encryption and/or authentication at other protocol layers. This may result in super-encryption of higher layer encrypted packets by encryption also performed at lower layers. Such "super-encryption" makes cryptographic synchronization loss detection and recovery more difficult for the lower layer encryption processes.

One reason that encryption has been implemented at the lower communication layers is that such communication equipment can be "pooled" in a single "crypto room" which is physically secured and staffed by trained personnel who key and operate the encryptors. When encryption is moved to higher protocol layers and implemented in the hardware or software of the user's computer, the trained crypto custodian is no longer present. As encryption functions move out of the "crypto room" into the workstation, new methods of providing assurance to the user that the encryption is keyed and operating properly will be required.

Scalability and Interoperability

Prior to the availability of "variable bit rate" communication services, encryption/decryption processes at both ends of the network were required to operate at the same rate. Link encryptors, encrypting each and every bit of a synchronous communication line, have to decrypt each and every bit at the same rate as encrypted. End-to-end encryptors likewise matched a common media clock rate. With the advent of "variable bit rate" communication services via ATM switchgear, a requirement now emerges for implementations which can inter-communicate, yet operate at different data rates. For example, ATM cells encrypted for transmission into an ATM network via an OC-3 (0.155 Gb/s) or even OC-48 (2.4 Gb/s) interface may be decrypted at a DS3 (0.045 Gb/s) interface. The lower speed, lower cost encryptor/ decryptor will implement internal encryption/ decryption processes at a lower data rate, likely with a lesser degree of "parallelism".

This requirement, for encryption/decryption processes which can be scaled for high speed, yet interoperate with unscaled or lesser scaled versions, is difficult to achieve in most cryptosystems which use feedback methods to protect against dictionary lookup and replay attacks.

For example, the "output feedback" (OFB), "cipher feedback" (CFB) and "cipher block chaining" (CBC) modes[1] defined for the Data Encryption Standard (DES)[2] can all be scaled and interoperated with a

1. American National Standard for Information Processing Systems- Data Encryption Algorithm-Modes of Operation," ANSI X3.106-1883, American National Standards Institute, New York.

similarly scaled implementation by processing multiple 64-bit blocks of data in parallel. However, these scaled modes are difficult or impossible to interoperate with an unscaled or lesser scaled implementation. This is because the lesser scaled implementation will require cyphertext or plaintext which has not yet been computed, or will require access to previously processed data. The buffering of the previously processed data will be expensive in terms of circuitry required and will increase the cryptographic processing time. The increased processing time will subject the encrypted/decrypted traffic to additional delay, thereby reducing end-to-end throughput.

Reliability

Experience shows that encrypted communications suffer lower reliability than unencrypted communications. One reason is that encryption prevents the use of communication diagnostic techniques which involve data monitoring and/or injecting signals across the encryption equipment.

Monitoring cyphertext (examining for correctness) and injecting cyphertext to be decrypted cannot be done without another properly keyed encryption unit in the diagnostic equipment. "Loopback" methods are the primary tools available to the diagnostician of encrypted links.

These loopback methods, although powerful when used properly, suffer from certain drawbacks. Placing a communication link in loopback usually changes the method of clocking data at the loopback point. If the communication fault involves an improper clocking mode, a circuit can operate properly in loopback but improperly when "normal". Another "pathological" condition not detectable with loopback methods is that of a data inversion in both transmit and receive channels at a single point between encryptors and decryptors. Again, in loopback, the circuit operates properly (through both inversions) but fails in normal non-loopback mode (through a single inversion). Similarly, loopback methods will show mis-keyed encryptor/decryptors to operate properly in loopback mode (since the encryption and decryption processes at a single end will be similarly keyed, successfully processing the looped-back data), but fail to operate in the normal end-to-end configuration.

Another reason for the increased unreliability of encrypted communication compared to non-encrypted communication is simply the logistics of securely getting encryptor/decryptor pairs keyed with the same key. Murphy's law applies to encryptors: the probability of mismatched keys is great. To overcome this difficulty, asymmetric (public key) key management systems can be employed, but with a corresponding increase in signalling complexity and overhead.

Once properly keyed, cryptosystems which employ feedback to deter dictionary lookup and playback attacks must be synchronized in order to encrypt and decrypt properly. Once synchronized, these cryptosystems remain synchronized until/unless bits are inserted into or deleted from the cyphertext to be decrypted. When this occurs, more or fewer bits are received for decryption than were encrypted. This causes the decryption process to become "out of step" with the encryption process. These "bit count integrity loss" errors, unlike errors which change ones to zeros or vice versa, cause all subsequent plaintext to be improperly decrypted until re-synchronization occurs. Except for a special class of "self-synchronizing" cryptosystems, the communication reliability is a function of how fast and how accurately this "crypto sync loss" state can be detected and recovery achieved.

Most crypto sync loss detection methods involve identifying expected patterns in the decrypted data. An extension of this method involves the insertion of patterns specifically for the purpose of sync loss detection. Such insertion introduces "data

2. "American National Standard for Data Encryption Algorithm (DEA)" ANSI X3.92-1981, American National Standards Institute, New York.

expansion" which is bandwidth-inefficient and difficult to scale for long patterns.

These methods work well for low data rate encrypted communications. Because unsynchronized, improperly decrypted data resembles cyphertext (random data), these "pattern matching" methods do not scale well for high data rate communications. The frequency of occurrence of any given pattern in the random unsynchronized output increases with data rate. These detectors become unreliable at high speed because the patterns expected in properly decrypted data are quickly found also in the unsynchronized, improperly decrypted data.

A Sandia developed method (U. S. Patent # 4977596) which measures the randomness of the decrypted data overcomes this problem and scales well for implementation at high data rates.

Once the synchronization loss has been detected at the decryptor, a resynchronization request must be communicated to the encryptor. The encryptor must then send sufficient information to the decryptor to re-establish cryptographic synchronization.

Cryptographic synchronization recovery for these systems requires a period of time to detect sync loss and a round trip delay to request and receive the synchronization information. During this time, encrypted communication bandwidth is unavailable to the user.

While synchronization recovery is taking place, some method of preventing the delivery of "random" improperly decrypted data to end equipment or processes should be implemented. If not, these processes (expecting plaintext) may interpret patterns in the random data as proper control information. This may cause these end processes to transition to unexpected states, so that data processing cannot continue when communication is restored.

Performance

To achieve high performance, the end-to-end delay and error rate must be minimized. This is because "reliable" transport protocols (such as TCP) must "stop and wait" after transmitting some maximum amount of yet-to-be acknowledged data. This can be thought of as a limited ability to fill a large, long "pipe" between the transmitter and receiver. As the delay-bandwidth product increases, the diameter and/or the length of the pipe increases, producing a larger volume to be filled by the transmitter (with data yet to be acknowledged by the receiver).

In addition, when an error is detected at the receiver, the transmitter is instructed to restart transmission of all data after the error. Sometimes this is called a "go-back-N protocol) This has the effect of discarding all the data presently filling the pipe between transmitter and receiver and then re-filling the pipe. Implementation of "selective acknowledgment" of data segments received after detection of an error can help. Most transport protocols (including most implementations of TCP) do not implement the "selective acknowledgment" of data segments received after detection of an error. Selective acknowledgment can reduce (but not eliminate) the sensitivity of throughput to the delay-bandwidth product and error rate.

Some encryption/decryption processes can add significant network delay to that experienced by unencrypted traffic. Some encryption/decryption processes decrypt single bit errors in the cyphertext into multiple bit errors in the decrypted plaintext, "magnifying" the error rate to which the cyphertext is subjected.

As network data transfer rates increase, communication becomes less and less efficient due to the increased delay-bandwidth product associated with the end-to-end transactions. This delay-bandwidth product can be thought of as the number of bits in transit which have left the transmitter and have not yet been received by the distant receiver. As this amount of "data in transit" increases, so do

difficulties in implementing efficient error control and flow control. The performance of high speed communications systems will be increasingly sensitive to delay and error characteristics.

One of the reasons Asynchronous Transfer Mode (ATM) switching technology is expected to scale to support high speed networks is the low delay incurred as traffic passes through each switching node. Short cell headers and fixed length cells enable ATM switching implementations which achieve switching delays limited only by the time required to assemble the five byte cell header.

Clearly, implementors of encrypted services should strive to minimize additional network delay due to encryption, and to eliminate error magnification by careful choice of encryption method.

Summary of Requirements

For encryption/decryption in the high speed "variable bit rate" ATM networking environment, these characteristics are highly desirable:

1. Encryption of multiple identical plaintext messages into differing cyphertexts (to deter dictionary lookup and playback attacks).
2. Scalability of encryption speed (independently of speed of switching elements)
3. Interoperability of scaled and lesser scaled implementations.
4. Little or no error rate magnification.
5. Minimum traffic delay due to encryption.
6. Scalable key management.
7. Fast context switching of cryptovariable context between cell streams (less than header processing time).
8. Fast detection and recovery from cryptographic sync loss.

Research Prototype

A "research prototype" encryptor/decryptor has been developed. This prototype (not a product) is intended to demonstrate the viability of achieving these objectives by processing ATM cells in a SONET OC-3 payload.

A "Filter Generator" was chosen for implementation in the Sandia Research Prototype. Linear Feedback Shift Registers (LFSR) produce Linear Recurring Sequences (LRS) with long periods which have good "pseudorandom" properties. LFSRs are also easily scaled to generate multiple bits of the sequence in parallel. However, crypto-analytic methods exist to utilize the linear predictability of LRS to mount a cyphertext-plaintext attack against a purely linear sequence used as a keystream[3]. Several variations of periodic sequence key stream generators exist which deter such cryptanalysis. A filter generator uses a non-linear function to mask the linearity of a long linear recurring sequence generator.

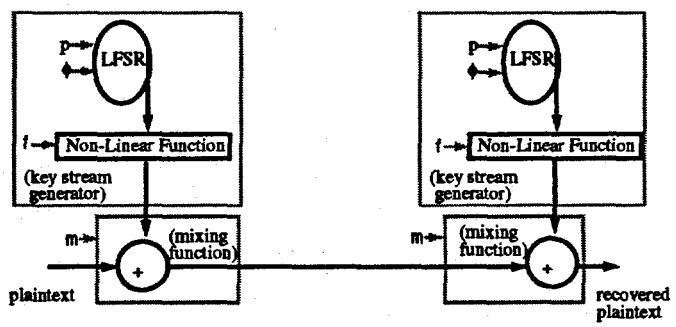

Figure 1: Encryption with key stream produced by masking linearity of a linear recurring sequence

This design involves no "feedback" around a non-linear function in order to both scale and interoperate with implementations of other scale factors. It also provides no error magnification. Single bit errors in cyphertext result in single bit decrypted plaintext errors. If the linear recurring sequence is sufficiently long, subsequent identical plaintexts are encrypted into different cyphertexts. This deters the dictionary lookup and playback attacks previously mentioned.

The prototype implementation encrypts and decrypts 8 bits at a time (scale factor of 8). This provides direct compatibility with the ATM/SONET framer used.

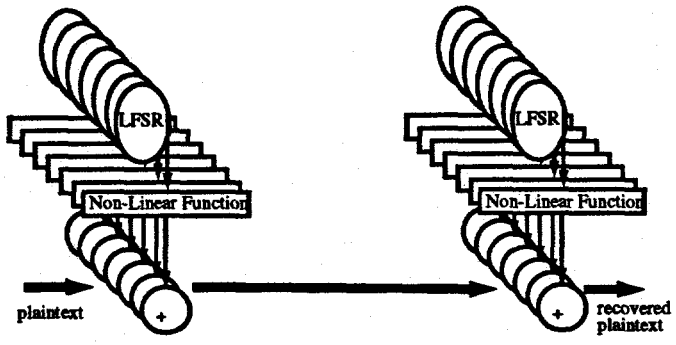

Figure 2: Parallel Encryption with key stream produced by masking linearity of a linear recurring sequence (bold arrows depict parallel transfer of data)

The linear and non-linear functions were designed to add minimum traffic delay. The traffic delay through the prototype was measured to be about 2.7 microseconds. This delay is due to one clock period required to encrypt each byte plus the time required by the ATM/SONET framer to assemble one cell.

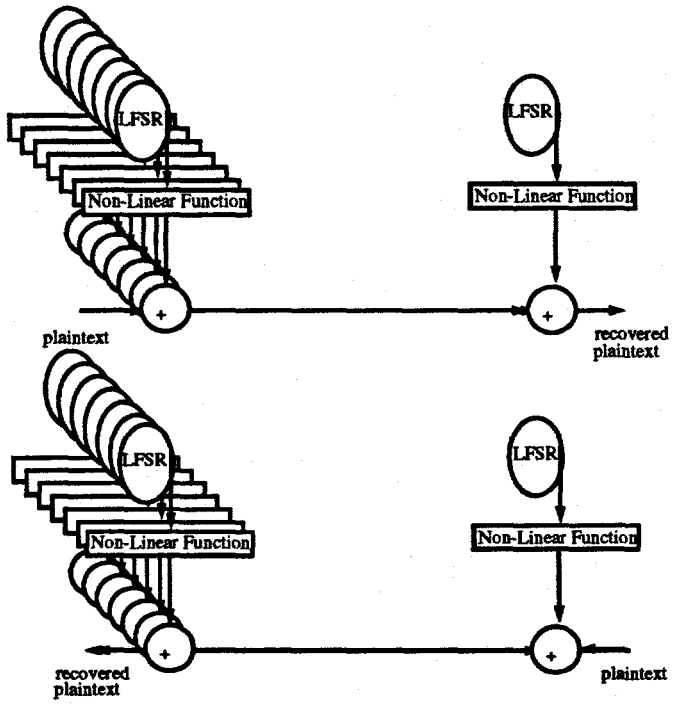

Figure 3: Interoperation of Scaled and Unscaled Encryptors/ Decryptors with key stream produced by masking linearity of a linear recurring sequence

A prototype which encrypts and decrypts 8 bits at a time was interoperated with an implementation which processed 32 bits at a time. Although this particular combination of scale factors was chosen for "proof of concept", implementations of any scale factor will also interoperate.

"Key-Agile" cryptovariable context switching is done on the basis of the VPI and VCI in each cell header. The prototype achieves a cryptovariable context switching time of 50 nanoseconds (one clock period). In order to rapidly demonstrate "proof-of-concept", only two cryptovariable contexts were implemented, and the prototype implements no "key management". Session keys are embedded in the prototype's Electrically Programmable Logic Devices (EPLDs).

Several crypto sync loss detection and recovery methods are under investigation. The initial testing has involved only the simplest of synchronization methods, 1) implicit re-synchronization of each cell payload to an initial state, and 2) synchronization upon demand. The synchronization upon demand is implemented by signalling between the encryptor and decryptor via "OAM" cells.

Methods to prevent delivery of cyphertext and unsynchronized decrypted plaintext to end equipment and application processes are under investigation.

In order to fully evaluate these concepts, encryption of communication traffic for real user applications are required. To accomplish this, a pair of the encryption prototypes were installed at a Lockheed Martin site in the California Research Education Network (CalREN). After initial testing of the prototypes, the CalREN experiment partners will collaborate to "stress" the encryptors with application data. The applications for which the impact of the encryptors will be evaluated involve distributed interactive simulation and concurrent collaborative design.

After correcting initial hardware problems with the prototypes, preliminary testing in the CalREN testbed has shown no measurable increase in end-to-end delay (measured via "ping" between workstations), no increased cell loss rate due to encryption

Conclusion:

The Sandia "research prototype" encryptor uses an encryption method known as a "Filter Generator" [3]. This "Filter Generator" design encrypts subsequent identical plaintexts into different cyphertexts in order to deter dictionary lookup and playback attacks. The "Filter Generator" design scales and inter-operates with unscaled implementations, and does not magnify the error rate to which the cyphertext is exposed. Delay of communication traffic through the encryptor has been measured to be 2.7 microseconds, essentially the time required for assembly of a single cell by the ATM/SONET cell framer. Cryptovariables were embedded in the programmable devices to speed implementation. The prototype encryptors demonstrate cryptovariable context switching between two contexts within 50 nanoseconds demonstrating the viability of the "Key Agility" concept.

Further work is planned in the areas of authentication of the called and calling party in the signalling for setup of ATM Switched Virtual Circuits (SVCs); key management via public key exchange of session cryptovariables; and synchronization methods.

Sandia's end-to-end ATM/SONET OC-3 encryptor prototype (not a product) is a vehicle for the study of the trade-offs between security, functionality, reliability, and performance.

Although the results of this research may be applied to other encryption schemes, very few encryption schemes have been found to "scale well" and still interoperate with unscaled implementations.

Initial testing of prototypes shows that speed scaling, interoperability, and delay objectives were met.

Sync loss detection/recovery/protection methods are yet under investigation.

Bibliography:

1. Padlipsky, M. A., et al., Limitations of End-to-End Encryption in Secure Computer Networks: (MTR-3592 VOL. 1), Bedford, Mass., Mitre Corporation, 1978.

2. Schneir, Bruce, Applied Cryptography, John Wiley & Sons, New York, 1994.

3. Simmons, Gustavus J. (Editor), Contemporary Cryptography - The Science of Information Integrity, IEEE Press, New York, 1992.