AUTHENTICATION OF THE PACKET RADIO SWITCH CONTROL LINK

Hal Feinstein, WB3KDU
Amateur Radio Research and Development
PO Drawer 6148
McLean, VA 22106-6148

Abstract

This paper discusses the design of a simple authentication method which is applied to a remotely sited packet radio switch. The control path to the packet radio switch is a very high frequency (VHF) radio channel which is easily monitored and accesed. Such ease of access requires that only authorized control stations be permitted to issue switch control and maintenance commands.

The authentication design discussed in this paper provides three functions: (a) positive identification of the switch and control operator, (b) safeguard message streams flowing between switch and control operator, and (c) rapid identification and rejection of false or manipulated messages.

While some aspects of the work are unique, many of the ideas we employed are disscused in the literature [Mayer,1982], [Needham,1978]. The work on challenge numbers discussed in section three was motivated by a description of World War II spoofing and IFF problems described in [Bethancourt,1979][Kahn,1976] while problems of message alteration are covered by various sources [ANSI,1985]. The slow dictionary attack discussed in section three was originally described by Blake Greenly of Citibank. Lastly, the area of jamming and imitative deception is covered in [Frick, 19451.

This paper is divided into four sections. The first section contains a brief overview of packet radio techniques. The second section discusses assumptions concerning the radio environment in which the packet radio switch operates. This envirnoment is characterized b unreliable radio path as well as occasionaI spoofing and malicious interference. In the third section we discuss the actions of each of the protocola four procedures and the make-up of its data construct. Section three also constains a discussion of the cryptographic considerations upon which this protocol is based. Lastly, in the appendix we describe an experimental one-way cipher based on a random program technique which we hope to incorporate into future versions of the radio packet switch.

1.0 Packet Radio Overview

Packet radio is the extension of packet switching to the radio media. The original experiments were performed by the University of Hawaii and the DoD Adavanced Research Projects Agency. Since that time packet radio networks are finding their way into satellite, police and commercial uses.

Packet radio is a technique for communicating digital information between stations which share a common line-of-sight radio channel. In this respect packet radio is similar to the local area network protocol ETHERNET; but, using radio as the transmission media instead of coaxial cable.

Communications between packet radio stations are governed by a number of design and protocol conventions which are organized into a seven layer model. Lower layers of the model deals with elementary aspects of communications such as how a station shares access to the radio channel or communicates directly with another station. Higher levels of the model detail common procedures for communicating within a network of stations.

A common packet radio station is composed of three components: a personel computer or terminal, a terminal node controller and a low power VHF transmitter/receiver (transceiver). The terminal node controller; which directs the actual packet radio operation; is commonly designed as a small, special purpose computer containing a microprocessor, protocol decoding hardware and various amounts 0f program and working memory.

Within a line-of-sight radio environment, these packet radio stations form a local network were direct station-to-station communications is possible; either directly or with the aid of unconnected packet repeaters. Typical populations range to a hundred or more stations in some large cities with a typical network diameters to about 50 miles.

Regional packet radio systems are linked by a second type of network which spans the line-of-sight user communities. The inter-city network consists of high speed radio links between local user communities linking packet radio switching stations. The packet radio switches provide a access to the local users and also serve as a reliable relay and routing control point for the backbone links.

A packet radio switch is an unmanned, automatic station which provides two functions: first, it provides local access to the high speed inter-city radio network. Second, it acts as a relay and switching point for the high speed inter-city backbone network. Several packet radio switches can provide service to users in a wide geographical area. In this way the inter-city links form a kind of "long distance service" while the local area can be viewed as a call within the same area code.

To control network operations each packet radio switch contains a specialized command link which permits local system operators control over the switches executive functions. The executive functions; which include the operator connection process; is an independent application process running within the switch at ISO layer (application layer) seven. The authentication protocol is designed to function as an adjunt to the operator application and hence, as a part of layer seven.

The switch is designed to reside in locations which are not readily accessible in order to take advantage of high buildings or mountain-top location, making frequent access by service personel difficult. For this reason the switches executive routines provides tools which are quite sophisticated allowing maximum control over the switch.

2. Authentication on the Command Link

Safeguarding the radio packet switch control link is required to prevent unnecessary interruption of service. The common methode of protecting such a link is to use authentication cryptography which offers some unique opportunities to employ various types of cryptographic ideas. The central problem which must be solved is authentication in an open and easily monitored radio channel.

In this section we examine four different assumptions concerning types of attacks against the command link: pervasive monitoring, deliberate interference, playback, and spoofing. These threat assumptions are combined with several

design considerations to yield nine system requirements. In the second part of this section we discuss how each threat is met by a protocol defense or counter-strategy.

2.1 Threat Assumptions and Requirements

The packet radio control link provides the system operator and system developer with a wide range of opertions and testing function. For these reasons it would be quite easy for a malicious operator to disrupt service or "crash" the switch by gaining control of the control operator functions. Short of this, a malicious station could spoof or jam the link, blocking all communications on the link path. Therefore, there are essentially three type of problems to consider: (a) false identification of the control operator or switch, (b) manipulation, insertion or deletion of valid control messages and (c) deliberate interference. We state these as formal assumption below:

1. (Monitoring) It is assumed that a malicious operator is always monitoring the control link and that he has complete knowledge of switch operations.
2. (Playback) It is assumed that the malicious operator can reliably playback valid control sessions or parts of valid control sessions without error.
3. (Deliberate Interference) It is assumed that the malicious; operator can jam the control channel at will either through continuos jamming or through selective "spot" jamming.
4. (Spoofing) It is assumed that the malicious operator can perform selective editing of valid messages.

From these four assumptions nine authentication design requirements were developed. In most cases these represent countermeasures to the attacks described above but some reflect design choices made on the part of the authors.

1. The protocol should gracefully shutdown under error. That is, the protocol should not block a control link by continuos cycling in an error retry or resynchronization loop. (Malicious Interference).
2. Only a valid user can initiate a control session.
3. The authentication technique must not expose key bits on the open air (Monitoring).
4. A unique session key must be used for each new session. (Playback)
5. Critical protocol information must be authenticated (Spoofing).
6. Control message content must be protected (Spoofing).
7. Consequtive control message must be protected. (Spoofing).
8. A control session and its messages must be uniquly identified. (Playback).
9. Control messages must be in plaintext (FCC Regulation).

2.2 Interference Safeguards

The first requirement deals with deliberate interference which forms the most common expected attack upon the switch control link. It is easiest to mount and if done cleverly, one of the hardest to protect against. For purposes of this discussion two types of jamming will be considered: steady jamming which simply blocks the communication channel for a length of time and momentary jamming, in which the jammer induces errors on a regular basis to effectivly block the link.

Steady jamming is essentially a game of strategy in which the jammer attempts to block communications by transmitting signals which the control link receiver cannot tell from the valid signal. To do this the jammer must either overwhelm the switches control link receiver by transmitting a very powerful signal or closly imitate the control link signal characteristics so that the control receiver cannot distinguish the jammer from the valid signal.

To overcome jamming the control link must use special error correction coding and vary some characteristic of his transmitted signal in a fashion which is known only to himself and the link receiver. Military systems commonly employ spread spectrum modulation to provide this unpredictable changing. Simply, spread spectrum consists of either changing the link radio channel many times a second under control of a psuedorandom generator or change the waveform of the transmitter many times a second. In both cases, the link receiver will know what frequency or waveform is being used and can reject other signals not coded in the expected fashion.

For the jammer to be sucessful it must imitate the link as closly as possible; however, the signal is changing so rapidly, and in an unpredictable fashion, that the jammers chances of sucess fall as the links speed of change increases.

In the case of momentary jamming, the jammers strategy is to block communications by attacking the link protocol error detection mechanism. The jammers signal is meant to cause the protocol to perform error retries and hence clog the channel with retransmission. Cleverly applied, this form of jamming need only transmit for a fraction of a second making location by radio direction finding difficult.

Most links subject to deliberate jamming are specially coded with an effect-ive error correcting code (foward error correction) before transmission. The error correcting code can identify and correct many small single bit and burst errors. Commonly, bit rearrangment (bit interleaving) within a data block is also used to combat longer burst errors.

The AX.25 protocol upon which the link is based uses an error detection and retransmission strategy and is vulnerable to both momentary and steady jamming attacks. Foward error correction and spread spectrum hardware are still relativly expensive; hence, a different counter-strategy was needed.

The counter-strategy developed for the switch rests on two observations. First, the switch is designed to always reset and continue automatic packet switching operations in the event of an error or failure. Hence, if jamming interrupts a control operation and denies access over some time period, the switch will resume normal automatic operations.

The second observation concerns safe and unsafe states. Briefly, a safe state is one in which the switch can operate normally and is not in danger of erroneous operations. Comparatively, an unsafe state puts the switch in danger of crashing, perhaps through a temporary instruction patch or experimental manipulation of parameters.

Commands issued by the control operator affect the state of the-switch. These operator commands are issued individually or as part of a group of commands. Moreover, each intended operator action ends with the switch in a safe state, while commands within a sequence of operator commands may place the switch temporarily in an unsafe state.

The most dangerous time for the jammer to become active is when the switch is in an unsafe state. We assume that the jammer is monitoring the activity of the control link and can determine when this state occurs. To close the window of vulnerability the switches executive software sets a hardware timer whenever the switch enters an unsafe state. If the link should fail for any reason this timer will expire and trigger a hardware reset returning the switch to normal packet switching operations.

We have choosen not to employ special coding and spread spectrum hardware primarily as a trade off between the effects of jamming, jamming frequency and hardware expense. Jamming which occurs for a period of one second or more can be easily located with current., commercially available automatic radio direction finding techniques and specific legal remedies can be applied. Jamming attacks; while distruptive; are generally not too frequent; however, in certain specific areas of the country jamming is more prevalent and it may be necessary to employ some of the anti-jam hardware mentioned.

2.3 Imitative Deception and Spoofing

Imitative deception and its variations form the basis for several attacks. The ones considered here are attacks whose strategy is to imitate a vaild user action by manufacturing valid looking messages, altering selected parts of a valid message (spoofing) or by recording and replaying whole or parts of a previous session (playback attacks).

The strategy of imitative deception is to inject a valid appearing messages into a link which is taken for valid traffic. One of the primary [Frick,1945] instances of this type of attack occured in the opening days of WW I on the German-Russian front durring the battle of Tannenburg. In this battle the Germans; using the call signs and radio procedures of the Russian High Command; were able to send false instructions to various commanders countermanding previous orders and altering battle plans. This resulted in a military disaster for the Russians with far reaching consequences.

The major safeguard against imitative deception is an authentication procedure which can guarantee that each message has originated from the authorized user. Communications systems commonly rely on cryptographic authentication procedures which require the user to perform some enciphering operation with a known key. The system performs the same operation in parallel and compares its result with the potentical users. If the results are identical, it is assumed the user possesses the proper key and hence is an authorized user.

The authentication protocol described in this paper uses a parallel encipher and test procedure to safeguard three aspects of the control session: establishment, message flow and exception handling. Each of these three aspects of control link communications offer the spoofer an oppurtunity to mount an attack against the system.

The first safeguard is placed at the point where a user requests the switch to begin an operator dialog. The user would up to this time have established at least an AX.25 level connection to the switch and request to connect to the switches executive software. In some respects, the connection request resembles a conventional computer logon with a secret password; however, the radio channel is always being monitored and the password would not stay secret very long.

The mechanism used by the protocol relies on parallel encipherment of a known constant by both the user and switch. A challenge number (CN) is generated by the switch based on a randomized timer value and is transmitted to the user in response to his connection request. Both the user and the switch then encipher the challenge number using a previously distributed secret key. The user returns the enciphered value to the switch which then performs a comparision with its locally calculated value yielding a match for a valid user.

If the returned value does not match the switches enciphered value it is assumed the user does not possess the secet key. The connection request is denied and the switch breaks the AX.25 connect ion ("hangs up"). At this point no new operator dialog connection requests will be accepted for a period of fifteen seconds. The goal is to prevent a brute force attack by microcomputer which could rapidly test many trial keys.

A benefit of the randomly generated challenge number is that it provides a unique value to associate with each operator connection. This unique value will be used to differentiate messages in the operator session from those of an old session and hence block a playback attack. It is important to note that the switch and not the user generated the challenge number. If the user could generate the challenge number, a spoofer could simply playback an old challenge number and setup to replay the associated old session.

Having been blocked from directly impersonating a valid control operator an attack now open to the spoofer is to attempt to disrupt the control link by insertion,,modification or deletion of valid control link messages. Neither the link nor network protocols offer protection against inserted messages [Borden,1985]. In fact their action is to accept the first correctly numbered packet and ignore subsequent packets with the same send and receive counters; --accepting the spoofer packet while ignoring the valid one.

To overcome this difficulty we have included a message sequence number as part of the authentication construct. The authentication protocol tracks the sequence numbers by computing the next expected sequence after each valid message is received. This value is saved by both the switch and the control operator and is used to compute the next expected message authentication value.

Message alteration is an attack open to the sophisticated spoofer. The goal of the spoofer is to intercept and use a valid frame and packet structure but insert a spoof message into the I field portion of the packet. This attack is more common on wire line systems were an attacker need only insert a cornputer in the line to perform the alteration. In the radio environment, an attack of this type is still possible if a spoofer could automatically intercept: a valid packet, insert the spoof text and retransit the packet in under a few seconds.

A senerio of this attack might be for a spoofer to intercept the incoming packet from the control station by aiming a highly directional antenna at the control station while a second spoofer, closer to the packet switch momentarily blocks reception. This can be done by transmitting a short noise burst to jam the packet switch control receiver making the switch unaware that a valid packet was sent. After intercepting the original valid packet, the first spoofer would overlay the packet I field with the command of interest, recompute the frame check error value and then quickly re-transmit the message to the packet switch.

To defeat this attack the authentication protocol contains a modification detection indicator (MDI) which detects any modifications to the message text. Various types of check-sums have been studied for this purpose and certain vunerablilities have been identified where the "sum" is composed of linear operations. In the authentication protocol used by the packet switch a nonlinear approach is used to reduce this types of exposures.

The MDI value is computed over the message contents including the authentication state indicator of the authentication protocol construct which prevents spoofing. The authentication value is then computed by enciphering the combination of MDI value with both the Challenge and internally stored message sequence number.

3.0 Authentication Protocol

The authentication protocol is essentially a computer oriented protocol with the control operator executing part of the protocol on a personel computer and the remainder executing within the switch. The two communicants exchange an authentication data unit (ADU) which is affixed to each operator message. In the return direction an ADU is used to signal acknowledgments and special conditions and can be received independent of a switch message. Contained within the ADU is the authentication state indicator (ASI) which signals conditions between the two ends.

Figure la. Control Message Format

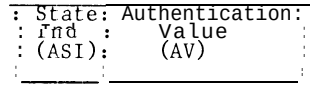

Figure lb. ADU Fields

Figure la shows the operator control message and ADU layout. The common control message typically takes the form of an operator command such as instructions to check status, start or stop operations and on-line debugging. The actual contents of the message can take either a conversional English format or a machine readable format depending upon where the actual command parsing is performed. In the packet radio switch under discussion, command parsing is done at the control operator location and so only machine readable information is exchanged.

Figure lb shows the format of the ADU. The authentication state indicator (ASI) is used to signal the state of authentication and indicates status such as acknowlegements or a command such as session termination.

There is an associated message count field which is stored internal to the switch and the control operator. The count field is used to track the message number and detect lost,inserted or duplicate message. This field is 16 bits in length which was judged adequate for even exceptionally long control sessions.

3.1 Authentication Protocol Description

The authentication protocol consists of four procedures: authentication establishment, message transfer, terminate and error. The four procedures of the protocol operate in series with the exception of the error procedure. This procedure is charged with recovery when a message fails to authenticate. In the authentication establishment procedure the control operator is attempting to connect with the packet radio switch. The procedure calls for the control operator to establish a lower level connection with the switch operator control routines. Once this is done a reliable path is assumed to exist between the control operator and the packet radio switch.

The next step is for the control operator to authenticate himself to the switch by properly encrypting a challenge number (CN) under an initial key derived from the master key and the challenge number (the challenge number serves as a key indicator function). The switch generates the CN via a clock driven random number generator and sends this value to the control operator. The control operator now enciphers the CN by combining it with this key and sends it back to the switch over the control link.

In parallel with the control operator, the switch has also enciphered the CN value for comparision with the one received from the control operator. A match! indicates that the control operator is in possession of the same initial key thus completing the authentication initialization phase.

To add security, the validated CN value is combined with a constant and re-enciphered to produce a session randomizer value (SR) which is stored internally by both the switch and the control operator. This step limits the usefulness to a spoofing station of a intercepted CN value since it will not be used futher in the current form within the authentication procedure. An important purpose of the SR is to prevent playback of a previous session. The SR associates with each session serves as a randomly selected session identification number. The SR is also cne of the fields used to construct the message authentication value for each operator message;, making this value reflect the individual session SR.

Two different approaches were considered in generating the SR value. The first approach is to consider a monotonically increasing counter value to provide unique session numbering. The drawback with the counter approach is it requires long term storage of the counter value. This conflicts with one 0f the switches key design principles of never relying on values which could be modified by a software error for critical operations. Typically, constants and routines are comitted to ROM for reliability; comparatively, values contained in RAM are subject to both software error and system resets.

The second approach; which is used in this implementation; is a SR value produced by a twice enciphered clock driven random number generator. The first encipherment producing the CN value and the second yielding the SR. This approach produces evenly distributed random numbers which have a very small but non-zero probability of repetition, but which needs no long term counter storage.

When an initialization attempt fails, the switch returns an ADU with the authentication reject flag set. At this point there is an oppurtunity for a dictionary 'attack if the switch allowed instant reconnection. Instead. the switch does the equivlent of "hanging up" by signalling the lower level protocols to break the X.25 connection to the control operator and then enforces a 15 second wait before it accepts any new authentication establish requests. This action blocks the ability to qucikly retry successive test keys; one of the prime elements necessary for an economical dictionary attack.

One additional style of dictionary attack should be mentioned. This style of dictionary attack submits test keys intermittently over a long period of time. The conventional dictionary attack relies on speed to submit many test keys in a short time. This style might be termed the fast dictionary attack; comparatively, a slow dictionary attack spreads its tests out over a long time in order not to arouse the suspicions of a control operator. The slow dictionary attack relies on the fact that the system is available around the clock so that tests need only be conducted sporadically.

The goal of the slow style of dictionary attack is; like the fast attack; to find the key currently being used. The slow dictionary attack may be launched when breaking the system is of low priority sort of a bonus if it could be done. It is difficult to guard against this attack since few attempts would occur over an interval. The obvious defense is to use a very large key space to forces the attacker to maintain a large dictionary and perhaps close the time window during which the attacker might test. For example: allow connection requests only during business hours. This strategy is similar to what was done durring World War II with the identification friend or f'oe (IFF) systems. The enemy would test the bombers IFF by sending trial messages hoping to find the current settings. To limit this, the IFF boxes were commonly switched off until within a range where they were needed. This deprived the enemy of their window.

The second procedure of the authentication protocol is the message transfer procedure which authenticates each arriving message. The actual authentication value (AV) is generated by:

$$AV = \text{ENCRYPT}(\text{Key}, \text{MDI}(\text{message-text}) + \text{SR} + \text{Sequence-number})$$

The AV itself is 64 bits in length while the modification detection indicator is a 64 bit value (non-secret) which is calculated on the state indicator. The ADU will be affixed to the end of the current message and transmitted. When the switch receives this message, it will recompute the AV using the expected next message sequence value; and the SR; both of which are retained independently of the control operator. The switch will also recompute the MDI on the received message and derive a test AV value.

Authentication of the current message is performed b comparision of the test AV with the received A%.

Once authentication has taken place there are three possible actions: pass authentication, fail or fail on retry. Messages which pass authentication cause the message counter to be incremented and the message to be passed to the switches operator function. If the message fails to authenticate then no message will be passed to the operator function, instead an error procedure is entered.

In the error procedure, the counters are not updated to prevent a counter synchronization problem which would complicate things later. Instead, the message as received is sent back to the control operator with the reject flag set. The control operator can then retry seven times before the switch declares a formal authentication failure. When such a failure is declared, the authenticated session is cancelled, the switch "hangs up" and the unconnected state is re-entered.

Finally, the termination procedure is used to gracefully shutdown the authentication mechansim. Termination of an authenticated session is always initiated by the packet switch in response to a disconnect command, software failure or idle terminal timeout.

3.2 Cryptographic Considerations

There are a number of design issues surronding the choice of cryptography-applied to this authentication problem. In this section we examine several of these issues including legal, technical and operations and, identify why specific choices were made.

A legal requirement which steered the choice to authentication cryptography rather than-message encryption; which offeres alternative solution techniques; is the plaintext requirement for message information. The plaintext requirement is a regulation which the Federal Communication Commission (FCC) has levied on different communications services, generally prohibiting them from transmiting enciphered information. Until quite recently this meant any type of cryptographic or "scrambling" technique.

When the spectre of unauthroized use of control links such as satellite command channels began to attract serious attention, the FCC relaxed its strict interpertation to permit some use of authentication cryptography. The general test is if the specific application enhances or facilitates communications as opposed to masking its meaning.

With the relaxed definition it has been possible to utilize an on-the-air control link. Commonly, in the past it was necessary to utilize either a dialed or leased telephone line for control of a repeater to solve the operator authentication problem. The expense of installation and maintance of telephone lines; specifically to inaccessible places such as mountain tops; often contributed to the choice of switch placement.

3.3 Implementation Issues

The current system is based on private key ideas allowing us to employ DES, a commonly available commercial cipher algorithm. Three factors drove this choice: economic considerations, cryptographic strength and commercial acceptability.

Economically, DES is available in various hardware formulations and hence, including it at a low cost was possible. We chose a hardware formulation because in terms of storage, corn utational power and delay a software implementation would not be advantageous. A second reason was the large outlay in time and expense for the software assurances necessary for cryptographic processing.

Software assurance techniques for cryptographic systems are concerned with prevening information compromise due to software failure. A well implemented cryptograpic system commonly seperates the secure and non-security aspects of the system and in addition, requires the software design to go through a rigorous failure mode analysis identifying each failure mode and correspondin g assurance. The increased reliability afforded by this procedure guards againt failures in which the cryptographic process fails to encipher, garbles the information in a reversible way or worse, allows key bits to erroneously appear on the communications channel.

In choosing the DES algorithm we surveyed the relavent literature on its cryptographic strength and have found no clear example of a succesful attack. We find the arguments against DES fall into three classes: (a) super-computers have put a known plaintext attack within reach, (b) the government already posses such computers (c) there exists a secret process (a trapdoor) that reduces or negates the computational burden required in solving DES.

In considering each of these three classes of aurguments only the first seems to be clearly verifiable, implying that a careful re-examination of DES will be needed in the future. In the second class; who currently possess such computering power; is largly irrelavent to the switches threat environment. Lastly, the existance of a trapdoor procedure could jepordize the security of this authentication procedure. No clear indications of such a trapdoor have been found in the literature, but if such an attack should come to light DES would have to be replaced.

3.4 Form of DES Utilization

The cryptographic aspects of this protocol are based on parallel encipherment; by both the switch and control operator, of non-secret information under the control of a unique secret session key. With the exception of the key all information is either broadcast or easily calculated and hence an attacker could assemble a sizable collection of plaintext-ciphertext pairs for study. The goal of such a study is to deduce the key value by a close examination of the enciphering process. Therefore, algorithms selection must be limited to those which can resist a known plaintext attack.

3.5 Key Management

A simple key management scheme was developed for the switch to increase the period between rekeying (the cryptoperiod). The technique also increases security by creating a unique key for each session thereby limiting the amount of message traffic under the same key. The scheme implemented for the switch employs a fixed master key of 512 bits which is shared by both the switch and control operator.

The switch generates a session key after the authentication establishment procedure is sucessfully performed by computing a privatly held function on the CN. The output is then used to select 56 bits for the current session key. Since each session uses a different schedule of key bits from the master key, the rate at which an attacker could infer the next session key would be very low.

4.0 Conclusion

This paper has decribed an authentication procedure to control a remotly sited packet radio switching node. The chief problem to be solved was to provide a authentication technique which operates in the face of occasional jamming, spoofing and prevasive monitoring.

The protocol is designed to authenticate a single direction of message flow -- from the control operator to the packet radio switch. Bi-directional and full duplex extensions of the protocol are possible; however, there are special complexities, chiefly in the area of counter and error management which must be solved.

Finally, there is interest in extending the design of the authentication protocol to address new and wider problems within the packet radio network. Among these are extension to the full duplex case, authentication of the switch-to-

switch control protocol and multiple operator situations in which there is a need for split authority. Public key techniques seem to offer many interesting approaches and will probably be used in futher protocol versions.

Appendix

An alternative One-Way Enciphering Function

In the commercial setting in which the packet radio switch is to operate, DES was choosen and implemented as a hardware accessary to the main switch microprocessor. To utilize the DES as a one-way function, the key was first combined with the value to be enciphered and then the result is used as both the message and the key. The output from the encryption operation is taken as the result of a one-way transformation.

While DES was choosen for reasons of its commercial acceptability, test versons of the switch employ a previously described one-way function based on a "random program" (RP) technique first described by Evans, Kantrowitz and Weiss of MIT [Evans,1974]. The goal of developing a new one way algorithm is one of research since one of the one-way cipher described in the literature would suffice for a DES alternative. In as much as the RP one-way function is strictly experimental it could not be offered in place of DES, yet in terms of efficency and compactness it seems superior at this point.

A one-way function is relativly efficent to compute (encipher) but computationally infeasible to invert. In most cases one-way functions are based on a known hard problems choosen from complexity or number theory such as the knapsack, prime factorability or root extraction.

The random program (PR) algorithm is not based on one of the accepted hard problems but on the execution of a series of machine language instructions whose order depends on the specific contents of the algorithm argument. Abstractly, the function consists of a register (R), which contains the value to be enciphered; a set of machine language instructions (M), whose operations alter the contents of the register and a selector function (S) which specifies the machine language instruction to be executed next.

Roughly, the algorithm is executed by the selector function taking a selection of bits from the register and computes an index which designates one of the machine language instructions in M. The designated instruction is then executed, altering the contents of the register. This cycle is repeated some number of times and yields an output value in R.

In order to invert this algorithm the analyst would have to know what order the instructions in M were executed. This in turn depends on the contents of the original value to be enciphered which is unavailable after the algorithm completes execution.

Three factors make the RP function attractive; first, the function does not rely on extended precision arithmetic commonly seen in many of the public key approaches. Second, the computation and storage requirements are competitive if not better than conventional approaches, an important element in the switch design and lastly, the low complexity of the algorithm leads to simpler implementation.

The "pure" RP algorithm described above has a number of weaknesses which must be addressed before an solid implementation could be offered. In the RP algorithm the cryptanalytic strength is derived from the unpredictability of the order of instruction execution. Essentailly, each argument to be enciphered should select a unique series of instructions with each instruction given equal chance of execution at all times.

Both the selector function and the value to be enciphered contribute to the specific order of execution. The selector function output must therefore have a flat probability distribution so that all choices are equally likly over the range of values to be enciphered. Since this range is composed of all values which can be held in the register, all representable bit patterns must be expected.

A simple selector function could be heavily influenced by the bit patterns in the input. This inturn would reflected in the order of instruction selection and execution. The selector function therefore must attempt to "whiten" the values selected from the register and hopefully limit the vulnerability from patterns in the input.

A second area of concern is degeneration in the radomness of intermediate values. For example if the selector function operators on the results of each instruction execution, it is quite possible for patterns to appear which tended to "converge" either by alternating or by containing less and less variety. In either case the equiprobable instruction choice would be compromised.

The chief disadvantage of the RP algorithm is the lack of in-depth understanding of its cryptanalystic strengths and weaknesses. Currently, it is undergoing detailed study and analysis at the end of which we hope to submit the results to peer scrutiny.

References.

ANSI 1985, "X9.9 Financial Institution Message Authentication". American National Standards Institute, New York

Bethancourt, T.E., 1979, "Instruments of Darkness", Holiday House, New York

Borden,D,Fox,T. 1985, "Spoof Resistance of HDLC,X.25 and Certain Commercial Transport Protocols", Private Communications

Diffie, W and Hellman, M. 1976, "New Directions in Cryptography". IEEE Trasactions on Information Theory IT-22, 16(Nov).

Evans, Rantrowitz and Weiss, 1974, "A User Authentication Scheme Not Requiring Secrecy in the Computer", CACM, Jan 1974

Frick, 0. 1945 "War Secrets in the Ether", Aegean Park Press, Laguana Hills, CA

Kahn,D. 1967 "The Code Breakers, The Story of SecretWriting". Macmillian,New York

Konheim, A. G. 1981 "Cryptography, a Primer"', John Wiley and Sons, New York

Meyer,C.and Matyas,S. 1982 "Cryptography". John Wiley and Sons, New York

Needham,R.M. and Schroeder, M.D., 1978 "Using Encryption for Authentication in Large Computer Networks". Comm of ACM 21, 12(Dec)

Newland, P. 1984, "A Few Thoughts on User Verification Within A Party-Line Network" Proc. of The Fourth ARRL Computer Networking Conference, American Radio Relay League, Newington, CT