# POLONIUS: AN IDENTITY AUTHENTICATION SYSTEM

Raymond M. Wong - Thomas A. Berson - Richard J. Feiertag

*Sytek, Incorporated*
1225 Charleston Rd., Mt. View C.A., 94043

## ABSTRACT

Passwords have long been used as the most common method for providing user authentication when accessing remote computer systems. However, there are many security problems associated with passwords including their susceptibility to the attacks of eavesdropping, playback, and exhaustive search. This paper describes a system which offers an innovative solution to the problem of establishing identities over insecure communications channels. The system embodies the security concept of a one-time pad because it requires that a different password be used for each access. Through the possession of a personal authentication device (the PassPort) and knowledge of a unique PIN number, an authorized user is able to generate the correct password to be used for each access.

## INTRODUCTION

This above all, to thine own self be true,
And it must follow, as the night the day,
Thou canst not then be false to any man.

Polonius, Hamlet Act I. Scene iii.

We must often establish our identities to a remote party in order to receive a particular service or begin a transaction. The most common method of personal authentication is to convey to the other party a piece of information which is in theory known only to you and them, i.e., "something you know". The use of a password in logging into a host system is a frequently used example of this. This method of providing personal authentication can easily be compromised because the communications channel over which the password is communicated is often insecure. By insecure it is meant that the communications channel can be easily identified, and that with little further difficulty an intruder can obtain the information which is being used as the basis for the authentication. In addition, passwords are often not selected at random, and are often short for ease of remembrance, which makes them susceptible to exhaustive search attacks.

Polonius is a family of products which permits personal authentication over insecure channels. Polonius addresses the need for personal authentication in both person-to-person and person-to-machine applications. The system can be easily extended for machine-to-machine authentication applications and to mutually suspicious environments. Through the possession of a portable Personal Authentication Device or PassPort and the knowledge of his Personal Identification Number (PIN), a user can authenticate himself to a remote Authentication Server (AS).

This paper describes the Polonius system, the authentication protocol and the cryptographic algorithms used, and then gives a typical application for the system.

## THEORY OF OPERATIONS

The problem with passwords for personal authentication over insecure channels is that the information that is the basis for the authentication is reused. This section gives an abstract description of an authentication system that solves this problem. The Polonius system, which applies similar ideas, is then described.

### One-Time Pads

The knowledge of a specific piece of information can be used for reliable authentication if the information is always different after every valid transaction, and if knowing all preceding pieces gives an intruder no information as

CH2150-1/85/0000/0101$01.00 © 1985 IEEE

to the next valid piece. The consecutive pieces of information are often referred to as a one-time pad. A one-time pad is a list of random values given to each of the communicating parties, Figure 1. Each entry in the one-time pad is used only once.

There are two major practical problems with the use of one-time pads as a method for authentication.

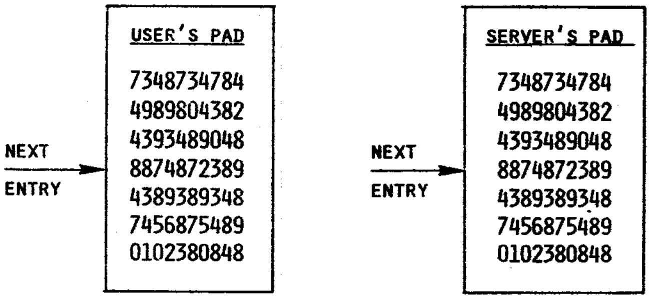

FIGURE 1. ONE-TIME PADS

First, the entire list must be kept secure against purposeful or accidental disclosures since this would destroy its value as a means for authentication. Although providing secure storage for a potentially large list of numbers may be feasible in a fixed site such as a central facility, it incurs the cost of special handling procedures for safeguarding. Providing secure distribution and storage for large amounts of information in portable units for use in remote sites or by persons traveling to different environments is much more difficult.

Second, each of the communicating parties must retain the knowledge of what is the next valid entry in the one-time pad. In effect, the parties must be using the one-time pad in synchronization.

These two problems can be solved by providing the users with the capability of generating entries of the one-time pad upon demand. The requirement for secure storage of large amounts of data is immediately eliminated. The problem of pad synchronization can be addressed by having the party requiring authentication request a particular pad entry by name. Entries may be requested in any order but must not be reused, since a reused entry is insecure. The two parties are able to generate entries from their shared one-time pad using a cryptographic algorithm and a Secret Key known only to them. The disadvantage with a scheme which generates pad entries is that the entries are no longer truly random, but are pseudo random and are therefore subject to mathematical attacks.

The next section gives a description of the Polonius system which is based upon the pad generation scheme described above.

## Polonius System Description

It is envisioned that the Polonius system will be used by service providers which require reliable personal authentication over insecure communications channels of their users before a service is provided. Authentication can be provided by giving each user and the service provider the ability to generate entries from a one-time pad. Each user is given the ability to generate entries from a different one-time pad, but the service provider has the ability to generate entries from the one-time pad associated with any user. The Polonius system consists of two classes of devices. The first is a Personal Authentication Device or PassPort which is a portable device issued to the users. Providing each PassPort with a unique Secret Key allows each PassPort to generate a different one-time pad. The second device is the Authentication Server (AS) that is used by the service provider to determine whether the user has provided the correct authentication information. The AS must have knowledge of the Secret Keys for each of the user PassPorts.

Figure 2 is a diagram of the Polonius authentication protocol. In order to use the PassPort for personal identification the user must first supply a Personal Identification Number or PIN. The PIN is used to protect against the use of the PassPort by someone other than the owner. The Polonius system therefore requires that for valid authentication, the user must possess a unique device (the PassPort with the Secret Key associated with the user by the AS) and know a unique value (the user's PIN). The PassPort will permit the user to proceed as normal for any PIN value entered but will only provide correct authentication for the correct PIN. If use of the PassPort were denied for incorrect PIN values, the correct PIN could be determined by searching the space of all possible PINs. The PIN is never transmitted to the remote location, and is therefore secure against channel eavesdropping.

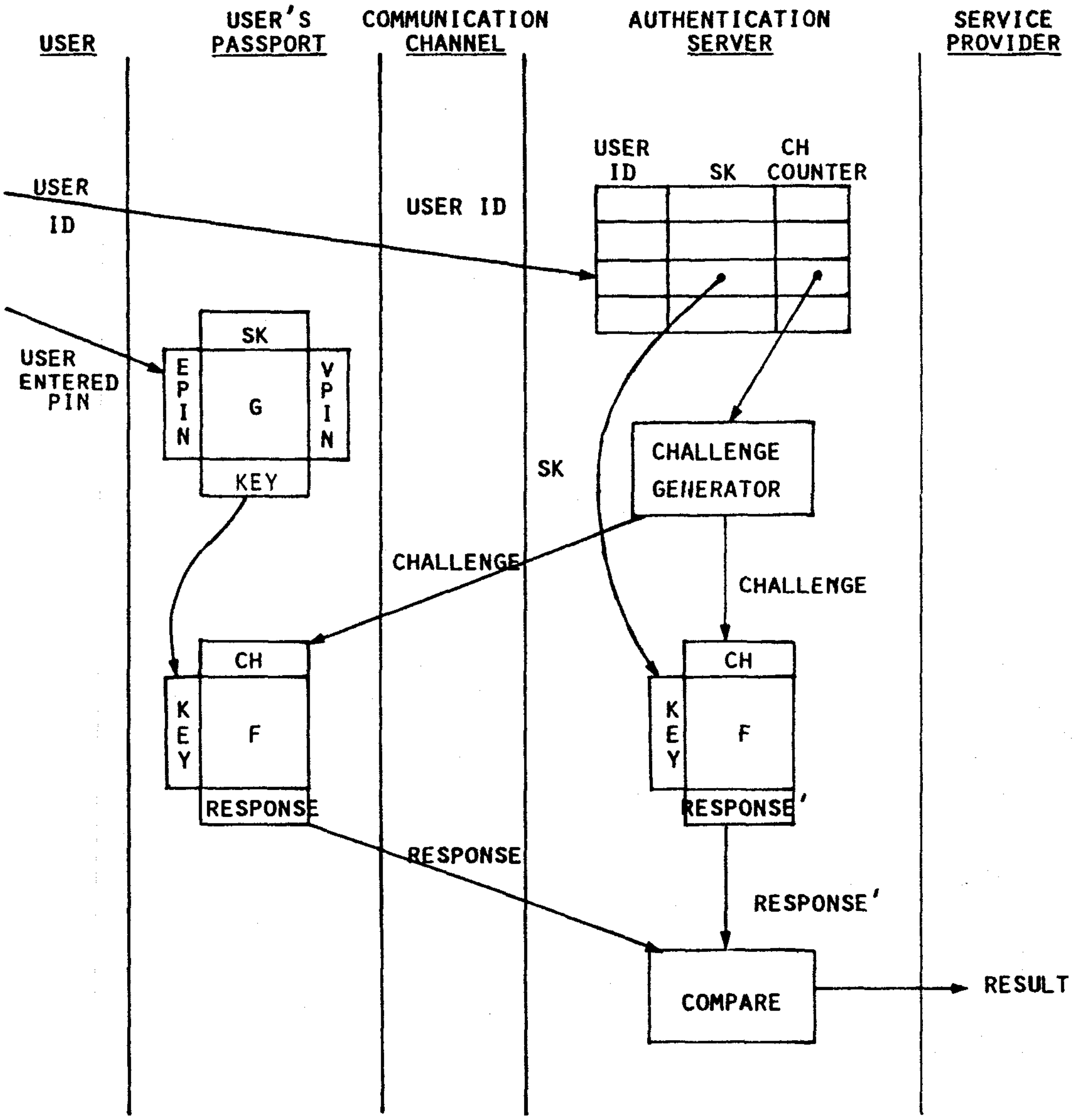

FIGURE 2. POLONIUS AUTHENTICATION PROTOCOL

The user must establish communications and identify himself to the service provider's AS. The user must provide a unique User ID which allows the AS to distinguish his PassPort from that of other users. The User ID may be a user name or an account number. The AS maintains a table that consists of an entry for each User Id. Each entry consists of the Secret Key for the PassPort, and a Challenge Counter that allows the AS to request a previously unrequested entry from the one-time pad. The AS uses the Challenge Counter to generate a 7 digit Challenge number which may be interpreted as naming the one-time pad entry the user must return to be authenticated.

The Challenge is communicated to the user who enters it into his PassPort. The PassPort uses the stored Secret Key, the user entered PIN, and the Challenge to generate a 7 digit Response.

$$R := f(g(SK, ePIN), CH)$$

$$g(SK, ePIN) := SK \quad ; \text{ iff } ePIN = vPIN,$$

$$g(SK, ePIN') := SK', SK' \neq SK \quad ; \text{ for } ePIN' \neq vPIN,$$

$$g(SK, ePIN'') := SK'', SK'' \neq SK \quad ; \text{ for } ePIN'' \neq vPIN,$$

$$\text{and} \quad SK' \neq SK'' \quad ; \text{ for } ePIN' \neq ePIN''.$$

R = Response
SK = Secret Key
ePIN = user entered PIN
vPIN = valid PIN
CH = Challenge

The function g(.,.) uniquely modifies the Secret Key depending on the value of the user's entered PIN. If the user's entered PIN is equal to the valid PIN stored in the PassPort, then no modification is made.

The AS meanwhile calculates the expected response, R', using the user's Secret Key and the Challenge.

$$R' := f(SK, CH)$$

The AS determines whether the user has correctly authenticated himself by comparing the Response returned by user to the expected Response', and informs the service provider of the result.

As the Response returned by the user is a function of the entered PIN, a number of messages can be signaled to the AS through a subliminal channel hidden in the Response. With a single valid PIN, two messages, either authentication valid or authentication invalid can be signaled. More than one valid PIN can be defined in the PassPort, and the AS can determine which is used by examining the Response returned by the user. For example, a second valid PIN can be defined in the PassPort which is assigned the meaning that the PassPort is being used under duress. No observer who does not possess the PassPort's Secret Key can tell from the message content or timing that the duress PIN had been entered.

## THE PASSPORT

As described in the previous sections, Polonius applies cryptographic techniques to the solution of the personal authentication problem. Polonius is made practical by innovative implementation of cryptographic mechanisms in the user PassPort. In order to gain user acceptance, the PassPort must be easy to use, portable, and highly reliable. In addition, the PassPort must be affordable, and the cryptographic keys and PINs stored within the PassPort must be protected from tampering. This section will describe the PassPort device which has been designed and implemented to meet these criteria.

### Packaging

The PassPort is packaged as a pocket-sized calculator which can easily be carried by the user. It appears at first glance to be a typical pocket calculator, and does perform calculator functions in addition to those of personal authentication.

### PIN Registers

PINs are used for both assuring proper use of the PassPort for personal authentication, and for subliminal message signaling to the Authentication Server. A valid PIN is at least 4 and not more than 8 digits long.

### Secret Key Registers

The Secret Key is used in the computation of the Response. The PassPort is capable of storing two different Secret Keys. If more than one Secret Key is stored the user is permitted to select which key is used for each Response computation.

Each Secret Key register has two fields. The key field holds the value of the Secret Key, and the mode field specifies one of 16 encryption algorithms. Therefore, by specifying a particular Secret Key register for a Response computation, the user is actually specifying both the algorithm and key to be used.

### Response Computation

The Response is computed as a function of the Challenge, the user entered PIN, and the Secret Key. By making the Response a function of the user entered PIN, the authentication procedure signals three different messages:

1. valid authentication resulting from the use of PIN1.

2. valid authentication resulting from the use of PIN2 (if present).

3. invalid authentication.

Assuming that the mode bits of the Secret Key register have selected the DES algorithm, Figure 3 illustrates the computation of the Response.

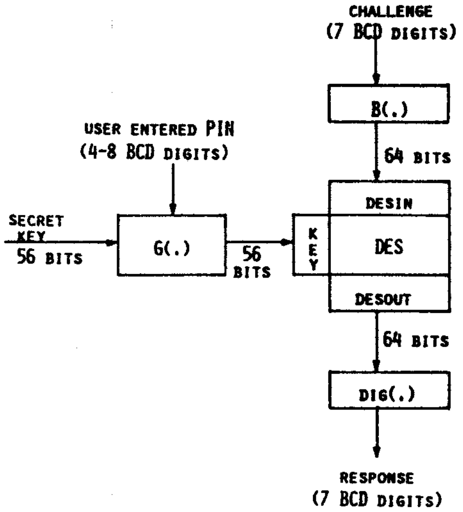

FIGURE 3. RESPONSE COMPUTATION

Since the DES algorithm needs a 64 bit input, B(.) converts the Challenge, a 7 digit decimal number, to 64 bits.

The output of the DES operation is a 64 bit number which must be reduced into a 7 digit Response. This is performed by the function DIG(.). DIG(.) has been designed to ensure uniform distribution of output under the assumption of uniform distribution of input.

## Message Signaling

Message signaling is accomplished by transforming the Secret Key depending on the value of the PIN entered. The transformation g(.) performs the following.

1. If the entered PIN = PIN1, the key is the Secret Key.
2. If the entered PIN = PIN2, the key is the Secret Key altered in a particular fashion.
3. Otherwise the key is obtained by altering the Secret Key such that each wrong PIN yields a different key, none of which are the key obtained in case 1 or case 2 above.

The transformation g(.) insures that the key used in the DES operation is different for each PIN value entered. This generates different and unpredictable Responses for different values of the user entered PIN, and prevents an unauthorized user from searching for the valid PINs by observing the Responses as a result of holding the Challenge fixed and varying the PIN.

In order to determine the validity of the authentication, the AS must compute the Responses using the keys associated with the two valid PINs. If the Response returned by the user matches any of these computed Responses, the authentication is valid and the message associated with the PIN has been signaled. Otherwise, the result of the authentication procedure is invalid.

## Use of Multiple Secret Keys

A PassPort has the capability of storing more than one Secret Key in order to permit a user to have the ability to authenticate himself to more than one independently administered application without requiring him to possess a PassPort for each application. The user is able to specify which key is to be used in the computation of the Response.

## Implementation Notes

The PassPort block diagram is shown in Figure 4. The POLONIUS ENGINE chip is the heart of the system.

PINs and Secret Keys are entered through the keypad during initialization of the PassPort. Once entered these values never leave the POLONIUS ENGINE and can not be displayed on the LCD. The POLONIUS ENGINE retains these values only so long as the battery is connected. If the battery power is disconnected because of tampering or replacement of the battery, the PIN and Secret Key values are lost and must be reinitialized. This is the only method of changing the PIN and Secret Key values stored in a PassPort.

Much careful attention was paid to space-time tradeoffs during the design and implementation of the cryptographic algorithms. The PassPort user perceives no delay during the calculation of a Response.

## A TYPICAL APPLICATION: SECURE LOGIN

Figure 5 illustrates an application of the Polonius system for secure logins. Each user of the system is issued a PassPort and PIN. The host is attached to an AS which acts as an advisor for determining authenticity of users. The host has only a transaction level interface

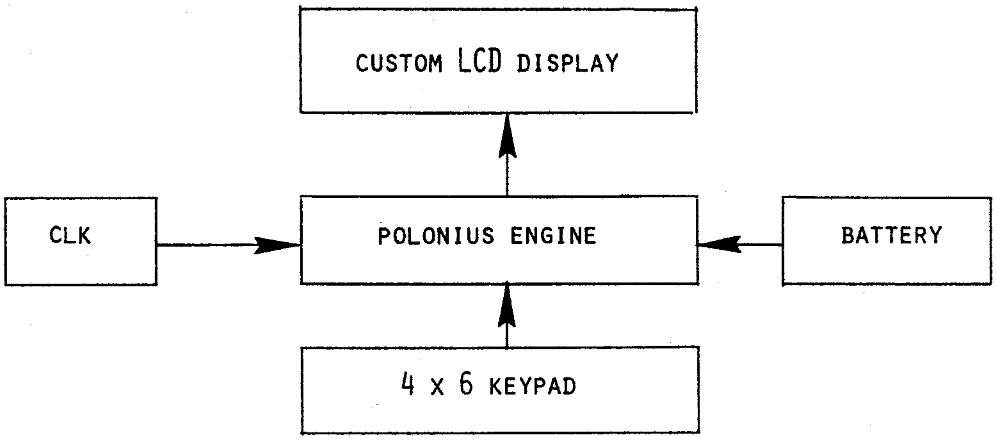

FIGURE 4. HARDWARE BLOCK DIAGRAM

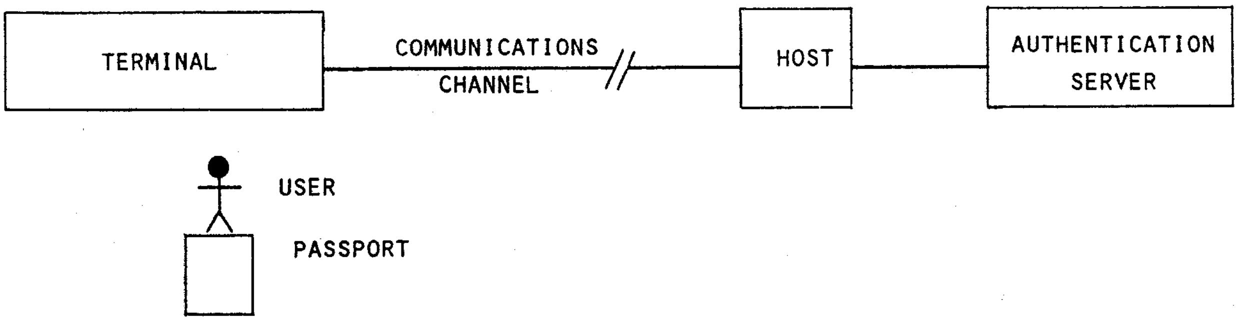

FIGURE 5. SECURE LOGIN APPLICATION

with the AS. The database of user Secret Keys is encapsulated with the AS to prevent unauthorized access and modification of this information. This is an example of the computer security design principle called "minimum privilege".

A typical login scenario would proceed as follows:

1. The user establishes a connection with the host system.

2. The host prompts the user for his user ID.

3. The user enters his user ID.

4. The host receives the user ID, and issues to the AS an Authentication Request message containing the user ID.

5. The AS determines if the user ID is in its database. If yes, the AS retrieves the record for this user ID which contains both the Secret Key and the Challenge Counter. The Challenge Counter keeps track of which Challenges have been previously used and allows the AS to compute the next Challenge. The Response is then computed using the Secret Key and Challenge. Both Challenge and Response are returned to the host. If the user ID is not in the AS database, the host is so informed. However a randomly generated Challenge is still returned in this case. The Challenge prevents hackers from searching for valid user IDs by noticing a difference in host responses for valid and invalid IDs.

6. The host issues the Challenge to the user and prompts for the Response.

7. The user enters his PIN followed by the Challenge into his PassPort which computes the Response. The Response is sent by the user to the host.

8. The host compares the user's Response with the Response predicted by the AS and determines authenticity.

## CONCLUSION

A system for establishing identities over insecure channels of communication based upon the concept of a one-time pad has been described. Polonius offers a much higher level of security over the commonly used password systems. Many of the problems associated with passwords such as eavesdropping, playback and exhaustive search are solved by Polonius. In addition, a user must both possess a PassPort and have knowledge of a unique PIN in order to be authenticated. The system is made practical through the innovative engineering of the user PassPort device.

## FURTHER READINGS

The following references are included for additional information on the subject of personal authentication.

[1] Denning, D.E., Cryptography and Data Security, Addison Wesley, Reading, Mass. (1982).

[2] Diffie, W. and Hellman, M., "Privacy and Authentication: An Introduction to Cryptography", Proc. IEEE Vol 67(3) pp. 397-427 (Mar. 1979).

[3] Feistel, H., Notz, W.A., and Smith, J.L., "Some Cryptographic Techniques for Machine to Machine Data Communications", Proc. IEEE Vol. 63(11) pp.1545-1554 (Nov. 1975).

[4] Morris, R. and Thompson, K., "Password Security: A Case History", Comm. ACM Vol. 22(11) pp. 594-597 (Nov. 1979).