

Advantages of **INDALA** **FLEXPASS**<sup>™</sup>  
High Security Proximity Products

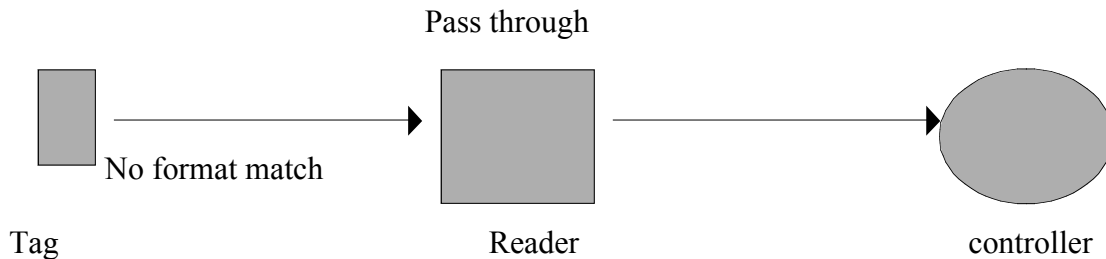
---

To the layman, access control components look alike. They are all based on similar technology, adhere to the same laws of electro-magnetic physics and appear to work alike. A potential buyer is likely to view them as mere commodities with little consideration of the unseen differences.

There are clear differences in the way the access control products are designed, manufactured and integrated into a customer's solution. A clear explanation of the technology will help educate the discerning security professional to the differences so they will recognize the best solution.

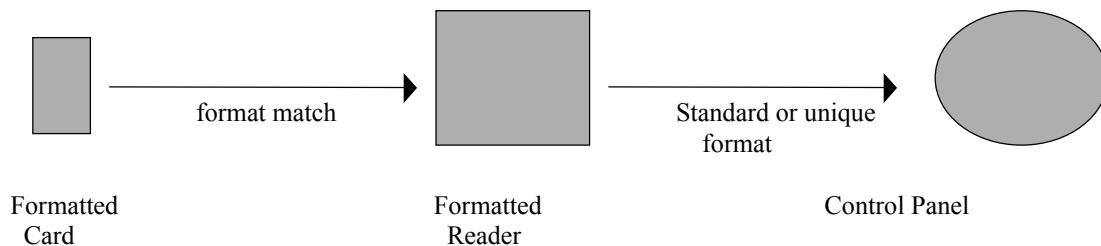
What are the popular Prox topologies implemented today?

The basic access control system architecture consists of a card, a reader, a door controller and a central control panel. The card (or a tag) contains the necessary personal identification data and transfers the data in a format which is recognized by a Prox reader and transported to the controller for the necessary action.



In many Proximity solutions the reader interprets the card's format. The input signal is passed through to the controller, which then does the job of deciding on the validity of the access credential presented. This architecture means that the manufacturer's reader will read any card manufactured by that vendor and pass the data to the controller for a decision to open the door or not. A person may accidentally discover that their card can be read on the building next door. If the facility code and identification numbers are valid for both systems, the one card could allow unwanted access to the second building.

### ***FlexSecur: Security is the big difference***



In Indala's *FlexPass* architecture (figure above), both the card and the reader are formatted. The reader will not recognize a card unless it has the correct format. The reader not only recognizes the card, but also has a format of its own which validates the card format when presented, before allowing the signal to pass on to the controller. This means that the card and the reader both have a format which must match. This prevents anyone with casual knowledge of security systems from attaching readers and cards to a system to gain access.

## **What is the implication of these two approaches?**

In the first example, the reader does not represent an additional layer of security. So, any reader can be added to an installation and immediately communicate with the control panel. Or any card can communicate with the control panel. Direct access to the control panel can permit a security breach. This method also puts the entire systems burden on the control panel. In large systems the transaction load could slow the system down and possibly giving a slow response to the card read.

In Indala's *FlexPass* solution the reader represents an additional level of security by virtue of a 'format', which does not allow cards without the proper encryption to communicate with the control panel. This additional security also keeps the system's response quick because the control panel is not burdened with unnecessary transactions. Indala *FlexPass* readers also have a "Quick Flash" feature that immediately informs the credential holder that the card has been successfully read. This is an advantage in larger systems or during "rush" hour when the control panel is receiving a high volume of transactions in a short period which may slow down system responses.

## ***Security is the issue***

Indala *FlexPass* Proximity system allows the system designer up to 172 bits of formatting in the card. This permits the systems professional all that is necessary to design a high security encryption for the card.

The following document describes Indala's technical advantages over the competition when it comes to security.

---

This document provides information on the security features of the Indala, Flex-PassASP+ cards and Flexpass ASP+ readers, and ProxSmith Toolkit. These features are configured and enabled by the ProxSmith Toolkit. These features are implemented by the combination of ProxSmith project definition features, data structures in the cards and configurable options and firmware internal to Flexpass ASP+ readers. Project File Security, Card Data Security and Reader Data Security implement these security features. These items comprise a suite of security layers that protect card data from compromise.

## **I. Card Data Security**

### **A. Link Layer Protocol**

The Indala FlexPass ASP+ Cards utilize a proprietary through the air protocol that intersperses synchronization bits with the programmed data bits. This means that a card programmed with another card's link protocols will not be read by Indala readers and that cards that are programmed by others without knowledge of the link layer protocol and synchronization bits will not be read by Indala readers.

### **B. Card Data Encryption for Programming**

The data to be programmed onto a card as defined by the ProxSmith project file is processed by the ProxSmith program by the following process prior to the data being programmed to a card:

1. Each field's data is placed into the appropriate position in a temporary data buffer; this in effect scrambles the data field order on the card.
2. The card reader password, either default or user selected, is placed in the temporary buffer.
3. The temporary data buffer is then manipulated to apply either the default, or the user selected encryption key, if it has been enabled.
4. The temporary data buffer is then manipulated to insert the link layer protocol bits.

### **C. Card – Reader Password**

A 30-bit default password, (or a user-defined password) defined by the project file may be selected and programmed into each card. This card reader password is subjected to the card data encryption as noted above. When this feature is enabled the reader may only read those cards whose password matches the one stored in the reader. Cards with no password or cards with a different password will be ignored by readers that have card-reader password enabled. The card user data capacity is reduced by 30 bits when card reader password is used.

### **D. Primary Bit Count**

FlexPass ASP+ cards may be programmed to contain primary data, secondary data or both. Upon reading a FlexPass ASP+ card, the FlexPass ASP+ reader outputs only the data bits marked as primary.

All FlexPass ASP+ cards are programmed with the primary bit count. If enabled as part of the project file definition, the primary bit count will be validated by the reader.

---

If the reader is programmed with Primary Bit Count Matching enabled, then the number of primary data bits on a card must match the number programmed into the reader in order for the reader to output the primary data. For example, a system is designed with Primary Bit Count Matching enabled and the card format is specified to have 26 primary data bits. The reader format will therefore be specified such that readers can only read cards whose number of primary data bits are 26. These readers will ignore cards with other than 26 primary data bits.

### **E. Card Block Locking**

User data on FlexPass ASP+ cards are stored in blocks of bits. A ProxSmith project definition can be configured to lock each of these blocks individually.

Once a block is locked it cannot be reprogrammed by the ProxSmith programmer or by others. This makes the cards a read only device. Were the end user desires to update card data from time to time, Card Block Locking must not be used.

## **II. Reader Data Security**

Cards which are presented to the reader are validated by a set of processes that validate link layer, process lockout timers and enforce a set of data comparisons. There is a minimum of three validations done; link layer protocol, card data lockout, and card data compares. Additionally a ProxSmith project can add additional security features as described below.

### **A. Link Layer Protocol**

The Indala FLEXPASS ASP+ Readers utilize a proprietary through the air protocol that intersperses synchronization bits with the programmed data bits contained on the cards. The reader uses these protocol bits to validate the card that is presented to the reader. This means that a card programmed with other card to reader link protocols will not be read by Indala readers and that cards that are programmed by others without knowledge of the link layer protocol and synchronization bits will not be read by Indala readers.

### **B. Card Data Lockout Timer**

When a card is presented to the reader a set of lockout timers is checked. If the card presented is a repeat of a card presented within the lockout timer setting, the card will be ignored, and the timer restarted. If the card is different from the previous card, or the lockout timer is expired, the lockout timer is started and the raw card data is further validated by the card data compare process. A new card presented to the reader after the lockout time expires, or differing in value, also resets the card data compare counter.

### **C. Card Data Compare Counter**

After a card has passed the lockout timer tests the value is compared to the previous value received. If the card data matches the previous value a match counter is incremented. When the match counter equals the ProxSmith project file defined compare value, the card data is made available for further validation by the reader. If the data does not match the previous data the match count is

---

cleared, the lockout timer is started. The card presented to the reader must then be qualified by the card lockout timer and the card data compare counter before any output is produced by the reader.

#### **D. Card Data Decryption by Reader**

The reader further decodes card data that has been validated by the card data compare and lockout functions. The reader processes the card data by the following process.

1. The temporary data buffer is then manipulated to remove the link layer protocol bits.
2. The temporary data buffer is then manipulated to decode the data using the encryption key, if it has been enabled by the project definition.
3. If the project file has enabled the card reader password, the password received from the card is compared with the password stored on the card. If the passwords match, the data is further processed. If the password does not match the data is discarded. The validation is started over with a new card presentation and all of the qualification must be repeated.
4. If successful the reader then manipulates the data to allow for further processing .

#### **E. Card – Reader Password**

A 30-bit default password, or a user-defined password defined by the project file, may be selected and programmed into each reader. When this feature is enabled the reader may only read cards whose password matches the one stored in the reader. Cards with no password or cards with a different password will be ignored by readers that have card-reader password enabled.

#### **F. WatchDog Output**

When this reader option is enabled, as part of the project definition, the reader will automatically output a WatchDog pattern every 60 seconds. Enabling the WatchDog allows a host to monitor the on-line status of the target readers. Then the host, expecting the WatchDog pattern every 60 seconds, can signal an alert status in the event that the reader is tampered with.

#### **G. Primary Bit Count Matching**

If enabled as part of the project file definition, the primary bit count stored on a card will be compared to the primary bit count stored in the reader. For example, if a ProxSmith project is created with Primary Bit Count Matching enabled and the card format is defined to have 26 primary data bits, the primary bit count programmed to the card and reader will be 26. The reader can only read cards whose number of primary data bits are 26 and will ignore all others.

#### **H. Host Data Stream**

The reader output data stream sent to the host controller is configured by the ProxSmith project file and downloaded to the reader when programmed. The project file configures the reader for output timing, output bit sequence, constant

---

data fields and other data patterns that are defined by the card format definition as part of project file is creation. The card data layout as well as the reader output fields are also included in the card data structure when the card data is defined. The reader can additionally compute parity on the data stream prior to inserting the parity bits into the output stream. This enables a host controller to detect corrupted data by a parity check at the host.

### **I. Reader Configuration Data**

The reader may be configured to allow downloading of the configuration data only one time. When this feature is enabled as part of the project definition, the reader can be programmed using ProxSmith only once. A special unlock card must be presented to the reader to allow re-writing of the configuration data.

The programming of the reader utilizes a proprietary protocol to configure the reader and is only accessible under special reader operating conditions. This configuration protocol is not available when the reader configuration has been locked.

## **III. Project File Security**

The ProxSmith project file defines the data and security relationships between cards, readers and host data. A project file can have the following security attributes.

### **A. Card and Reader Configuration Matching**

The ProxSmith project file contains the definitions of the fields that are to be programmed onto cards, the mapping of these fields into the memory map of the cards, as well as the fields that will be processed by the reader to produce a host data stream. The cards as well as the readers must both be programmed using the same project file in order to successfully read a card, and produce the correct host system data sequence. The ProxSmith project file is used by both the card programmer tool and the reader programming tool to ensure the the card and reader are a matched pair.

### **B. Project File Encryption**

Project files are encrypted using the Microsoft's CryptoApi and RSI Data Security Inc's MD5 algorithm as supplied as part of product. The MD5 algorithm creates a "key" that is used to encrypt the project file. The "key" is not made available to the user and the use of the key is encompassed by the ProxSmith code.

All projects utilize a default encryption key that can be changed by the purchaser to a different password by use of the Administrator Password (See manual K001945-000 T5 page 21-26)

### **C. User Access Levels**

Two access levels are available, Owner and User. Owner's are a class of ProxSmith operators that has the ability to create and edit card and reader data that makes up a project file. Owners must log in with the Administrator Password if password protection has been enabled. User level operators do not have the ability to create or edit card or reader configuration data contained in the project files.

---