IBM Client Security Software version 5.10 Deployment Guide

First Edition (September 2003)

Preface

IT administrators must understand and plan for numerous factors when deploying IBM® Client Security Software. This guide is not intended to explain how to use the Embedded Security Chip or Client Security Software; rather it is a guide for how to deploy the software to Embedded Security Chip-equipped computers across an enterprise.

Audience

This guide is intended for IT administrators, or those who are responsible for deploying IBM Client Security Software (CSS) on computers in their organization. The guide is intended to provide the information required for installing IBM Client Security Software on one or many computers, provided that licenses for the software are available for each target computer. The IBM Client Security Software application provides a User Guide, an Client Security Software Administrator's Guide and application helps, which you can consult for information about using the application itself.

Contents

Chapter 1. Considerations before deploying IBM Client Security Software 5.10

There are various ways to deploy IBM Client Security Software (CSS), which uses the IBM Embedded Security Subsystem (ESS) hardware that is integrated into IBM personal computers. This document will help you determine how to deploy the ESS in your environment. It is important to look at the process of how your company deploys computers from image creation to the way the PC is given to an end user. This process will greatly influence how your company deploys ESS. The IBM ESS is composed of essentially two parts. (See Figure 1):

1. Client Security Software
2. Embedded Security Chip

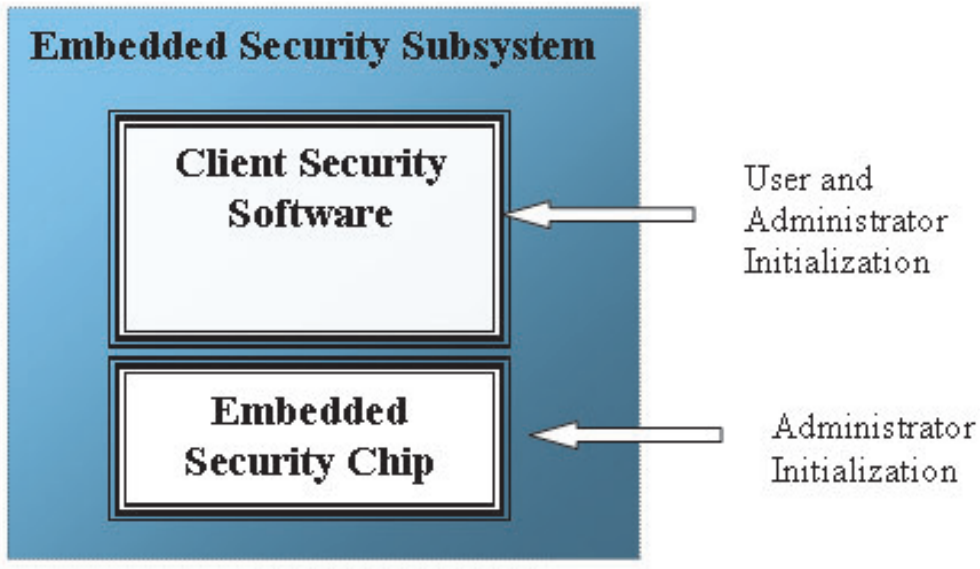

Figure 1. IBM Client Security System components

Requirements and specifications for deployment

If you plan to install IBM Client Security Software on computers that are equipped with the Embedded Security chip, plan on the following server storage and download requirements and installation times:

1. IBM PC with Embedded Security Chip
2. Server Storage requirement for installable code: approximately 12 MB
3. Average per-user server storage requirement for key archive data: 10 K per user for archive storage

Chapter 2. How the Embedded Security Chip functions

The IBM Embedded Security Chip is represented graphically in Figure 2. There are three major components:

1. Administrator password
2. Hardware public key
3. Hardware private key

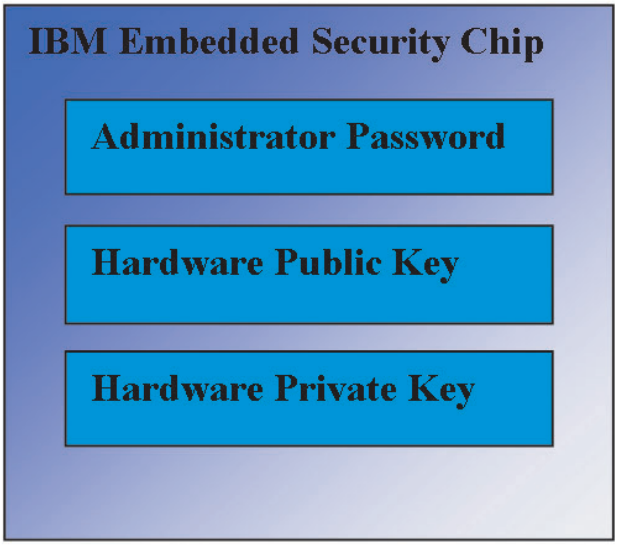

Figure 2. Data held in the IBM Embedded Security Chip

The hardware public and private keys are unique on every computer. The hardware private key can never be extracted from the chip. New key pairs can be generated, but the hardware keys cannot be extracted from the chip.

The administrator uses the administrator password to access the following functions, including:

- Adding users
- Setting security policy
- Setting passphrase policy
- Enrolling smartcards
- Enrolling biometric devices

For example, an administrator might need to enable an additional user to take advantage of the Embedded Security Chip features and functions. The administrator password is set when the Client Security Software is installed. Details regarding how and when the administrator passwords are set are covered later in this document.

Important: Develop a strategy for maintaining administrator passwords, which must be established when first configuring ESS. It is possible for each computer with an Embedded Security Chip to have the same administrator password, if the

IT administrator or security administrator so determines. Alternatively, each department or building can be assigned different administrator passwords.

The other components of the IBM Embedded Security Chip are the hardware public key and hardware private key. This RSA key pair is generated when the Client Security Software is configured.

Each computer will have a unique hardware public key and a unique private key. Random number capability on the IBM Embedded Security Chip ensures that each hardware key pair is statistically unique.

Figure 3 on page 5 describes two additional components of the IBM Embedded Security Chip. Understanding these two components is critical for effectively managing your IBM Embedded Security Subsystem infrastructure. Figure 3 on page 5 shows the administrator public and private keys as well as user public and private Keys. The following is a summary of public and private keys.

- Public and private keys are considered a ″key pair.″
- The private and public keys are mathematically related such that:
  - Anything encrypted with the public key can only be decrypted with the private key.
  - Anything encrypted with the private key can only be decrypted with the public key.
  - Knowing the private key does not enable you to derive the public key.
  - Knowing the pubic key does not enable you to derive the private key.
  - The public key is generally made available to everyone.
- The private key must be aggressively protected.
- Public and private keys are the basis for public key infrastructure (PKI).

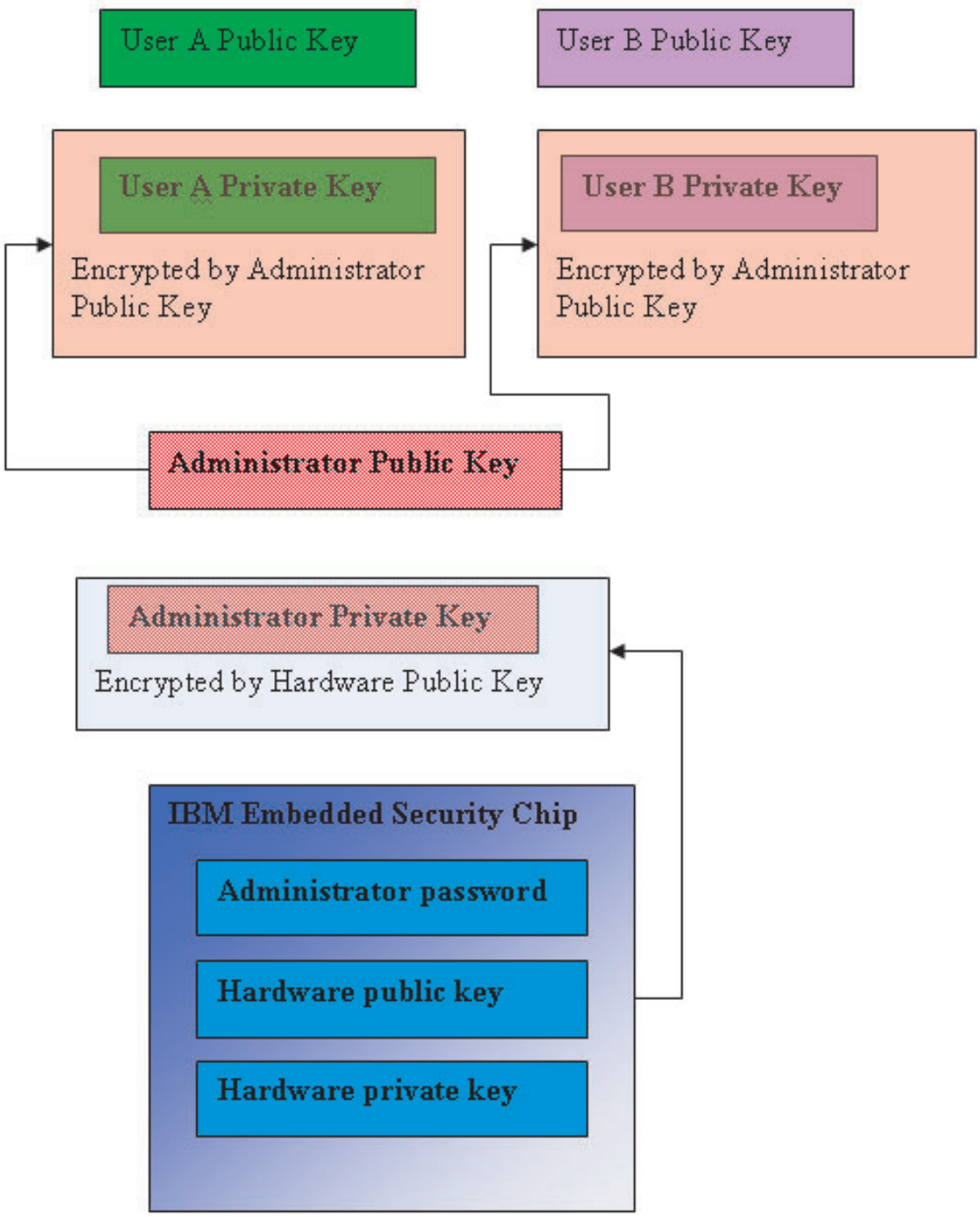

Figure 3. Several layers of encryption provide strong security

Key-swapping hierarchy

Part of the IBM ESS architecture is a "key-swapping" hierarchy. The details of precisely how this works will be covered in the *Administrator's Guide*; however, we introduce the concept here as it applies to mass configuration, deployment, and management. In Figure 3, you can see the Hardware public and Hardware private

key. As mentioned previously these keys are created by the Client Security Software and are statistically unique on each client. Above the IBM Embedded Security Chip you can see the Administrator public and private Key pair. The Administrator public and private key pair can be unique on all computers or they can be the same on all clients or a subset of clients. The advantages and disadvantages will be discussed later in this document. The Administrator public and private keys perform the following:

- Protect user public and private keys
- Enable archiving and restoration of user credentials
- Enable user credential roaming, which is described in the *Administrator's Guide*

Why key swapping?

In the following sections you will read about users in the IBM ESS environment. The details of how to set up IBM Client Security Software and ESS to accommodate these users will be covered in those sections. In this case we will simply state that each user has a public and private key. The user's private key is encrypted with the Administrator public key. From Figure 3 on page 5, you can see that the Administrator private key is encrypted with the hardware public key. Why do we bother encrypting these various private keys?

The reason goes back to the hierarchy mentioned earlier. Due to limited storage space in the IBM ESS, only a limited number of keys can be in the chip at any given time. The Hardware public and private keys are the only persistent (from boot to boot) keys in this scenario. In order to enable multiple keys and multiple users, IBM ESS implements a key swapping hierarchy. When ever a key is needed it is ″swapped″ into the IBM Embedded Security Chip. By swapping the encrypted private keys into the chip, the private key can be decrypted and used only in the protected environment of the chip.

The Administrator private key is encrypted with the Hardware public key. The Hardware private key, which is only available in the chip, is used to decrypt the Administrator private key. After the Administrator private key is decrypted in the chip, a user's private key (encrypted with the Administrator public key) can be passed into the chip from the hard disk and decrypted with the Administrator private key. From Figure 3 on page 5, you can see that you can have multiple users' private keys encrypted with the Administrator public key. This provides the ability to set up as many users as necessary on a computer with the IBM ESS.

Chapter 3. Client Security System in action

The following is an example of how the IBM Embedded Security Subsystem and Client Security Software work together: The Windows log-on prompts User A to log on and User A does so. The IBM Client Security System determines who the current user is through information provided by the operating system. The

Administrator private key (encrypted with the Hardware public key) is loaded into chip. (See Figure 4.)
The Hardware private key (which is only available in the chip) decrypts the

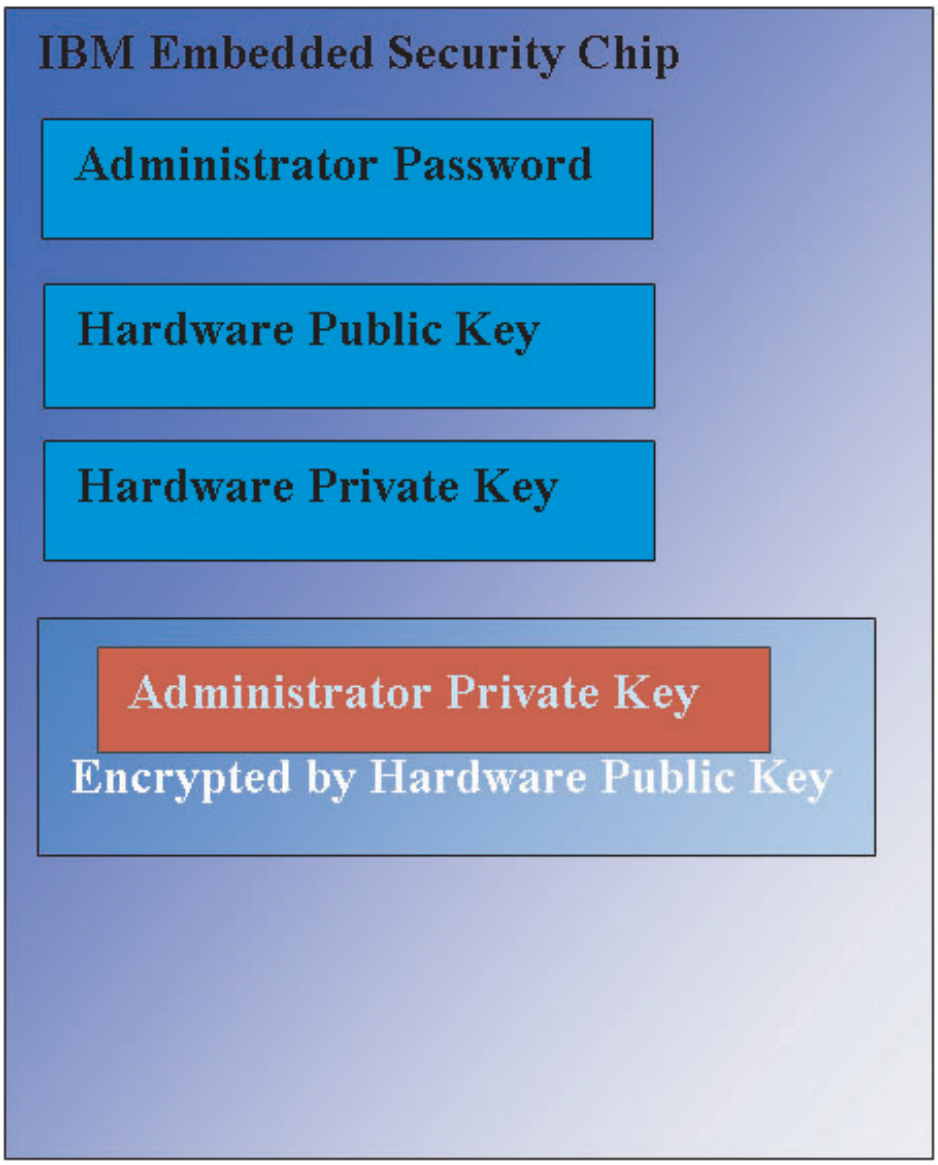

Figure 4. The Administrator private key, which is encrypted by the hardware public key, is loaded into the Embedded Security chip,

Administrator private key. Now the Administrator private key is available for use in the chip (See Figure 5.)

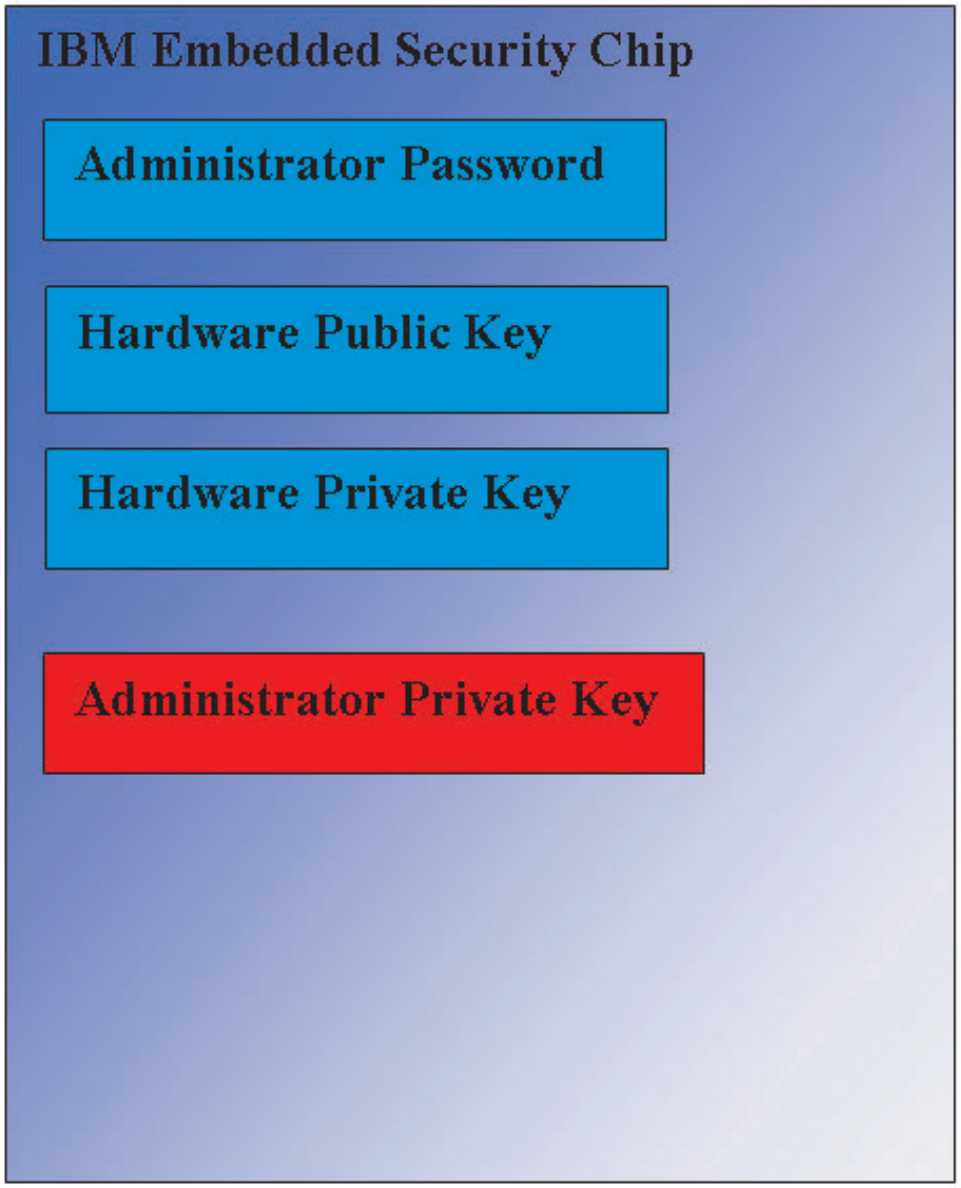

Figure 5. The Administrator private key is available for use in the security chip.

Because User A is logged onto the computer, User A's private key (encrypted with the Administrator public key) is passed into the chip (See Figure 6).

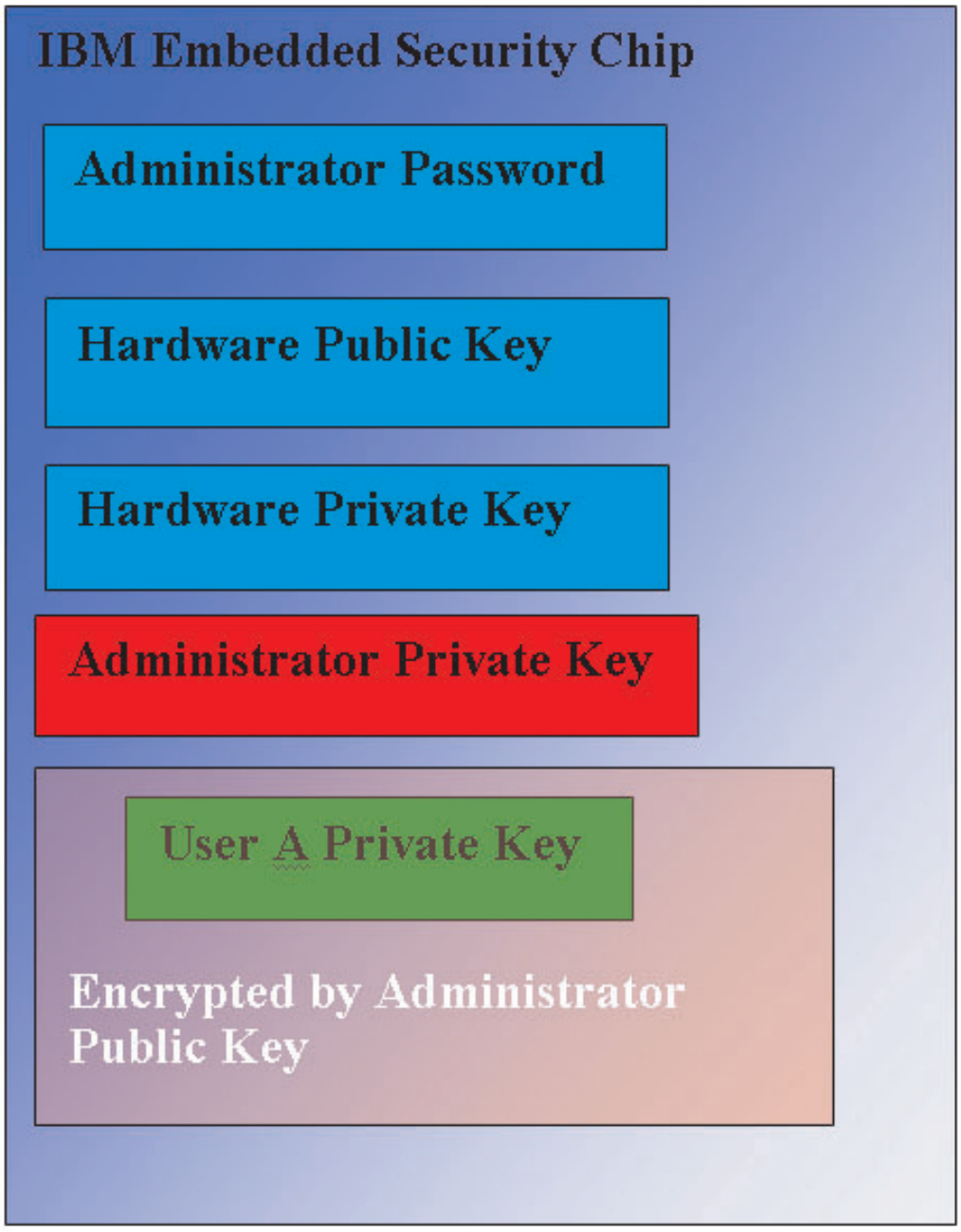

Figure 6. User A's private key, which is encrypted by the Administrator public key, is passed into the security chip.

The Administrator private key is used to decrypt the User A's private key. Now User A's private key is ready for use (See Figure 7).

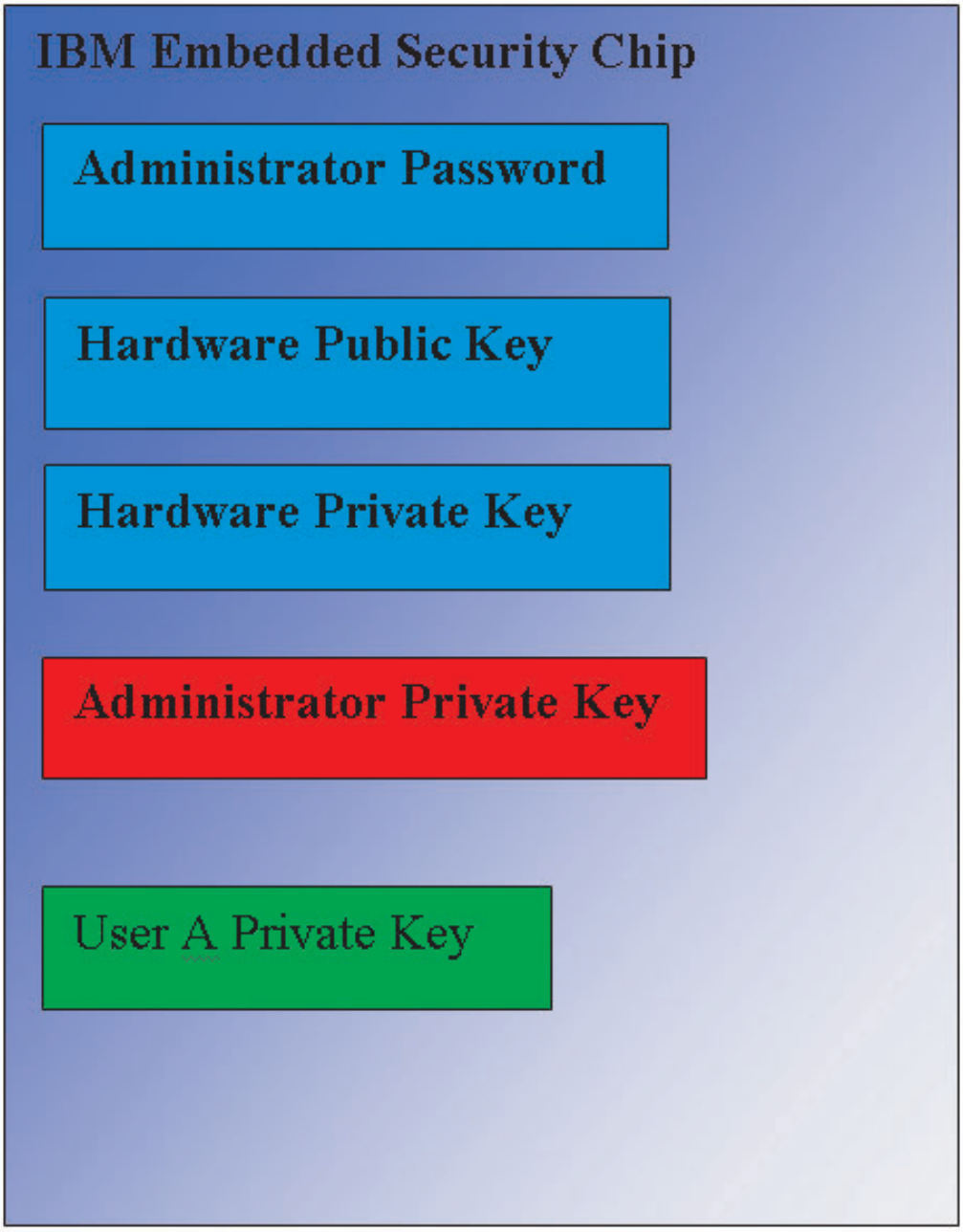

Figure 7. User A's private key is ready for use.

There are several other keys that can be encrypted with the User A's public key. An example would be a private key used for signing e-mail. When User A goes to send a signed e-mail the private key used for signing (encrypted with User A's public key) would be passed into the chip. User A's private key (already in the chip) would decrypt User A's private signing key. Now User A's private signing key is available in the chip to perform the desired operation, in this case creating a digital signature (encrypting a hash). Note the same process of moving keys into and out of the chip would be used when User B logs onto the computer.

Why an administrator key pair?

The main reasons to have an administrator key pair are for archive and restore capabilities. The Administrator key pair serves as an abstraction layer between the chip and the user credentials. The user-specific private key information is encrypted with the Administrator public key as shown in Figure 8.

Important: Develop a strategy for maintaining administrator key pairs. It is possible for each computer with an Embedded Security Chip to have the same administrator key pair, if the IT administrator or security administrator so determines. Alternatively, each department or building can be assigned different administrator key pairs.

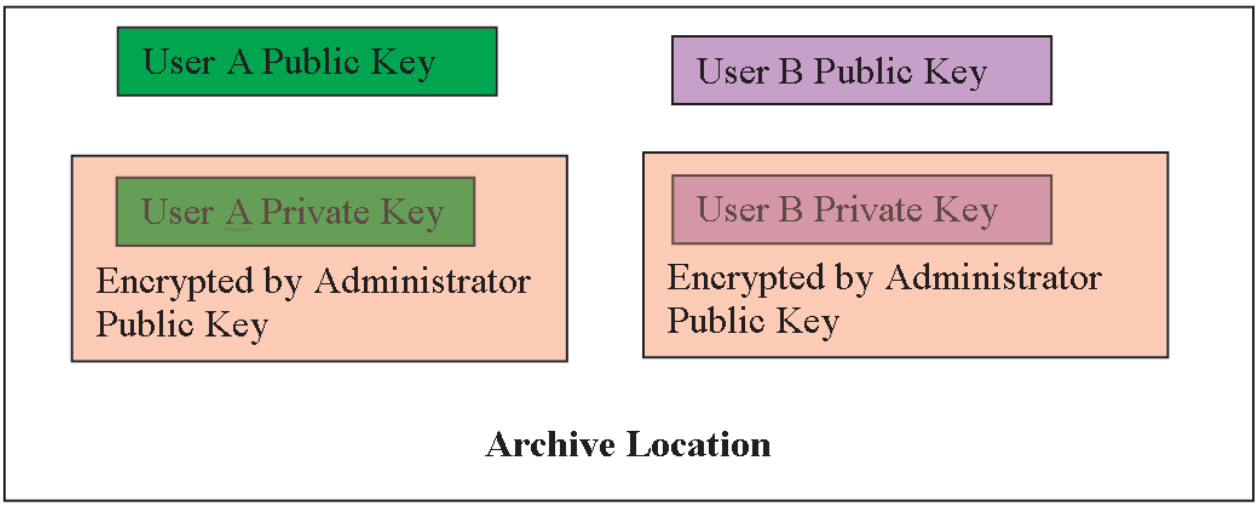

Figure 8. The user-specific private key information is encrypted with the Administrator public key.

Another reason to have an administrator key pair is to sign the client security policy file, thereby preventing anyone except the administrator from changing security policy. In order to achieve a high degree of security for client security policy file, you can split the administrator private key among up to five individuals. In such a case, the five individuals who hold part of the private key, must all be present to sign and encrypt files, such as the client security policy file. This prevents a single individual from unilaterally performing administrator functions. For information about splitting the administrator private key see the Keysplit=1 setting in Table 4 on page 37.

During IBM Client Security Software initialization, administrator key pairs can either be created by the software or can be imported from an external file. If you want to use a common administrator key pair, you will specify the location of the necessary files during client installation.

This user specific information is backed up (written) to an administrator defined archive location as shown in Figure 8. This archive location can be any type of media that is physically or logically connected to the client. The IBM Client Security System installation section will discuss best practices for this archive location.

The Administrator public and private keys are also archived. The user data in the archive location is encrypted with the Administrator public key. Having the user

archive data by itself does you no good if you do not have the Administrator private key to unlock the data. Figure 9 shows the Administrator public and private key archived. This is often referred to in IBM Client Security Software documentation as the "Archive Key Pair." Note that Archive Private Key is not encrypted. Special care must be taken in selecting the archive location for the Archive Key Pair.

Figure 9. The Administrator public and private key are archived.

As mentioned earlier, one of the most important functions of the Administrator public and private keys is for backing up and restoring disk contents. This capability is shown in Figures10–15. The steps are as follows:

1. Client A, for some reason, becomes unusable to User A. In this example, we will say that the computer, Client A, is struck by lightning. See Figure 10.

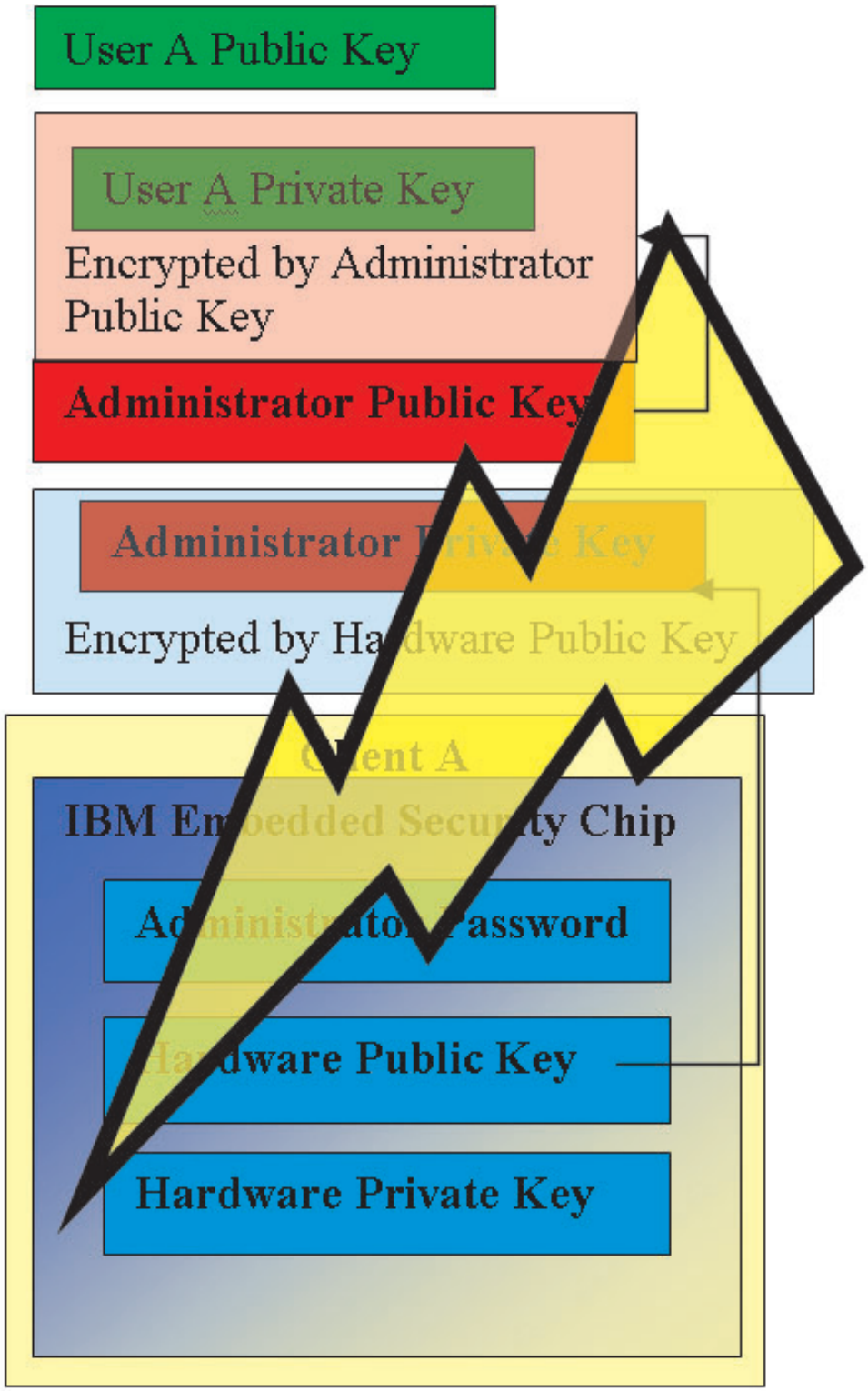

Figure 10. User A's computer is struck by lightning, making it unusable.

2. User A gets a new and improved IBM computer, called Client B. See Figure 11. Client B is different from Client A in that the Hardware public and private keys are different from those of Client A. This difference is visually represented by the gray color keys in Client B and the greenish color keys in Client A.

However, note the Administrator Password is the same in Client B as in Client A.

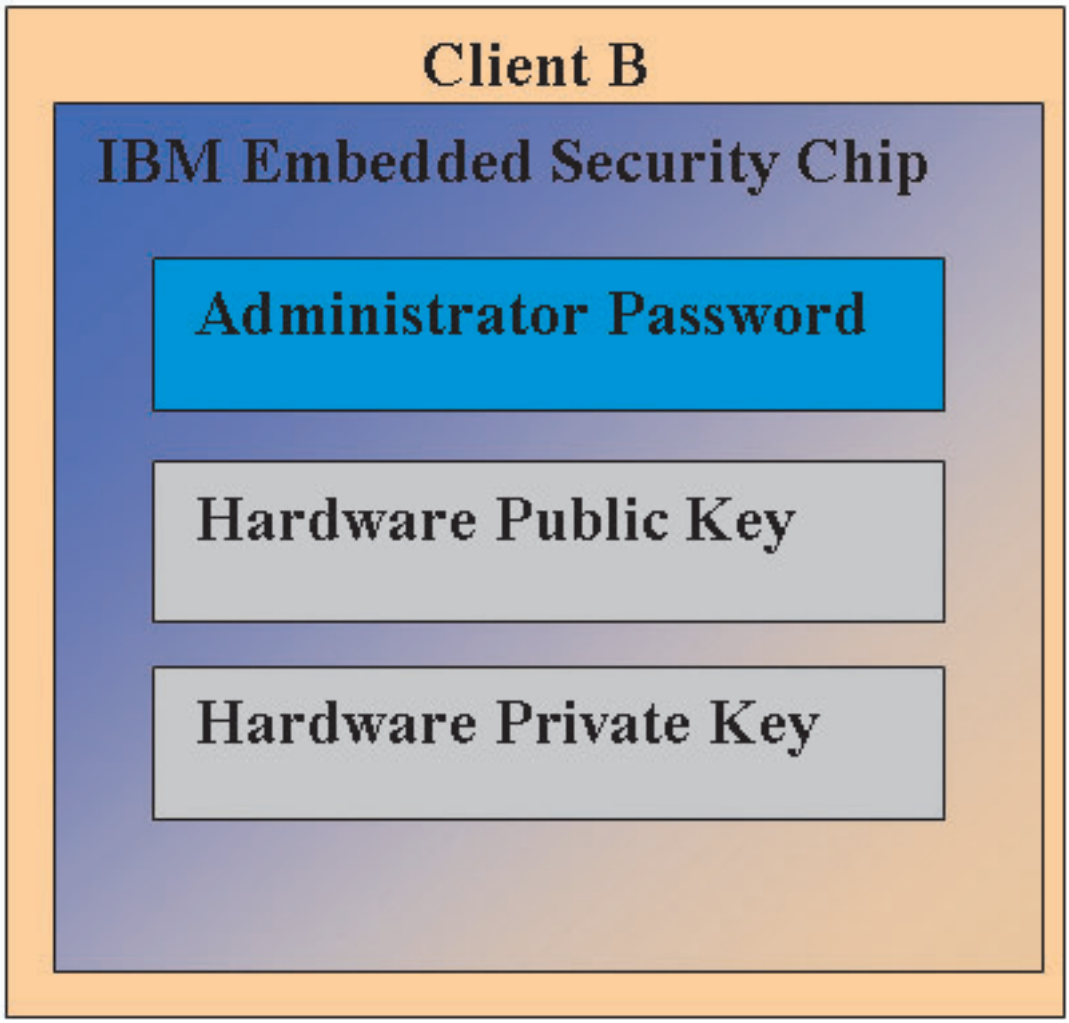

Figure 11. User A receives a new computer, Client B, with a new Embedded Security chip.

3. Client B now needs the same user credentials that were on Client A. This information was archived from Client A. If you look back at Figure 8 on page 12, you will recall that the user keys are encrypted with the Administrator Public Key and stored in the archive location. In order for the user's credentials to be available on Client B, the Administrator public and private keys must be transferred to this machine. Figure 12 shows Client B retrieving the Administrator public and private keys from the archive location.

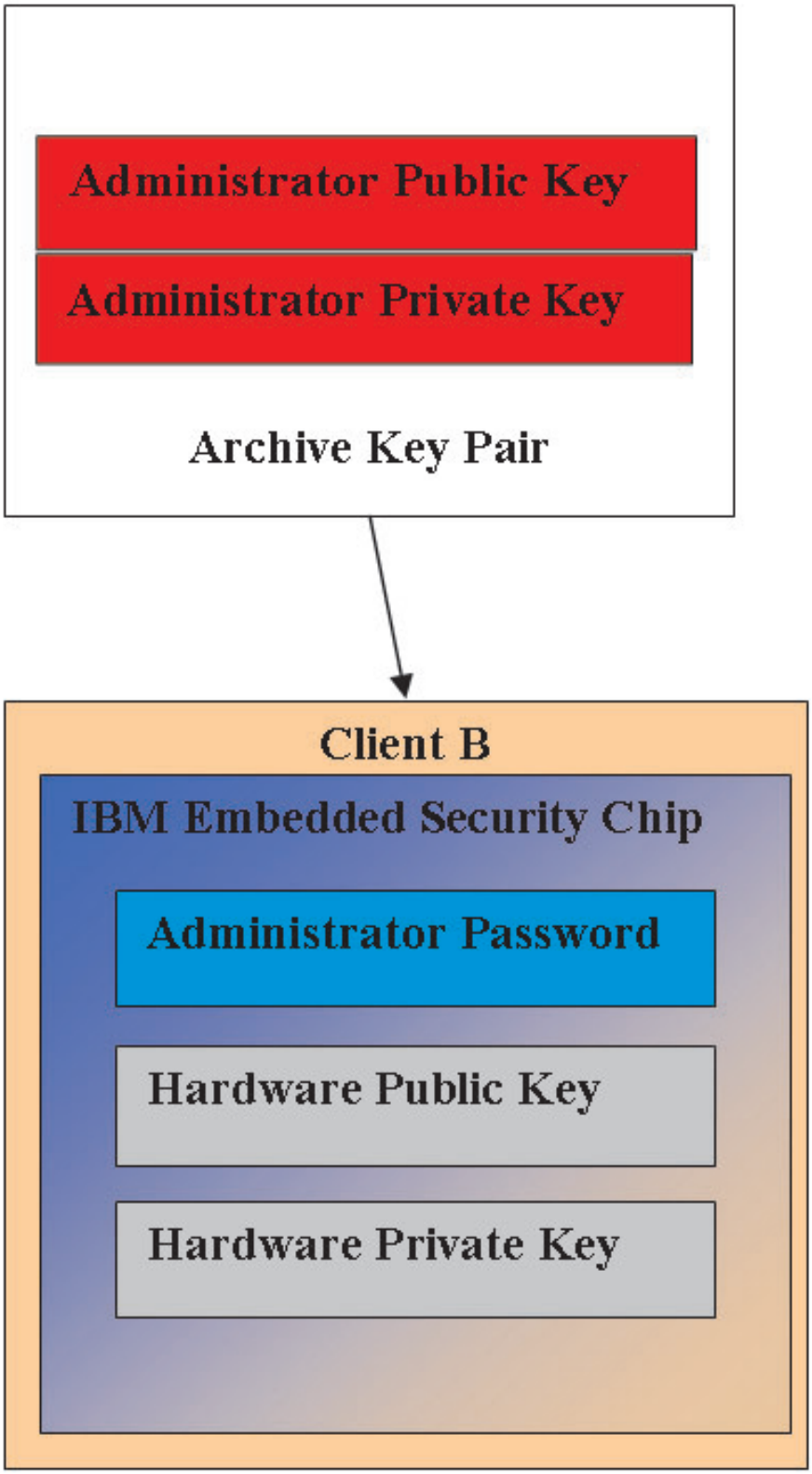

Figure 12. Client B retrieves the Administrator public and private keys from the archive location.

4. Figure 13 shows the Administrator private key being encrypted with Hardware public key of Client B.

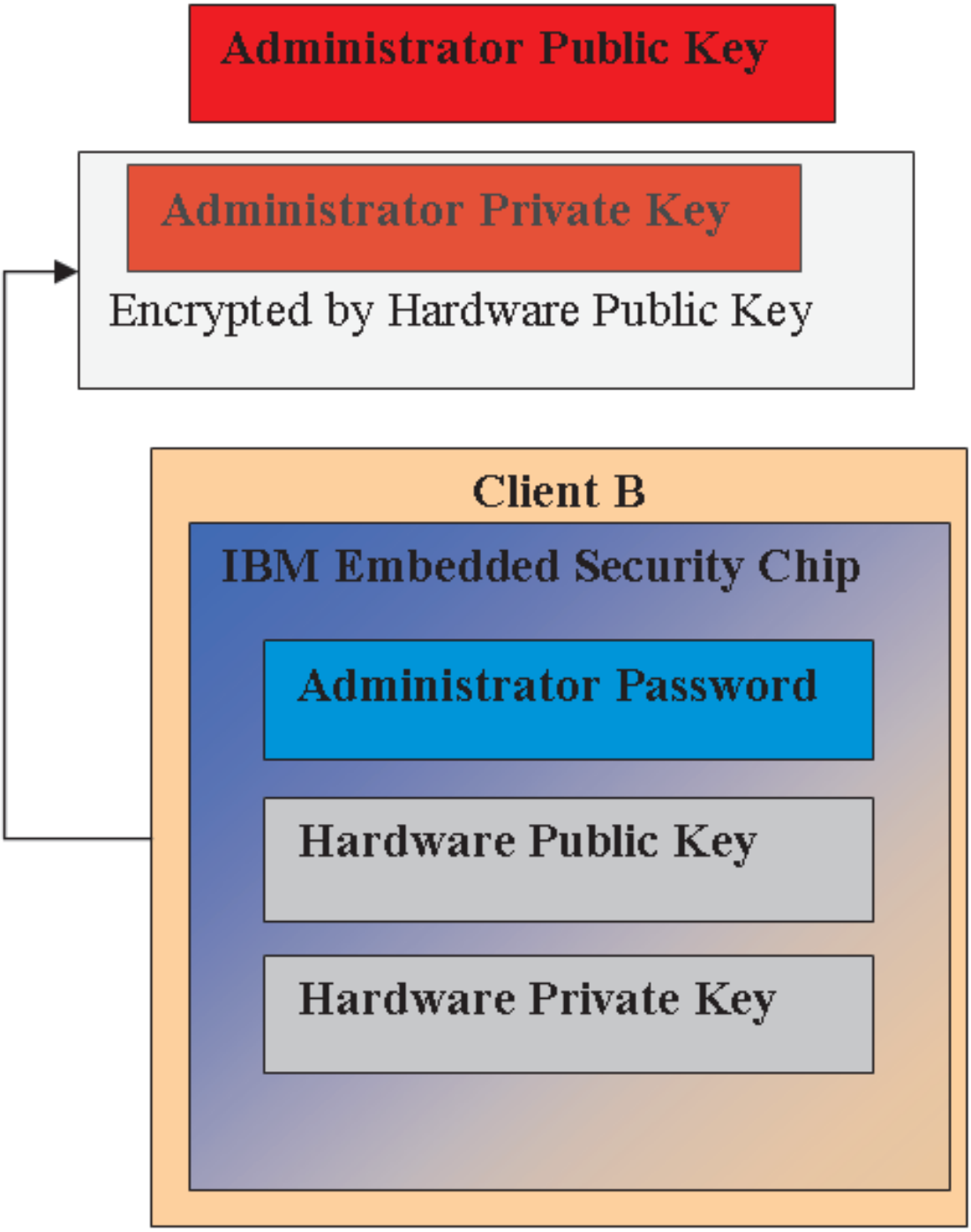

Figure 13. The Administrator private key is encrypted with the Client B hardware key.

Now that the Administrator private key is encrypted with the Hardware public key, the user's credentials can be brought down for User A on Client B. This is shown in Figure 14.

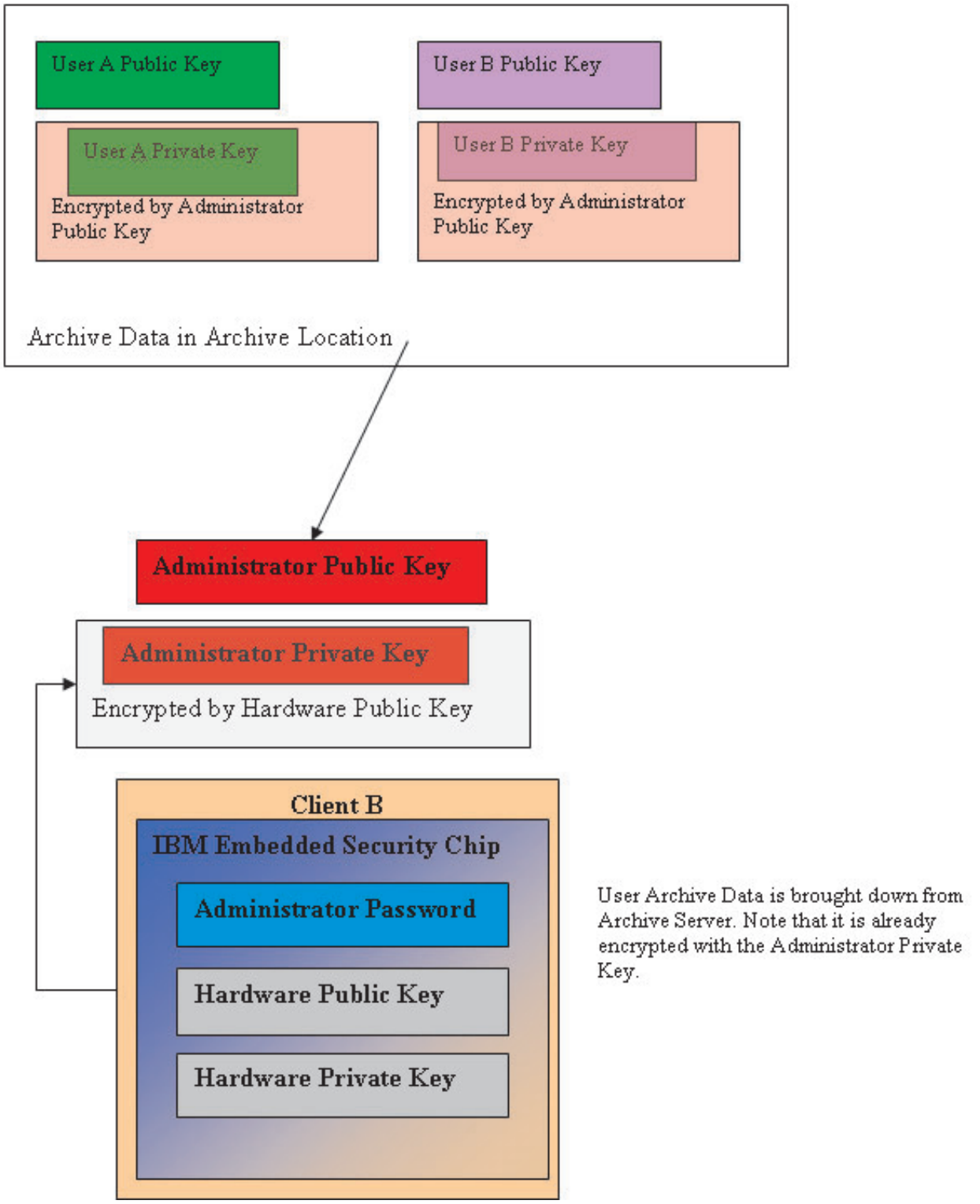

Figure 14. User A's credentials can be loaded on Client B after the Administrator private key has been encrypted.

Figure 15 shows User A fully restored on Client B. Note that User A's private key was encrypted with the Administrator public key while on the archive server. The Administrator public key is a 2048-bit RSA key and is virtually impossible to break. This means the archive location does not necessarily have to be protected or have strong ACL. As long as the Archive key pair (the Administrator public and

private keys) and more specifically the Administrator private key are kept secure the Archive location for user credentials can be essentially anywhere.

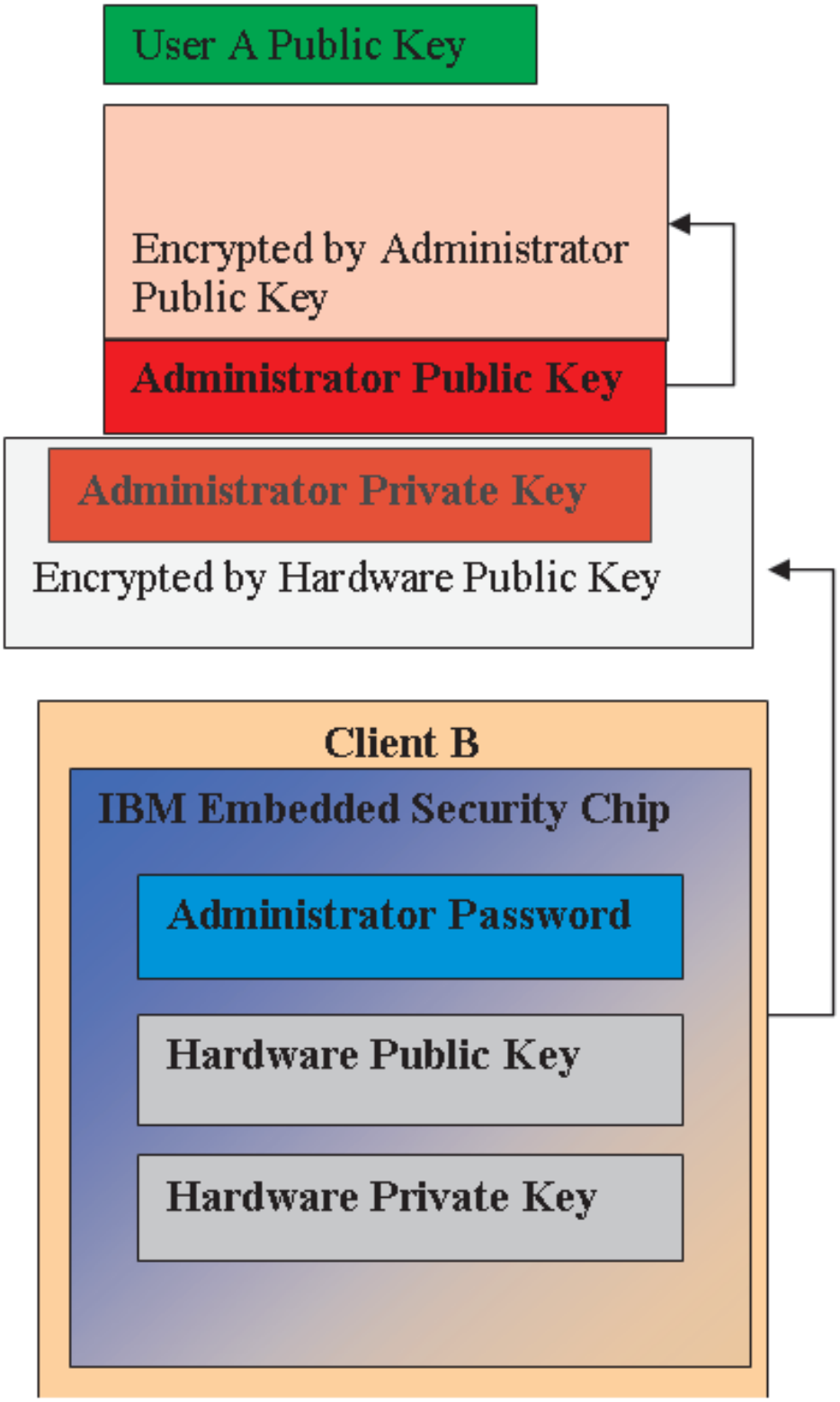

Figure 15. User A is fully restored on Client B.

The details of how the Administrator Password is set, where Archive locations should be, etc. will be discussed in greater detail when we get to the software installation section. Figure 16 shows an overview of the components in an ESS environment. The major points are that each client is unique from a hardware public and private key perspective, but has a common Administrator public and private Key. The Clients have a common archive location but this archive location could be for a segment or group of users.

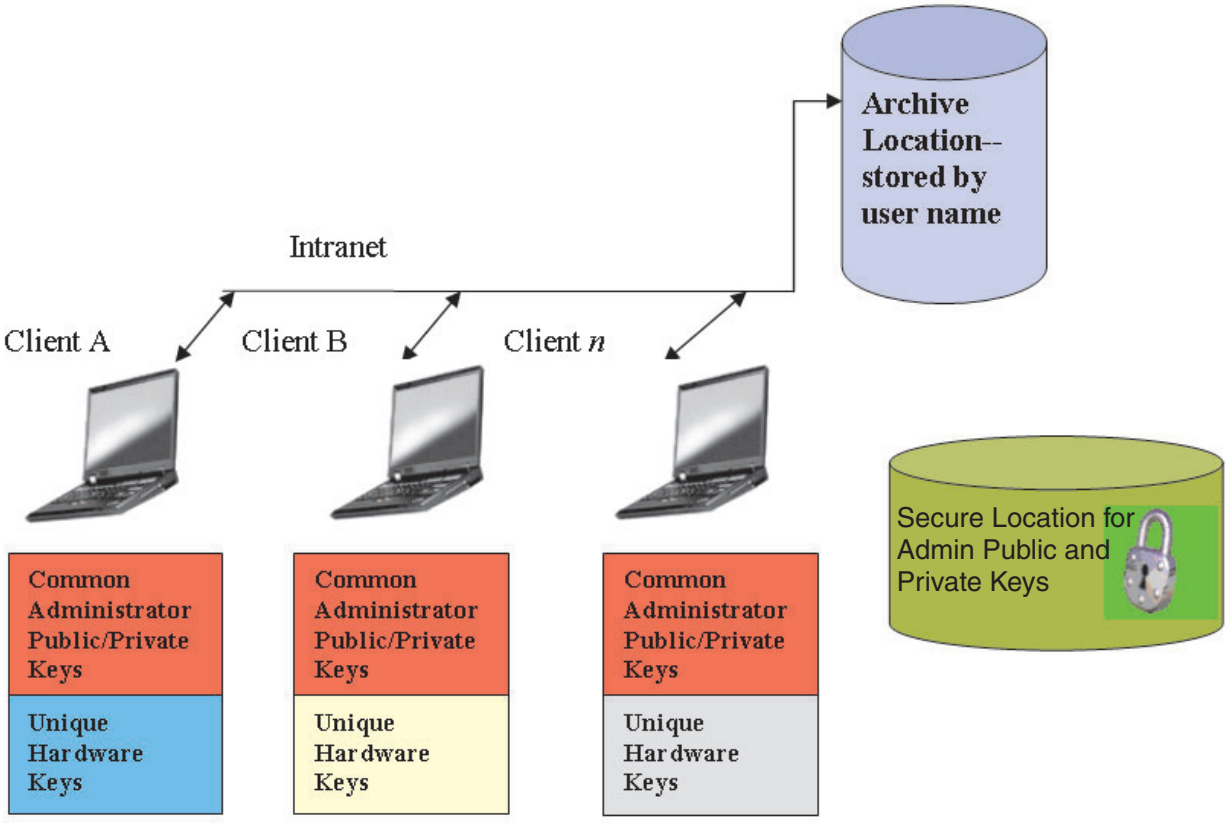

Figure 16. Major components of the IBM Client Security System.

Consider the following example. The Human Resources Department could have a separate archive location from the Engineering department. Archiving is done on a user-name basis. The IBM Client Security Software will archive the users of a system to the defined archive location based on the user name as shown earlier in User A and User B. Also note the secure location for the Admin Public and Private Keys.

Note: Each computer name and user name that will be archived in the same location must be unique. A duplicate computer name or user name will overwrite its first counterpart.

Chapter 4. IBM Client Security Software

The IBM Client Security Software is the ″middleware″ between applications and the IBM Embedded Security chip, as well as the interface to enroll users, set policy, and perform basic administration functions. The IBM Client Security System is essentially composed of the following components:

- Administrator Utility
- User Configuration Utility
- Administrator Console
- Installation Wizard
- User Verification Manager (UVM)
- Cryptographic Service Provider
- PKCS#11 module

The IBM Client Security System enables you to do several of key functions:

- Enroll users
- Set Policy
- Set Passphrase Policy
- Reset forgotten passphrases
- Restore user credentials

As shown in Figure 1 on page 1, user initialization is one of the key aspects of the IBM ESS. IBM ESS manages credentials on a user basis. For example, if User A logs onto the operating system, IBM Client Security System bases all decisions on the assumption that User A is logged on. (**Note:** Security Policy is machine based, not user based; the policy applies to all users of a single computer.) If User A attempts to leverage the IBM Embedded Security Subsystem, the IBM Client Security System will enforce security policies as set for User A on that computer, such as passphrase or fingerprint authentication. If the person logged on as User A cannot supply the correct passphrase or the correct fingerprint for authentication, IBM ESS will prohibit the user from performing the requested action.

Enrolling users and managing enrollment

IBM ESS users are simply Windows® users who are enrolled in the IBM ESS environment. There are several ways users can enroll, which will be covered in detail later in this document. In this section, we will cover what happens when a user enrolls. Understanding what happens during this process will give you a better understanding of how IBM ESS works and ultimately how to successfully manage this in your environment.

Each user in the IBM ESS environment has at least one personalization object associated with him or her that is used for authentication purposes. The minimum requirement is a passphrase. Every user in the UVM component of ESS (from the user perspective, UVM manages authentication and enforces security policy) environment must have a passphrase and this passphrase must be given a minimum of once per computer start-up. The following sections will explain why a passphrase is used, how to set one up, and how to use it.

Requiring a passphrase

Simply put, a passphrase is required for security purposes. Having a hardware element such as the IBM Embedded Security Subsystem is a tremendous benefit because it provides a secure, autonomous location for a user's credentials to be operated upon. However, the protection that a hardware chip provides is of little use if the authentication required to access the chip is weak. For example, consider that you have a hardware chip that performs security functions. However, the authentication required to invoke an action by the chip is a single digit. This leaves a potential hacker the choice of guessing a single numerical digit (0 though 9) to invoke actions with your credentials. The single-digit authentication weakens the security of the chip such that it provides little or no added benefit over a software-based solution. If you don't have strong authentication in conjunction with the hardware protection, you could have no security gain at all. The passphrase required by IBM ESS is used to authenticate a user before any actions take place with the user's credentials in the hardware.

Setting up a passphrase

All users have passphrases associated with their credentials. In Figure 3 on page 5, you saw that a user's private key is encrypted with the administrator public key. The user's private key also has an associated passphrase. This passphrase is used to authenticate the user with his or her credentials. Figure 17 shows the passphrase plus the private key component encrypted with the administrator public key.

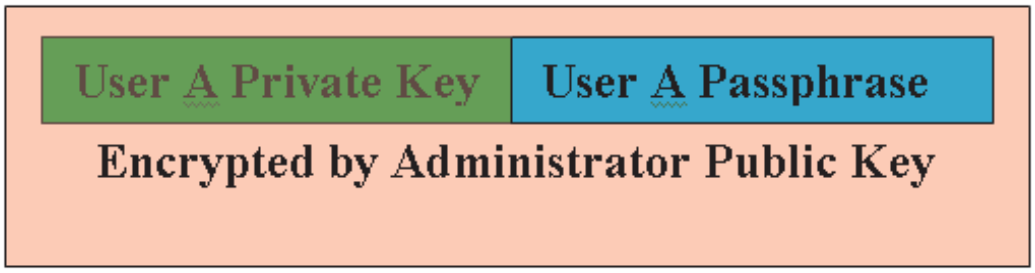

Figure 17. User A must provide the passphrase in order to perform any functions that require User A's private key.

The passphrase depicted in Figure 17 is created when a user is enrolled into UVM. Once again how this actually happens when rolling out IBM Client Security Software will be covered later in this document.

User A's private key is encrypted with the administrator public key, because decrypting the private key requires the administrator's private key. Therefore, if User A's passphrase is forgotten, the administrator can reset a new passphrase.

Using a passphrase

Figure 18 on page 23 through Figure 20 on page 25, we show how the user passphrase is actually used. (**Note:** A passphrase must always be used first and at least once per session. A passphrase is always required. You can choose to add additional authentication devices, but none of these can replace the initial user passphrase requirement. Briefly, the biometric or other authentication data are encrypted with the user's public key. Access to the private key is required to decrypt this additional security data. Therefore, the providing the passphrase at least once per session is required to decrypt the additional data.)

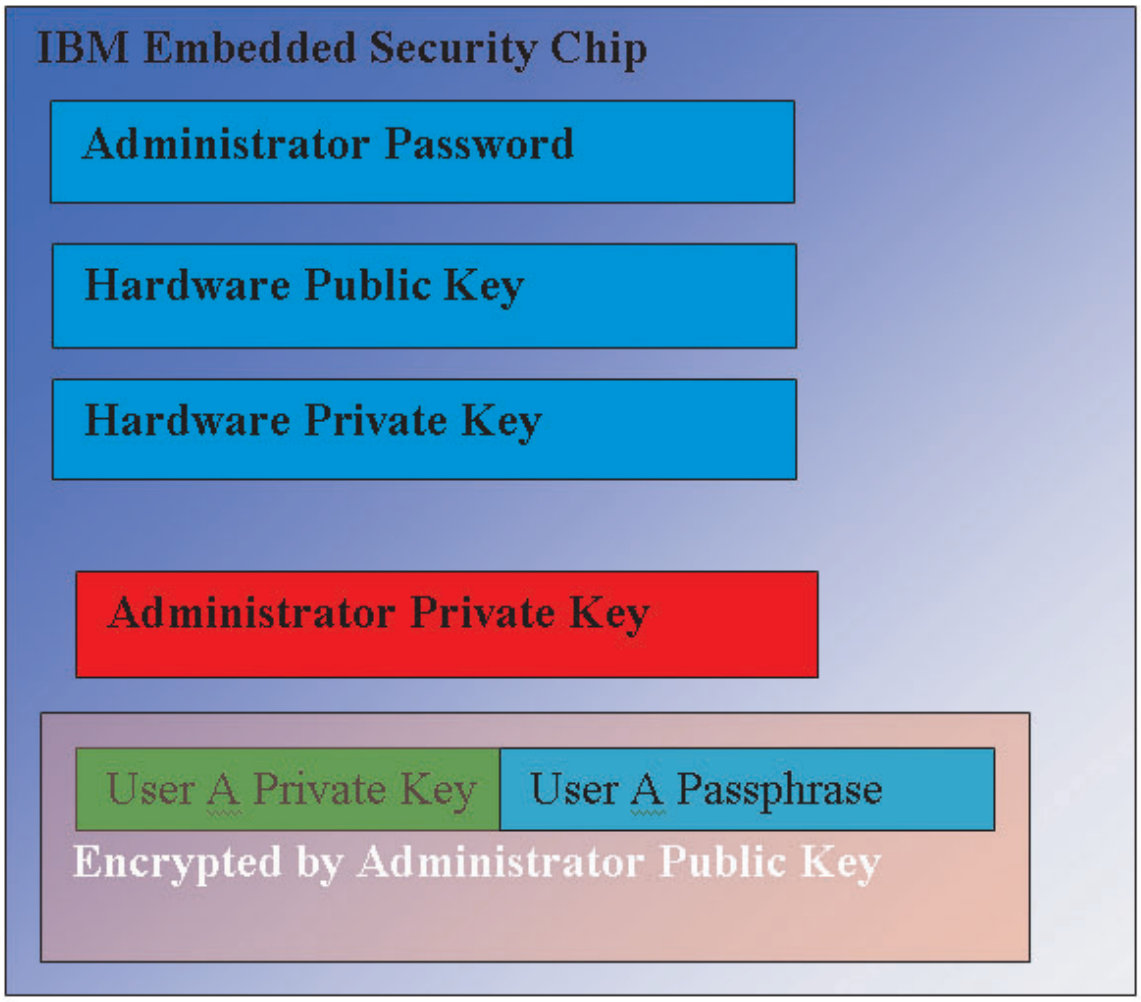

Figure 18. The Administrator's private key is decrypted in the chip.

The data that constitute User A's Private Key and User A's Passphrase encrypted with the Administrator Public key is passed into the IBM Embedded Security Chip. The Administrator's private key is already decrypted in the chip as described earlier. The data are passed in as described in Figure 19 on page 24.

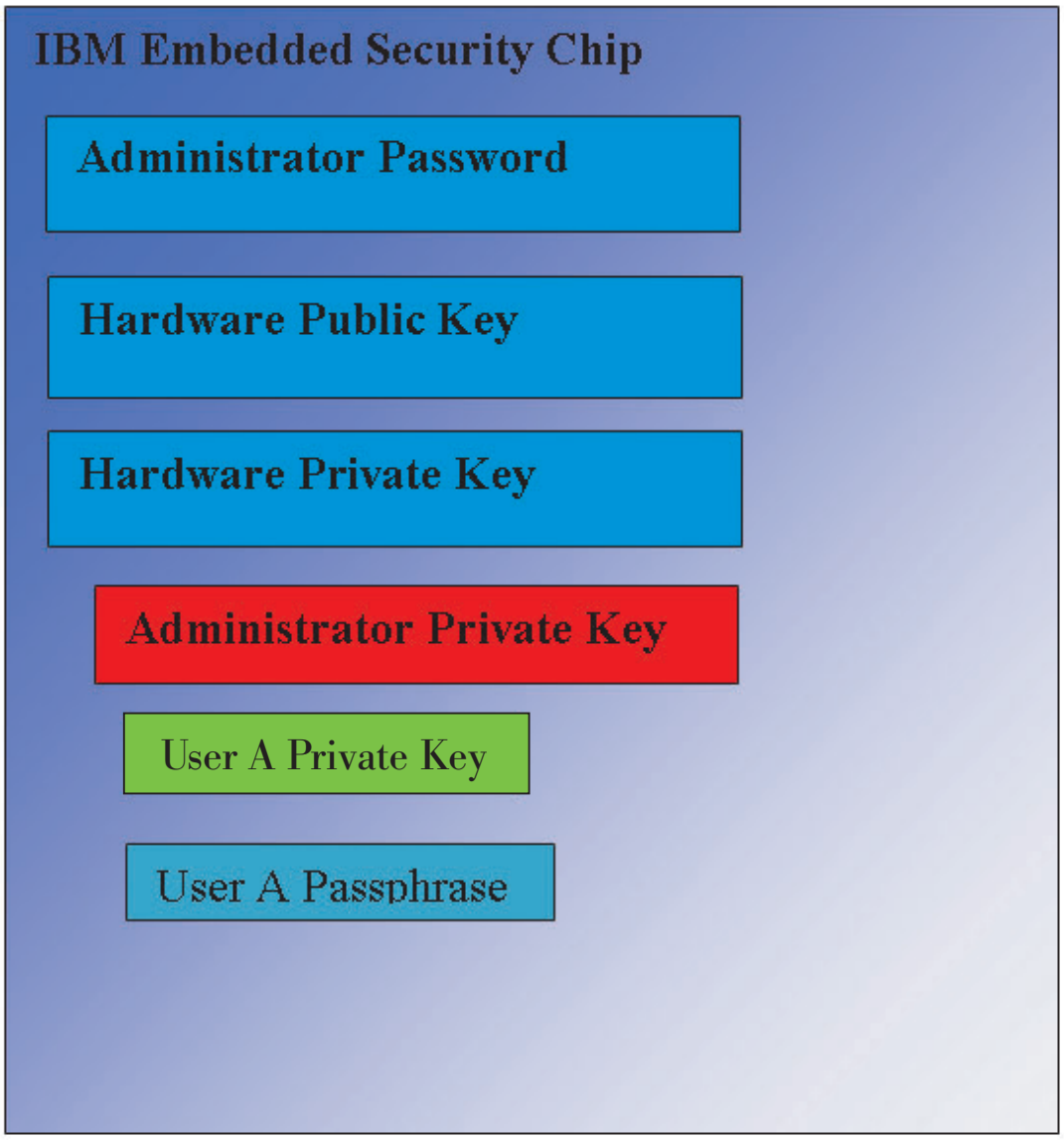

Figure 19. User A's Private Key as well as User A's passphrase are available in the chip.

The data are decrypted, making User A's Private Key as well as User A's passphrase available in the chip. When the currently logged-in user, identified by the IBM Client Security System as User A, attempts to use the credentials of User A then a passphrase dialog will open. (See Figure 20 on page 25.)

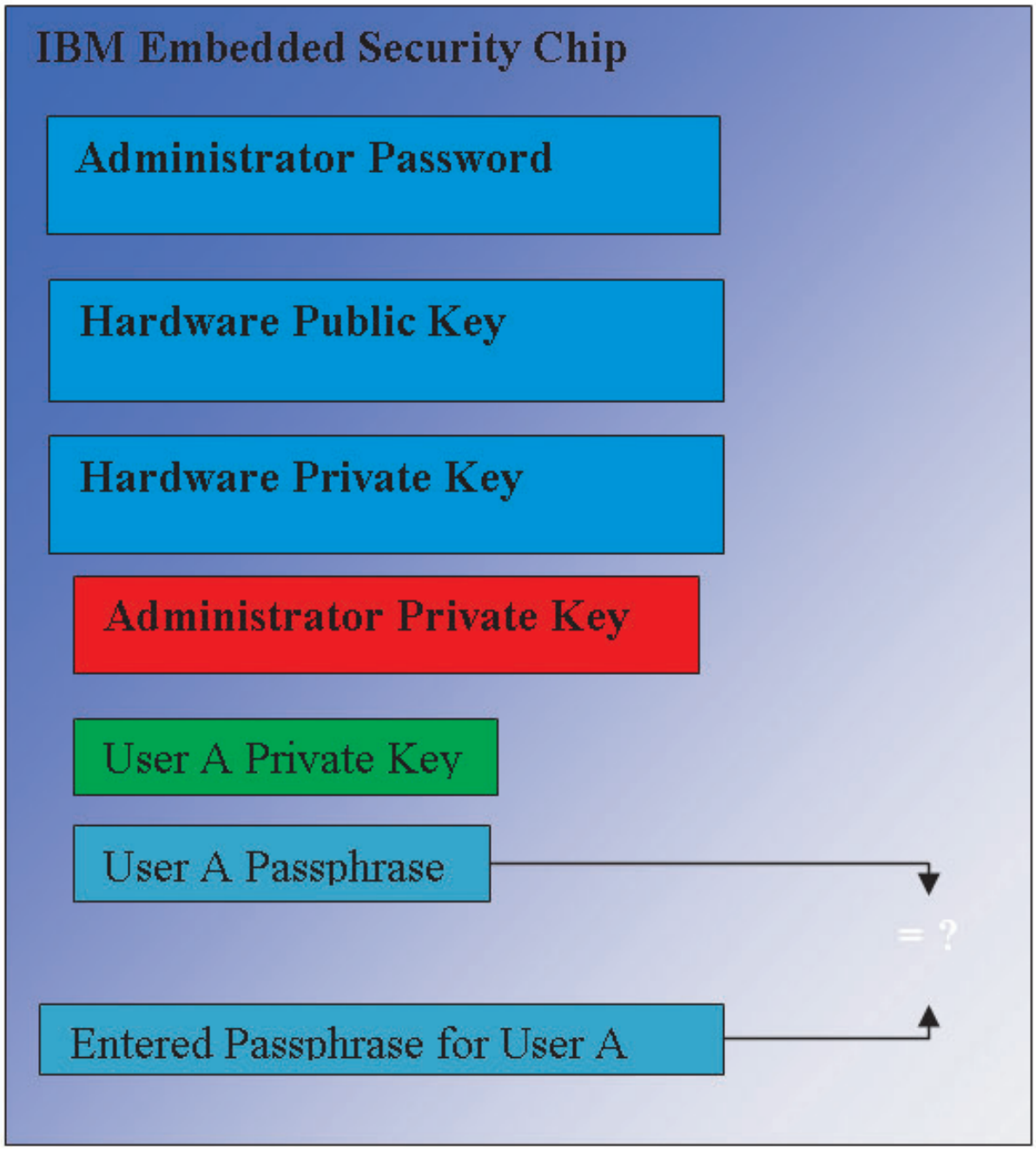

Figure 20. When User A attempts to use the credentials of User A then a passphrase dialog will open.

The typed passphrase is passed to the chip and compared to the decrypted passphrase value. If they match, then the credentials of User A can be used for various functions such as digital signatures or decrypting e-mails. Note that this passphrase comparison is done in the secure environment of the chip. The chip has anti-hammering capabilities to detect repeated failed access attempts. Also note that User A's registered passphrase is never exposed outside of the chip. As part of the IBM Client Security Software installation, users are enrolled. Part of this enrollment process is the creation of the user's passphrase. We will discuss the details of how this passphrase is set and how passphrase rules can be enforced.

Figure 1 on page 1 showed the IBM Embedded Security Chip as well as the IBM Client Security System. Figure 1 on page 1 also depicts Company initialization and user initialization. Company initialization is associated with the Embedded Security Subsystem and user Initialization is associated with the IBM Client Security Software. The previous sections described the initialization that takes place to offer understanding of the general concept. The following sections will give more details on the process of initialization.

Company Initialization

Company initialization is essentially the process of adding hardware public and private keys and a Administrator password. This process takes a generic machine as shipped from IBM and makes it unique for your enterprise. The following chart will show the methods for the initialization of public and private keys as well as Administrator passwords.

Table 1. Hardware initialization methods

Action	Can be created in BIOS	Can be created Manually by Administrator in CSS software	Can be created in a Script
Hardware Public/Private Key Creation	No	Yes	Yes
Administrator Password Creation	On some TCPA-compatible clients, yes. Check for BIOS entry.	Yes	Yes

This chart demonstrates that the Hardware Public and Private keys are not created automatically when the software is installed. The Hardware Public and Private Key creation must be initiated manually in the software or by script. The Administrator Password can be created in BIOS, the IBM Client Security Software application, or by script. The chip controls the values set for the hardware public and private keys; you cannot set the values. Random-number capability in the chip is used to produce statistically random Public and Private key pairs. However, you do set the Administrator Password.

The administrator password, however, is different because the administrator must set this value. Several issues regarding the administrator password must be addressed:

- What will you set as the administrator password or passwords?
- Will you have more than one for various groups? If so, how will you logically make the determination of which computers have which password?
- Which administrator will have access to the password? If you have more than one password for separate groups of users, who will have access to which passwords?
- Will self-administered end users have access to the administrator password?

To make an effective decision regarding the items above, it is important to understand what the administrator password enables you to do:

- Gain access to administrator utilities
- Add/remove users
- Define which IBM Client Security Software application/features can be used

There are also cases where the administrator password and access to the administrator key provide other capabilities. They are as follows:

- Define/change policy
- Create file to reset user's passphrase

Subsequent sections will explain the connection between the policy file and the administrator private key. Note for now that the administrator private key is required to change policy. Table 2 summarizes the abilities of having the administrator password and/or the administrator private key.

Table 2. Administrator actions based password and private key

Action	Administrator password	Administrator private key
Gain Access to Admin Utility	Yes	No
Add/Remove/Restore users	Yes	No
Define which CSS Application/features can be used	Yes	No
Define/Change policy	Yes	Yes
Create file to reset user's passphrase	Yes	Yes

Company initialization also refers to the Administrator public and private key. From the chart above you can see the capabilities associated with this key. Give some thought to setting the Administrator public and private keys. This key pair can be unique for each computer or it can be the same for all machines. When the IBM Client Security Software is initialized the administrator will have the choice of using an existing key pair or creating a new key pair for the client. Once again, the usage model will determine what is best for your enterprise.

Best Practices

Large enterprises report success with establishing administrator private keys as well as administrator passwords that associated with either a logical or physical differentiation of employee computers. For example, set an administrator password and/or administrator private key for all computers used in the human resources department, another for the engineering department, etc. You can also differentiate on a physical basis, such as by building or site location. Being able to determine which administrator private key to use when creating a passphrase reset file should be an easy process based upon who is requesting the reset. As Table 1 on page 26 and Table 3 on page 29 indicate, user and company, or hardware, initialization must also take place.

User initialization

The IBM ESS provides the ability for several users to carry out independent and secure transactions on a single computer. These users must have a passphrase associated with them and can have other authentication elements, such as fingerprints and/or smartcards. User initialization is a critical step in configuring client computers to use the IBM ESS. Note that user initialization is a two-part process:

1. Registration
2. Personalization

Registration

Registration is simply adding a user to, or registering a user with, the IBM Client Security System. In Figure 21, you can see the User Verification Manger (UVM) component of the IBM Client Security Software. UVM controls each user's credentials as well as enforces policy.

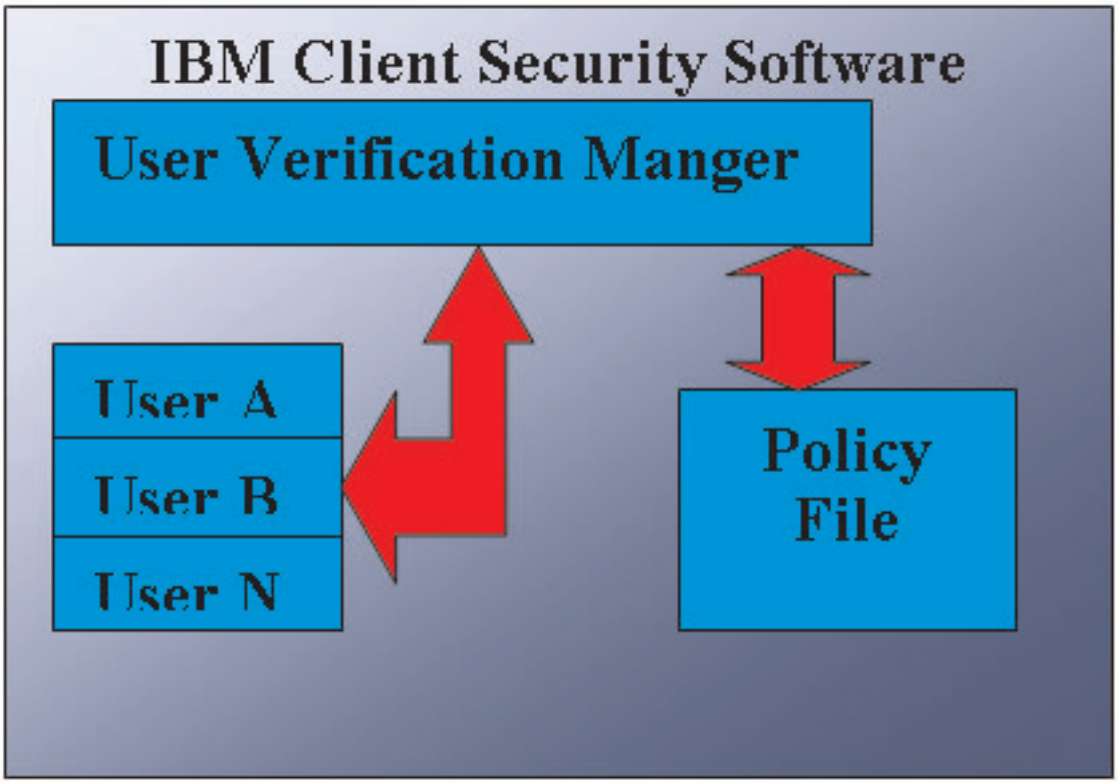

Figure 21. User Verification Manager controls each user's credentials and enforces security policies.

A policy file, such as that depicted in the diagram, contains the authentication requirements for each user that UVM manages. Note that UVM users are simply Windows users (local or Domain). UVM manages credentials based on who is currently logged onto the computer and operating system. For example, if User A logs into Windows and User A is also part of UVM, then UVM will enforce the policy when User A attempts to perform operations that require credentials. For example, User A logs onto the computer. User A then goes into Microsoft® Outlook and sends a digitally signed e-mail. The private key used to send that digitally signed e-mail is protected in the IBM Embedded Security Subsystem. Before UVM will permit that operation to be carried out, it will enforce policy as defined in the policy file. In this example, the requirement is for a passphrase to be authenticated before the operation is carried out. UVM will prompt the user for the passphrase and if it is verified correctly the private key operation will be carried out in the chip.

Personalization

Personalization is simply setting an individual's personal UVM passphrase. A distinction is made because different people can perform the distinct parts of the process. The individual's UVM passphrase should be known to the individual only. However, if each individual does not perform the initialization process that person might need to perform an additional step.

For example, User A is initialized by the IT administrator. The IT Administrator selects User A from a Windows list of users (from a domain, for example). UVM asks for the UVM passphrase to be associated with User A. The IT administrator enters a ″default value″ of ″IT Admin Passphrase.″ To ensure security of the system, after User A receives the system he or she must customize the passphrase so no one could conduct secure transactions using the default passphrase.

Note: Personalization is easily accomplished through the IBM Client Utility, but it is important to note this distinction when you are deploying the IBM ESS to your users.

Table 3. User initialization methods

Method	Command Process	Process Requirements
Manual	The Admin can manually personalize CSS for the user through the Admin Utility	Administrator must be present at each computer for setup.
Administrator configuration file	The Admin can create a configuration file, which contains an encrypted version of the Administrator Password. That file is sent to the user, who can then enroll individually without administrator intervention or presence.	User goes through setup process.
*.ini	The administrator creates a script that executes the.ini file and places a default or personalized password.	Admin or user presence optional.

Deployment scenarios

You are deploying 1 000 clients to 1 000 end users. One of the following might describe your approach to deployment:

- You know exactly which machine is going to which end user. For example, you know machine 1 is going to Bob, so you register Bob on machine 1. Bob must personalize (set his individual passphrase) when he receives the computer. Bob receives the computer, starts IBM Client Security Software, and then sets his passphrase.
- You do not know which machine is going to which user. You take client 1 and ship to end user X.

These two variable factors make deploying the IBM ESS different from deploying a typical application. However, there are several deployment options that provide flexibility in deploying IBM's ESS.

A typical flow diagram of PC delivery in your company may look like the following:

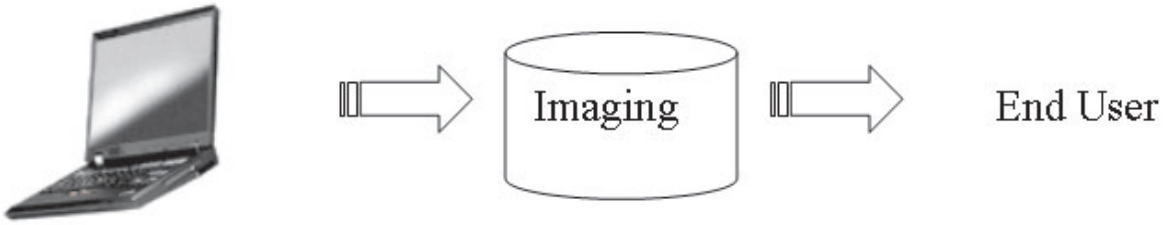

Figure 22. Typical PC deployment flow diagram

Six deployment scenarios

There are six deployment methods for IBM Client Security Software:

1. **Added component**—IBM Client Security Software code is not part of the disk image. It is installed, initialized, and personalized after computers have been deployed.
2. **Image component**—IBM Client Security Software code is part of the image, but is not installed. Neither company personalization nor user personalization has been initiated. (See Figure 23 on page 31.)

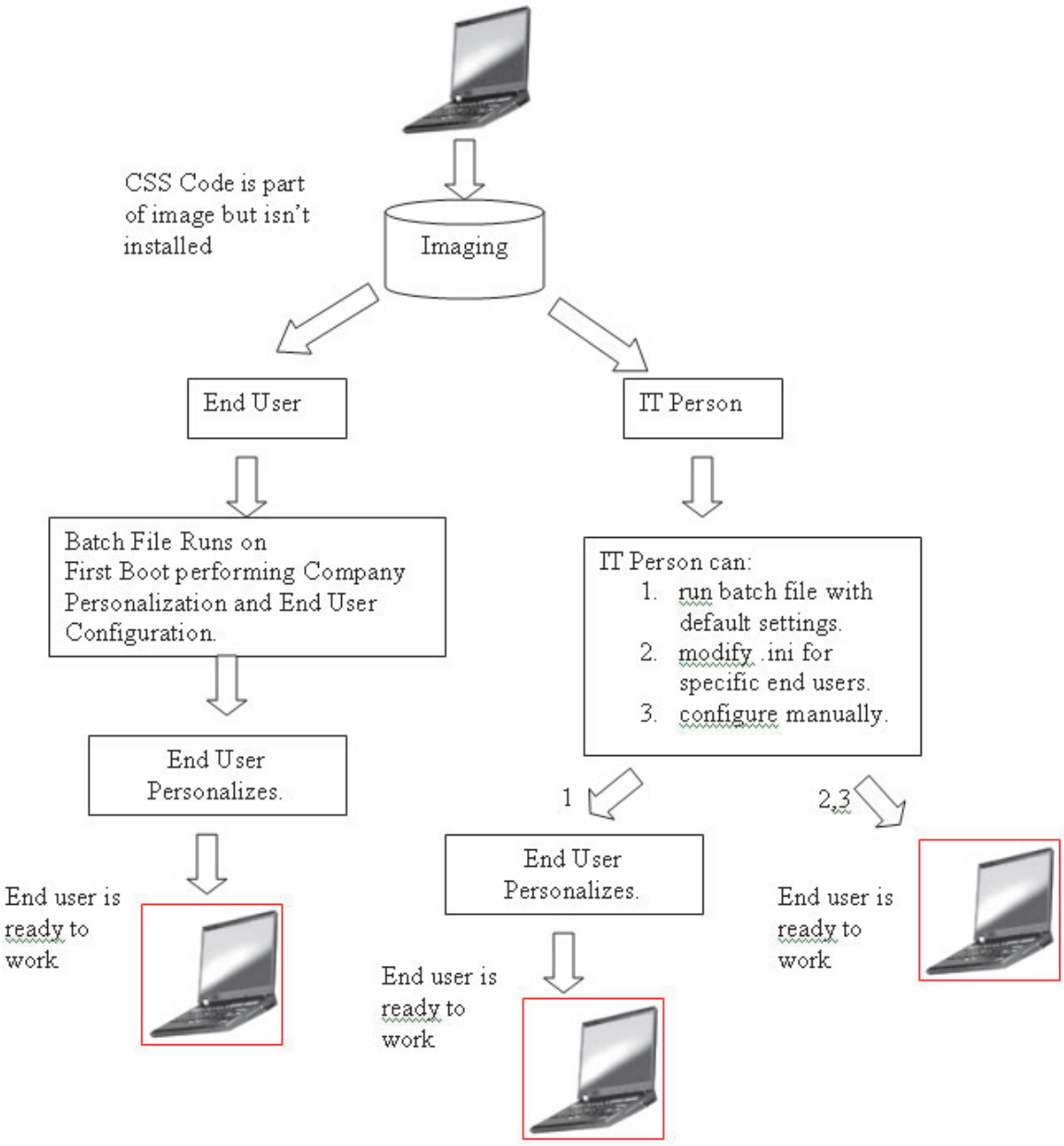

Figure 23. IBM Client Security Software code is part of the image, but is not installed.

3. **Simple installation**—IBM Client Security Software is installed and has been personalized for the company or the end user. (See Figure 24 on page 32.)

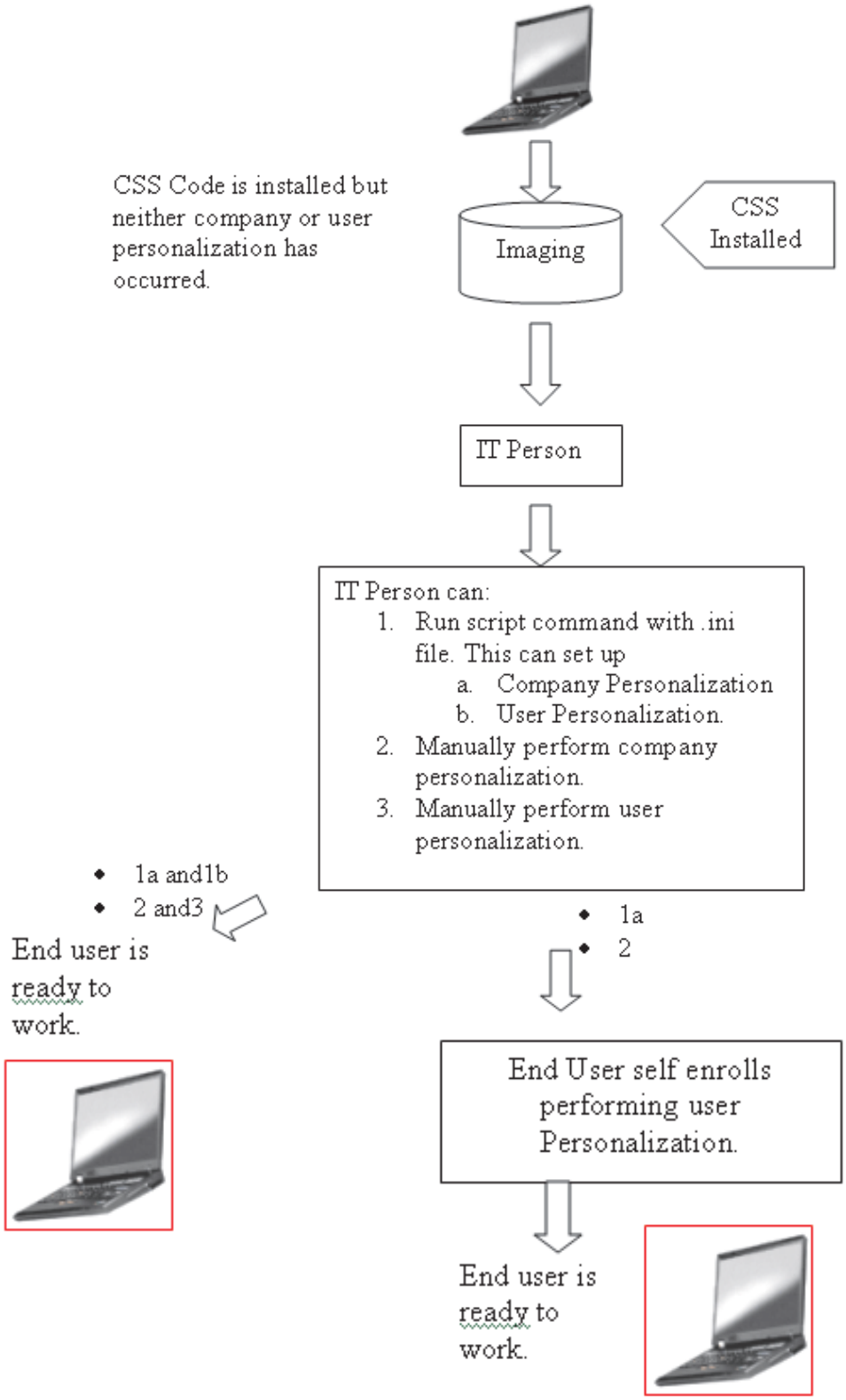

Figure 24. IBM Client Security Software Code is installed but neither company or user personalization has occurred.

4. **Partial personalization**—IBM Client Security Software is installed and company personalization has occurred, but end user personalization has not occurred. (See Figure 24.)

5. **Temporary personalization**—IBM Client Security Software is installed and both company and user personalization has been set. The user will need to reset the user passphrase and, if required, provide other authentication information, such as fingerprint scans or smartcard association. (See Figure 25 on page 34.)

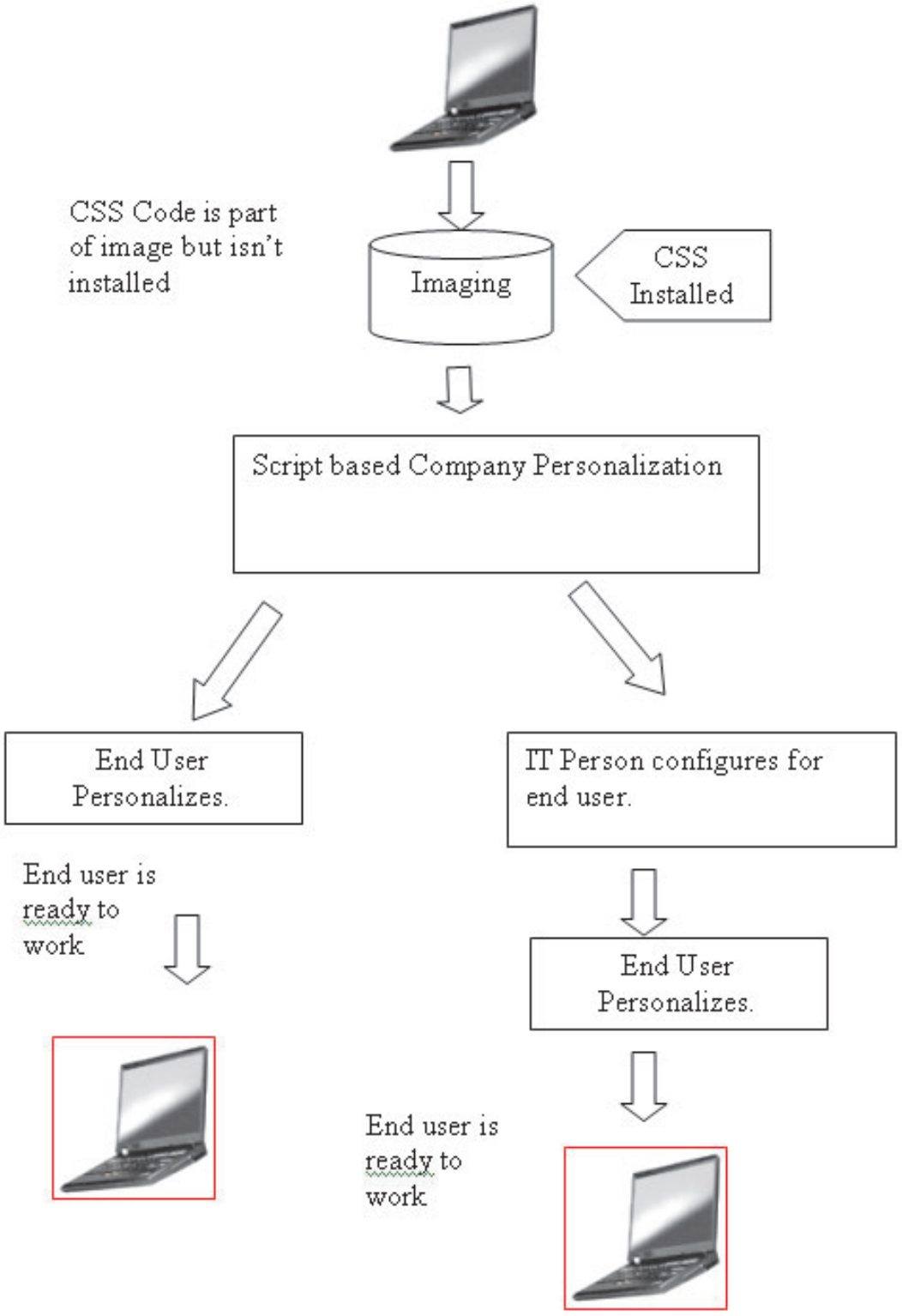

Figure 25. IBM Client Security Software is installed and both company and user personalization has been set.

6. **Full personalization**—IBM Client Security Software is installed and both company and user personalization has been set. The administrator sets the user passphrase. If a fingerprint scan or other authentication is required, the user must provide that personalization. (See Figure 25.)

In scenario 1, IBM Client Security Software is deployed after the disk image is placed on the computer. The IBM Client Security Software installed and configured and the Embedded Security Chip is configured after the disk image has been installed.

Scenarios 2-6 represent various options of software deployment and configuration and chip configuration. Depending on your needs and your environment, you can select the scenario and the installation method the best meets your requirements. See ″Installation and initialization″ for further information about installation methods.

Installation and initialization

The IBM Client Security Software is unique because there is a hardware component that must be personalized for each computer.

The IBM Client Security Software installation can be divided into two processes: installation and initialization. The installation process is similar to installing typical software. This installation can be accomplished by two methods:

1. Client Security Software is added to deployed computers. (See scenario 1 on page 30.)
2. Client Security Software is part of the base image. (See scenarios 2 on page 30 through scenario6 on page 34.)

In method 1 the IBM Client Security Software is added to an image that is added to every computer by programs such as Ghost or IBM's ImageUltra™ Builder.

In method 2, IBM Client Security Software is added to an end user's PC after the computer with the base image has been deployed. Method 2 can be accomplished two ways:

1. **User directed**—The starts and completes installation, clicking dialogs and providing all required user input.
2. **Silent install**—The installation process can be started remotely and completed unattended without user involvement.

There are two modes of initialization:

1. Mass configuration
2. Individual configuration

In the mass configuration option, a CSS.ini file must be used. This file provides parameters for options such as enrolling all users on a system and giving all of those users a set passphrase. In the individual configuration the end user can be given a file that enables self-enrollment and user defined passwords.

Adding IBM Client Security Software to deployed computers with the security chip

The administrator can deploy IBM Client Security Software (on base image) only (without personalization or configuration) and then configure on the clients. Alternatively, the administrator can mass deploy IBM Client Security Software, and then mass configure automatically. In either case, first install software then configure.

Installing IBM Client Security Software: To add IBM Client Security Software to the base image, the following components must be included:

1. Drivers: LPC and SMBus

2. IBM Client Security Software Code
3. Administrator password and private key defined
4. Install IBM Client Security Software applets (File and Folder Encryption and Password Manager, if required in the policy file must be installed. See the *IBM Client Security Administrator's Guide* for installing silently these applets)

After the three components listed above are added to the donor system, the Embedded Security Subsystem hardware-the security chip-must be initialized. To initiate a mass installation, complete the following procedure:

1. Create the CSEC.INI file. (You can create the CSEC.INI file, using the Client Security Wizard: CSECWIZ.EXE in Security directory. After completing the wizard, mark the check box beside **Save settings, but do not configure subsystem. (Settings will be saved in C:\CSEC.INI)**.
2. Extract the contents of the IBM Client Security Software installation package (csecxxxxx_00xx.exe) with Winzip using folder names.
3. Edit the szIniPath and szDir entries, which are required for a mass configuration, in the SETUP.ISS file. The szIniPath parameter is required for mass configuration. (See complete SETUP.ISS file below.)
4. Copy the files to the target system.
5. Create the \setup -s command-line statement. Run the command-line statement from the desktop of a user who has administrator rights. The Startup program group or the Run key is a good place to do this.
6. 6. Remove the command-line statement on the next boot.

The full contents of the setup.iss file is listed below with several descriptions:

```
[InstallShield Silent] Version=v6.00.000 File=Response File szIniPath=d:\csec.ini
```

(The above parameter is the name and location of the .ini file, which is required for mass configuration. If location of the .ini file is on a network drive, it must be mapped. If you are performing a silent installation that is not part of a mass configuration, remove this entry. If you want to install IBM Client Security Software only, delete szIniPath=d:\csec.ini from the above line of code. If you want to install and configure, leave that command in place and verify the path.)

```
[FileTransfer] OverwrittenReadOnly=NoToAll [{7BD2CFF6-B037-47D6-A76BD941EE13AD96}-
DlgOrder] Dlg0={7BD2CFF6-B037-47D6-A76B-D941EE13AD96}-
SdLicense-0 Count=4 Dlg1={7BD2CFF6-B037-47D6-A76B-D941EE13AD96}-
SdAskDestPath-0 Dlg2={7BD2CFF6-B037-47D6-A76B-D941EE13AD96}-
SdSelectFolder-0 Dlg3={7BD2CFF6-B037-47D6-A76B-D941EE13AD96}-
SdFinishReboot-0 [{7BD2CFF6-B037-47D6-A76B-D941EE13AD96}-SdLicense-0]
Result=1 [{7BD2CFF6-B037-47D6-A76B-D941EE13AD96}-SdAskDestPath-0]
szDir=C:\Program Files\IBM\Security
```

(The above parameter is the directory used to install Client Security. It must be local to the computer.)

```
Result=1
[{7BD2CFF6-B037-47D6-A76B-D941EE13AD96}-SdSelectFolder-0] szFolder=IBM Client Security Software
```

(The above parameter is the program group for Client Security.)

```
Result=1 [Application] Name=Client Security Version=5.00.002f
Company=IBM Lang=0009 [{7BD2CFF6-B037-47D6-A76B-D941EE13AD96}-
SdFinishReboot-0] Result=6 BootOption=3
```

Configuring: The following file is also essential when initiating a mass configuration. The file can be named anything, as long as it has a .ini extension. The following list details the settings and setting explanations for the .ini file you must create. Before you can open and revise the CSEC.INI file, you must first decrypt it, using CONSOLE.EXE in the Security folder.

The following command runs the .ini file from the command line when the mass configuration is not performed along with a mass installation:

```
<CSS installation folder>\acamucli /ccf:c:\csec.ini
```

Table 4. Client Security System configuration settings

[CSSSetup]	Section header for CSS setup.
suppw=bootup	Administrator/Supervisor password. Leave blank if not required.
hwpw=11111111	CSS hardware password. Must be eight characters. Always required. Must be correct if hardware password has already been set.
newkp=1	1 to generate a new administrator key pair 0 to use an existing administrator key pair.
keysplit=1	When newkp is 1, this determines the number of private key components. **Note:** If the existing keypair uses multiple private key parts, all private key parts must be stored in the same directory.
kpl=c:\jgk	Location of the administrator key pair when newkp is 1, if this is a network drive it must be mapped.
kal=c:\jgk\archive	Location of the user key archive, if this is a network drive it must be mapped.
pub=c:\jk\admin.key	Location of the administrator public key when using an existing administrator key pair, if this is a network drive it must be mapped.
pri=c:\jk\private1.key	Location of the administrator private key when using an existing administrator key pair, if this is a network drive it must be mapped.
wiz=0	Determines if this file was generated by the CSS setup wizard. This entry is not necessary. If you include it in the file the value should be 0.
clean=0	1 to delete the .ini file after initialization, 0 to leave the .ini file after initialization.
enableroaming=1	1 to enable roaming for the client, 0 to disable roaming for the client.
username= [promptcurrent]	[promptcurrent] to prompt the current user for the system registration password. [current] when the system registration password for the current user is provided by the sysregpwd entry and the current user has been authorized to register the system with the roaming server. [<specific user account>] if the designated user has been authorized to register the system with the roaming server and if the system registration password for that user is provided by the sysregpwd entry. Do not use this entry if the enableroaming value is 0, or if the enableroaming entry is not present.
sysregpwd=12345678	System registration password. Set this value to the correct password to enable the system to be registered with the roaming server. Do not include this entry if the username value is set to [promptcurrent], or if the username entry is not present.
[UVMEnrollment]	Section header for user enrollment.

Table 4. Client Security System configuration settings (continued)

enrollall=0	1 to enroll all local user accounts in UVM, 0 to enroll specific user accounts in UVM.
defaultuvmpw=top	When enrollall is 1, this will be the UVM passphrase for all users.
defaultwinpw=down	When enrollall is 1, this will be the Windows password registered with UVM for all users.
defaultppchange=0	When enrollall is 1, this will establish the UVM passphrase change policy for all users. 1 to require the user to change the UVM passphrase at next logon, 0 to not require the user to change the UVM passphrase at next logon.
defaultppexppolicy=1	When enrollall is 1, this will establish the UVM passphrase expiration policy for all users. 0 to indicate that the UVM passphase expires 1 to indicate that the UVM passphrase does not expire
defaultppexpdays=0	When enrollall is 1, this will establish the number of days until the UVM passphase expires for all users. When ppexppolicy is set to 0, set this value to establish the number of days until the UVM passphrase expires.
enrollusers=*x*, where x is the total number of users you will enroll on the computer.	The value in this statement specifies the total number of users that you will enroll. When enrollall is 0, this is the number of users that will be enrolled in UVM.
user1=jknox	Provide the information for each user to be enrolled starting with user 1. (There is no user 0.) User names must be the account names. In order to get the actual account name on XP, do the following 1. Start Computer Management (Device Manager). 2. Expand the Local Users and Groups node. 3. Open the Users folder. The items listed in the Name column are the account names.
user1uvmpw=chrome	Specify the UVM passphrase for user 1 UVM.
user1winpw=spinning	Specify the Windows passphrase for user 1 to be registered with UVM.
user1domain=0	Specify whether the account for user 1 is local or on the domain. 0 to indicate that this account is local, 1 to indicate that this account is on the domain.
user1ppchange=0	Specify whether user 1 will be required to change the UVM passphrase at next logon. 1 to require the user to change the UVM passphrase at next logon, 0 not to require the user to change the UVM passphrase at next logon.
user1ppexppolicy=1	Specify whether the UVM passphrase for user 1 expires. 0 to indicate that the UVM passphrase expires. 1 to indicate that the UVM passphrase does not expire.
user1ppexpdays=0	If user1ppexppolicy=0, set this value to indicate the number of days until the UVM passphase expires.

Table 4. Client Security System configuration settings (continued)

For each user provide a complete set of configuration settings in the order specified in the shaded portion of the table. Provide all parameters for one user, and then provide parameters for the next user. If for example enrollusers were set to 2, you would add the following group of configuration settings.	
user2=chrome	
user2uvmpw=left	
user2winpw=right	
user2domain=0	
user2ppchange=1	
user2ppexppolicy=0	
user2ppexpdays=90	
[UVMAppConfig]	Section header for UVM-aware application setup and UVM-aware module setup.
uvmlogon=0	1 to use UVM logon protection, 0 to use Windows logon.
entrust=0	1 to use UVM for entrust authentication, 0 to use entrust authentication.
notes=1	1 to use UVM protection for lotus notes, 0 to use notes password protection.
netscape=0	1 to sign and encrypt e-mails with the IBM PKCS#11 module, 0 to not sign and encrypt e-mails with the IBM PKCS#11 module.
passman=0	1 to use Password Manager, 0 to not use Password Manager
folderprotect=0	1 to use File and Folder Encryption, 0 to not use File and Folder Encryption.
autoprotect=0	When folderprotect is set to 1, this will establish whether auto protection is used. 1 to use auto protection, 0 to not use auto protection.

Notes:

1. If any files or paths are on a network drive, the drive must be mapped to a drive letter.
2. As IBM Client Security Software is enhanced and updated, the *.ini parameters may change.

After you have revised the CSEC.INI file, you must encrypt it using CONSOLE.EXE and then add it to the build.

Chapter 5. Remotely deploying new or revised security policy files

Whether you are updating security policies or creating different policies for different computers, the IT administrator with signing authority can revise and deploy policy files. Edit the policy file, using ACAMUCLI.EXE. (You can also edit the policy by double-clicking the IBM Security Subsystem icon in the Control Panel.)

Sign the policy file according to on-screen instructions after you click Apply. (**Note:** if the administrator private key has been split, all components must be entered in order to sign the policy file.) The files you have edited are GLOBALPOLICY.GVM and GLOBPOLICY.GVM.SIG. Distribute these files to appropriate users the file, making sure that they are saved to the Security\UVM_Policy folder.

You can update passphrase policies remotely after deployment. Updating the passphrase policy file enables you to change the passphrase requirements when (or if) the user next changes their passphrase. The administrator can define a period of time, after which the user is forced to change the passphrase. This time period is defined during user creation. An example would be as follows: The administrator creates a user, Jane, and the initial policy states that user Jane has to have an eight-character password that expires every 30 days. The admin could update the policy file and require that the next time that Jane changes her passphrase, the new passphrase must now be 12 characters. The administrator could also change the expiration period. For example, instead of every 30 days the administrator could require Jane to change passphrases every 15 days. What happens in the following scenario? You are in day 10 of the 30-day passphrase ″life.″ A new passphrase policy file is sent to the client computer that states that the passphrase must be changed every 15 days. Does the passphrase expire in 5 days or in 20 days. The passphrase expires in 20 days as the original policy stated. The passphrase expiration policy goes into effect when the passphrase is set. The 15-day change policy will commence when Jane changes her passphrase after the 20 days.

If you want to change the required characteristics of the passphrase, follow the instructions above. Then distribute the following files the SECURITY\UVM_POLICY folder: UVM_PP_POLICY.DAT and UVM_PP_POLICY.DAT.SIG.

Chapter 6. Known conflicts and resolutions

At this writing IBM is unaware of any conflicts or issues with ESS. As conflicts and their resolutions arise, we will periodically update this document and post it on the Web.

Appendix. Notices

IBM may not offer the products, services, or features discussed in this document in all countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

The products described in this document are not intended for use in implantation or other life support applications where malfunction may result in injury or death to persons. The information contained in this document does not affect or change IBM product specifications or warranties. Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of IBM or third parties. All information contained in this document was obtained in specific environments and is presented as an illustration. The result obtained in other operating environments may vary.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Non-IBM Web sites

Any references in this publication to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product, and use of those Web sites is at your own risk.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM
ThinkPad
ThinkCentre

Microsoft, Windows, and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.