2 Architectural Overview

2.1 Introduction

CMS is a web based smart card lifecycle management system, it can be leveraged to manage PKI, symmetric key, biometric, *one-time password* and other credentials on smart cards. In the context of the ABC smart card administration project, PKI type credentials are used solely.

The CMS sub-system is essentially made of four key components:

- **ActivIdentity CMS *engines* and web portals**
- **HSM for hosting GlobalPlatform Key (GPK) material**
- **A SQL Database Management System (DBMS)**
- **ActivIdentity ActivClient Client Middleware (crypto libraries and PIN management tools)**

The CMS sub-system also interacts with the following infrastructure elements:

- **ABC Issuing CA Servers**
- **ABC Active Directory (AD) Domain Controllers (DCs)**
- **ABC Exchange Email Service**
- **Network Load Balancing Services**

Figure 1 illustrates the CMS sub-system components and the relationship with the infrastructure elements listed above.

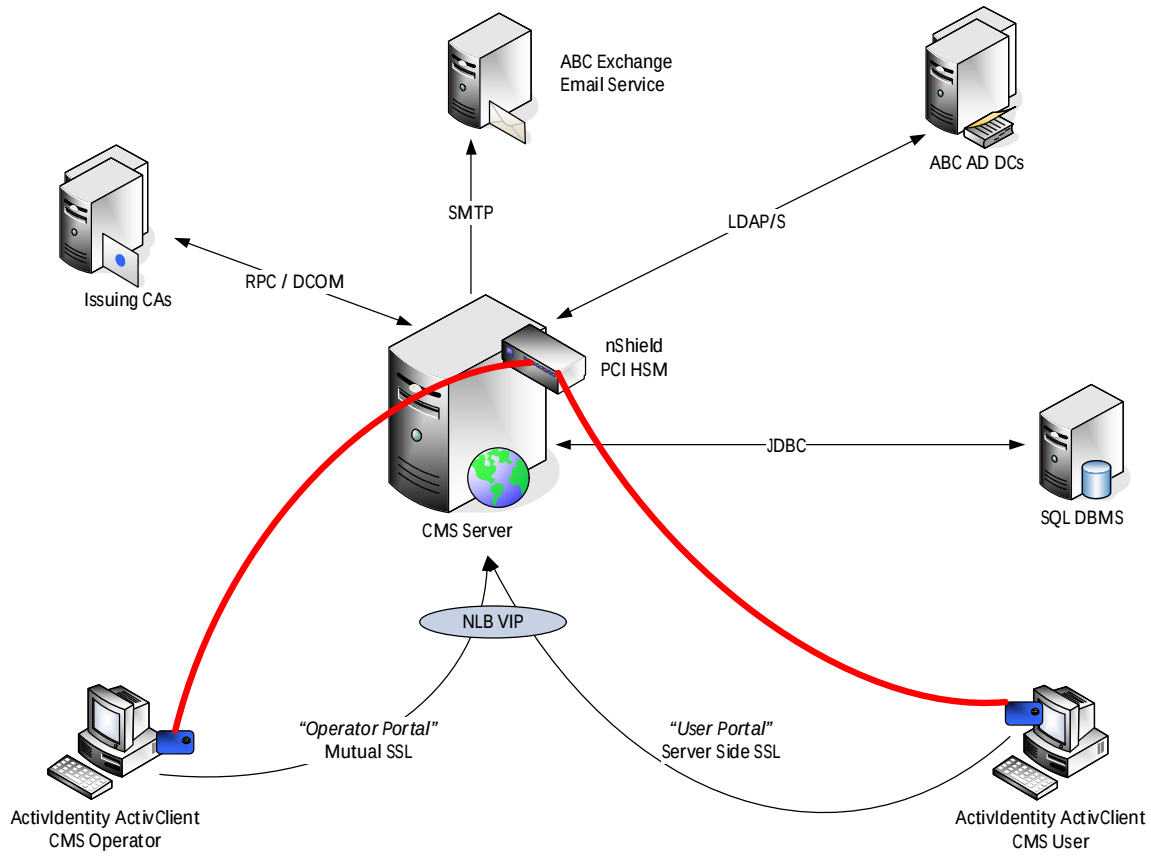

Figure 1: CMS Sub-System Logical Overview

2.2 CMS Sub-System General Overview

2.2.1 CMS Services

CMS engines and web portals are installed onto a Windows Server 2003 server and *run* on Internet Information Services (IIS) and TomCat Servlet / Java Server Pages. The CMS engines comprise three components (portal, card content server and audit server), these are described in Section 4.4. To provide the required levels of performance and availability, multiple physical CMS service (server) instances are implemented, they are configured such that they are presented as a single logical instance.

The CMS service is "presented" via two web portals:

- **Operator Portal: Accessed by nominated CMS operators to perform card bindings and help desk type functions, etc.**
- **User Portal (My Digital ID Card (MDIDC)): Accessed by users to complete "self service" card issuance and reset forgotten smart card PIN codes, etc.**

Both portals enforce SSL encrypted channels for all communications between the CMS server and client.

2.2.2 Hardware Security Modules

CMS leverages the GlobalPlatform key management standard to ensure that smart card management operations are performed in an extremely secure manner. For CMS to manage smart cards using the GlobalPlatform standard it needs to be able to store and access symmetric key material in a cryptographically secure environment; for this purpose, nCipher HSMs are required to be directly attached to all physical instances of CMS server. The means by which CMS uses the GlobalPlatform standard for smart card management is described in Section 3.1.

2.2.3 SQL Database Management System

CMS leverages a SQL DBMS for creation of a number of databases which are used to store CMS configuration, card content information, audit records, etc. The DBMS selected for the ABC smart card administration system is Microsoft SQL Server 2000, deployed on Windows Server 2003. It should be noted that SQL Server 2005 is not a supported product with CMS.

2.2.4 ActivIdentity ActivClient & Smart Card Reader

2.2.4.1 ActivClient

ActivClient provides both the smart card middleware and *limited* smart card management tools.

Applications (such as Windows smart card logon) access material held on the smart card using the ActivIdentity Cryptographic Service Provider (CSP) provided by the middleware. An example function of the smart card management tool in ActivClient is the ability to remotely reset (whilst disconnected from the network) the PIN using a challenge / response sequence where the response is generated by CMS; in this circumstance, the challenge is generated by the ActivClient software.

2.2.4.2 Smart Card Reader

The following smart card readers are to be used in the ABC smart card administration project:

- **Embedded smart card reader in TREDSS desktop PCs' keyboards**
- **PCMCIA Fujitsu Siemens LifeBook SmartCase Cardholder**

2.3 CMS Sub-System Physical Components

The CMS sub-system comprises a total of four CMS servers, this number has been recommended by ActivIdentity based upon performance and volume criteria specified in Appendix A. CMS version 4.0 SP3 is selected, this incorporates CMS peering capability (described in Section 4.6) and has improved database connectivity performance over previous versions.

Microsoft SQL Server 2000 DBMS is deployed in a *standalone* configuration, CMS therefore connects to a single logical instance of SQL Server. A primary SQL Server instance is deployed in Location A

and a *secondary* SQL Server instance is deployed in Location B. The secondary SQL Server is only ever be invoked in a disaster scenario preventing the use of the SQL Server instance in Location A. Under no circumstances are primary and secondary SQL Server instances both operational in supporting CMS.

Figure 2 illustrates a high-level physical representation of the CMS server and SQL Server deployment; CA servers, though not part of the CMS sub-system, are included in the illustration for *completeness*.

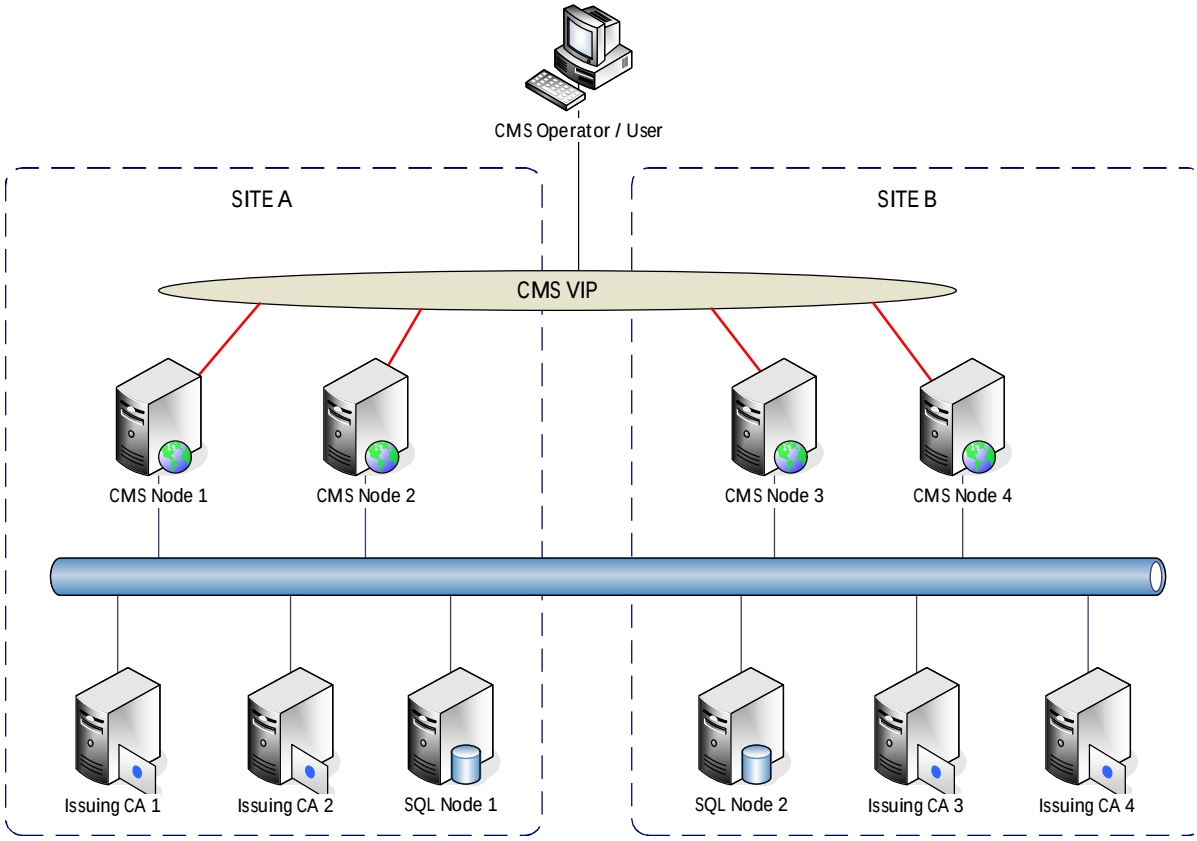

Figure 2: CMS / SQL Physical Implementation

It is worth noting that the deployment of a Microsoft SQL Server database on a separate server than the CMS server mandates that Windows integrated authentication cannot be used by CMS to communicate with the SQL database, hence, native SQL authentication must be used.

2.4 CMS Infrastructure Integration

2.4.1 "Enterprise Issuing" Certification Authorities

CMS is configured to request and manage certificates from the four Microsoft Enterprise CA servers deployed as part of the ABC smart card administration project. The "CMS AD service accounts" which CMS also uses to make Lightweight Directory Access Protocol (LDAP) binds to AD (see Section 2.4.2) require suitable entitlement at the CA servers to request certificates, manage certificates, etc., and also entitlement to AD based certificate templates.

2.4.2 Active Directory Domain Controllers

LDAP repositories are defined in CMS for it to be able to retrieve user information. Each AD domain in the ABC AD forest containing users which may be issued smart cards by CMS is defined independently. Binds from CMS to Active Directory (LDAP) domain controllers are made over SSL, i.e. LDAP/S, this requires server side authentication only (making use of existing certificates deployed to domain controllers).

LDAP binding and querying is a relatively performance intensive element of CMS operations and the implementation of the ABC data centre migration program ensures that suitable high-performance Active Directory domain controllers (in the `abc.gpn.gov.uk` domain) are available to the CMS

servers to optimize LDAP binding and retrieval performance. *Dedicated* AD domain controllers for CMS purposes are therefore not required.

2.4.3 Exchange Email Service

The smart card issuance workflow chosen for the ABC smart card administration project dictates that smart card *personalisation* (the generation and injection of PKI credentials onto the smart card) is performed by the user *at their desk* following a card binding process in the presence of a card stockholder CMS operator. This workflow requires that an initial password is emailed to the user following the binding, which is required by the user when authenticating to the user portal with the bound, but not personalised, smart card. For new users (who don't have a "legacy" Windows smart card logon capability) and therefore cannot retrieve an email, a separate PIN retrieval mechanism is used (which doesn't involve email).

CMS is configured to send the initial PIN as part of a custom formatted email message to users via a Simple Mail Transport Protocol (SMTP) Exchange messaging gateway service.

2.4.4 Network Load Balancing Services

To meet the availability and performance requirements of the ABC smart card administration project it is necessary to employ network load balancing for the web URL used by clients to connect to the CMS operator and user portals in the CMS sub-system. A Virtual TCP/IP (VIP) address hosting capability is provided by Cisco Context Service Switch (CSS) appliance based load balancers.

CMS users and operators connect to CMS via a single DNS host address:

- `abc.gpn.gov.uk`

The AD Domain Name System (DNS) contains a new host (type A) record which resolves to an inbound VIP defined on the load balancer. The corresponding outbound VIP definition specifies the *physical* TCP/IP addresses of each CMS node. The DNS record incorporates an identifier (`-np`) which ensures that traffic bypasses any ISA Server proxy services.

3 CMS Functional Overview

3.1 GlobalPlatform Key Management

3.1.1 Operational Functionality

As stated in Section 2.2.2, CMS leverages the GlobalPlatform Key management standard to support card issuance, management, termination, etc.

Essentially, the way that CMS employs GlobalPlatform Keys (3DES symmetric keys) means that it can establish a secure communication channel to the card directly. It does this without employing Microsoft Cryptographic Application Programming Interface (CAPI) or Public Key Cryptography Standard (PKCS) #11 interface, this results in CMS having a very powerful and secure method for managing smart cards.

The fundamental concept of the CMS implementation of the GPK standard is the use of symmetric keys for authentication and establishment of a secure channel between the CMS server and the smart card operating system. When a shipment of smart cards are procured from a manufacturer, it is necessary for the smart cards to be *injected* with a unique symmetric key derived by the smart card's serial number and an issuer master key in HSMs attached to the CMS servers deployed in the ABC smart card administration project.

When a card management operation such as initial issuance, certificate renewal, PIN reset, etc. is performed a mutual authentication is performed between the smart card and the CMS server using the aforementioned GPK and GlobalPlatform SecureChannel; further to the successful authentication, an "SSL-like" secure channel is established. Thereafter, any loading of applets, certificates, PIN updates, etc. is performed over the secure channel, as illustrated in Figure 3. Note that public / private keys are generated on the smart card.

It is worth noting that, due to security and *user friendliness* reasons, smart cards issued by CMS do not employ the concept of admin PINS / PIN Unlock Keys (PUKs) favored by most smart card management systems. Instead, challenge / response based on 3DES keys and / or the SecureChannel protocol is used to secure PIN unlock.

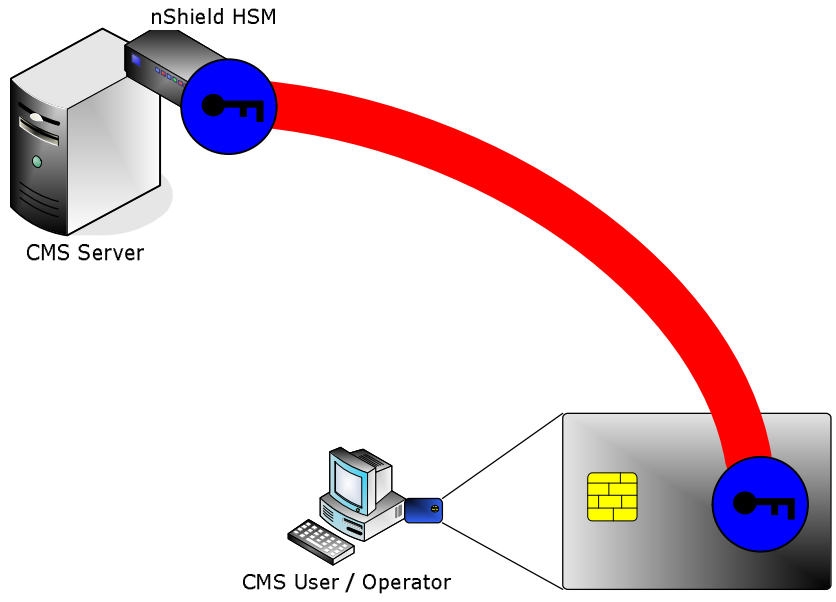

Figure 3: GlobalPlatform Keys

3.1.2 GPK Setup and Configuration

It is worthwhile understanding the mechanism used to set up the initial GlobalPlatform keys in the CMS HSMs, as well as the card personalization process which replaces the "AI Master GPK" with a "DWP Master GPK"; this is illustrated in Figure 4.

The AI Master Key and ABC Master Key sets are generated and brought into the nCipher Security World using ActivIdentity Key Management System (KMS) at the SIAG Key Management Station in a controlled, witnessed and audited procedure (i.e. a key ceremony).

The *Master Key* is actually a set of three distinct symmetric keys, however, for ease of understanding they can be considered as a single key.

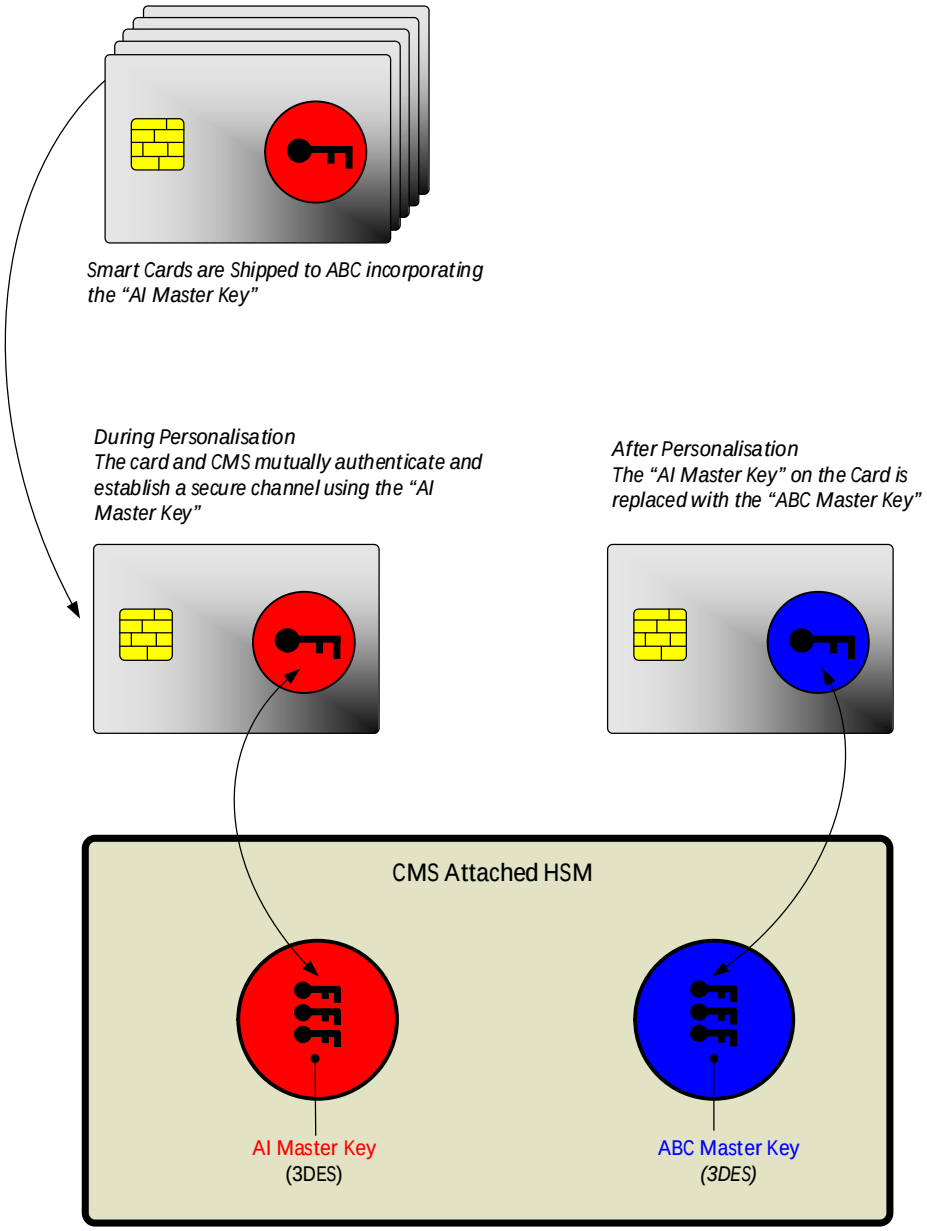

Figure 4: GlobalPlatform Key Establishment

Distribution of GPK material to the CMS servers is performed using the controls available from the nCipher Security World.

The manufacturers contracted to supply smart cards for the ABC smart card administration must be informed to inject the requisite AI Master Key onto the smart cards they are producing (the AI Master Key is "known" to all leading smart card manufacturers).

During initial issuance of a user's smart card, further to successful authentication with the AI Master Key GPK, a personalization process replaces the "AI Master Key" GPK with a key diversified by the "ABC Master Key" and the smart card's serial number.

Future management operations using the smart card solely use the new, derivative GPK.

3.2 Overview of CMS Smart Card Issuance

3.2.1 Introduction

This section provides a high level approximation of the processes and components involved in the issuance of a user authentication certificate to a bound smart card in the ABC smart cards administration solution. It must be recognized that this is:

- **A vast simplification of the processes to reduce the complexity for ease of understanding**
- **Some process elements are not fully discussed in this document, e.g. CA HSMs, etc.**

Figure 5 shows the components and process numbers, which are described Section 3.2.2.

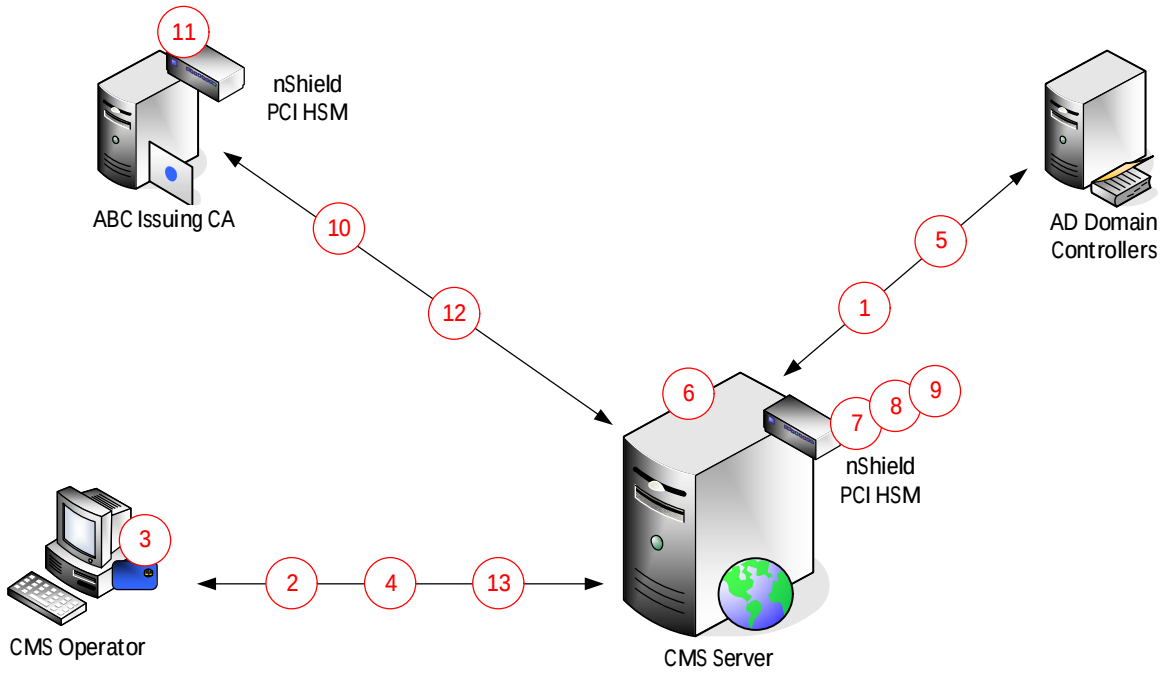

Figure 5: Smart Card Issuance

3.2.2 Process Descriptions

The following process description assumes a user is performing self-service issuance with an already bound smart card and possession of a requisite initial password.

1. **A user accesses the CMS user (MDIDC) portal to initiate smart card personalization. CMS retrieves from the smart card the smart card serial number and searches its database for the name of the user to which the smart card has been bound. The user is asked to enter the initial password to verify that he/she is the rightful owner of the smart card. CMS verifies the initial password against the smart card serial number in the database and directory.**
2. **A secure channel is established using the GPK in the nCipher nShield HSM attached to the CMS server and the GPK on the user's smart card. The smart card is pre-personalized by CMS, which includes processes such as replacing the GPK, configuring the smart card's layout (slots) and installing a PIN applet.**
3. **The public and private key pair is generated on the smart card (using the 32-bit microprocessor on the smart card module).**
4. **The public key is transmitted to the CMS server.**
5. **CMS uses the certificate template defined in the CMS card policy to determine which attributes (User Principle Name (UPN), common name, etc.) are required in a certificate request and retrieves the information from AD domain controllers.**
6. **CMS generates a certificate request incorporating the subscriber's subject information and the public key.**

7. **The Enrolment Agent (EA) private key, encrypted by the Security World (module) key, is transmitted from the CMS server's Security World key store on the CMS server's hard disk to the nCipher nShield attached to the CMS server; this is a one time operation as the key is subsequently stored persistently in the nShield HSM module until it is *de-activated.***
8. **The EA private key is decrypted by the module key within the CMS attached nShield HSM.**
9. **The certificate request is transmitted to the CMS attached nShield HSM, signed by the EA private key (within the CMS attached nShield HSM) and returned to the CMS server.**
10. **The signed certificate request is sent to the selected ABC issuing CA.**
11. **The ABC issuing CA verifies the EA signature and manufactures a certificate based upon information submitted in the request; it signs the certificate with its private key (which is stored in the nShield HSM attached to the issuing CA).**
12. **The certificate is returned to the CMS server.**
13. **The certificate is securely injected into the smart card via the secure channel established between the CMS server and the smart card.**

4 CMS Server Design

4.1 Server Build

The CMS servers are deployed on Windows Server 2003 Enterprise Edition in the `abc.gov.uk` **AD domain. The CMS application is deployed on a separate RAID 5 volume than the RAID 1 volume upon which the operating system is deployed.**

The hardware specification of the CMS servers is shown in Appendix B.

4.2 Availability / Disaster Recovery

The four CMS servers implemented in the solution enable both high availability and disaster recovery. The solution can withstand a failure of any of the CMS nodes which would be detected by the Cisco CSS load balancing service, the CSS service would then effectively take that CMS node *out of play*. In the event of a disaster at one of the sites, again, the CSS service would *mark* those nodes *out of play* and enable continued service with the remaining CMS nodes.

The high availability and disaster recovery capability described above requires no manual intervention.

CMS server binding to Active Directory domain controllers utilizes inherent LDAP availability capability configuration within CMS, i.e. network appliance load balancing capability is not required.

4.3 Performance

It should be noted that the specification of four CMS servers is largely geared towards meeting availability requirements rather than being increasing the overall performance throughput of the CMS system. ActivIdentity has stated that they only *guarantee* a performance correlation ratio of two CMS servers to one active SQL servers, i.e. additional CMS servers (over and above two) may not increase the overall performance of the *system*.

ActivIdentity has committed to a maximum of fifty concurrent connections to a single active SQL server node from a single CMS server, therefore they guarantee a minimum of one hundred connections (for the two CMS servers). The *one hundred connection limit* roughly corresponds to the number of smart card personalization processes which the CMS system can concurrently handle. However, ActivIdentity has validated the TREDSS CMS sub-system design and committed to endeavor to ensure that the additional CMS servers deployed for availability purposes are leveraged for system performance as far as possible.

4.4 CMS Web Portals

Clients will connect to the CMS portals, using the following URLs:

- **Operator:** `https://abc1.gpn.gov.uk:49153/aims/enterprise/operator`
- **User:** `https://abc2.gpn.gov.uk:49153/aims/enterprise/user`

It is possible to simplify the URL by implementing re-direction at the load balancer, such a URL would be: `https://abc.gpn.gov.uk/operator`**. However, this would require SSL termination at the load balancer (using the same certificate and private key which is used by IIS on the CMS server) and has not been expressed as a requirement for this project.**

The DNS name record incorporates an identifier which ensures that traffic bypasses any ISA Server proxy services, as this would work against the load balancing solution in that it would mask the client's TCP/IP address and make configuration of *stickiness* at the load balancer extremely difficult.

4.5 CMS "Engines"

The CMS service comprises three principal engines (all of which must be installed together on each CMS server *node*):

- **CMS Portal: Provides the user interface and workflow mechanisms for the user and operator portals**
- **Card Content Server: Manages the content and communication with smart cards**
- **Audit Server: Records all user and operator interactions with CMS**

Each of these engines communicates via mutually authenticated SSL channels which mandates the implementation of client authentication and server authentication certificates on each CMS node.

The client authentication certificate (and private key) must be available to CMS in a PKCS #12 file format on the CMS server's hard disk. The server authentication certificate (and private key) must be installed into the CAPI certificate store configured in IIS (this is the same certificate which is used by the CMS portal to support clients making HTTP/S connections).

See Appendix C for a tabularized "CMS infrastructure certificates" register.

4.6 CMS Peering

To enable a single common configuration to be employed across the four CMS nodes it is necessary to configure CMS peering. A pre-requisite of CMS peering is that all CMS nodes connect to the same SQL server.

With CMS peering established, changes can be made to any CMS node in the peer group and be immediately effective for all CMS nodes. CMS peering uses a specific CMS operator role defined for the purpose of peering and requires a certificate (and private key) to authenticate to each CMS node just like any other operator role. The certificate must be available to CMS in a PKCS #12 file format on the CMS server's hard disk.

It is worth noting that the use of CMS peering mandates that Windows integrated authentication cannot be used by CMS to communicate with the SQL database, hence, SQL native authentication must be used.

4.7 Enrolment Agent

In its role of enrolling certificates onto user smart cards, CMS effectively requests certificates on behalf of users. To facilitate this process CMS must have access to a Microsoft Enrolment Agent (EA) signing key (in many *PKIs* this is termed a Registration Authority (RA) signing key) to sign certificate requests prior to submission to the CA. The EA private keys (each CMS server is enrolled with EA certificates from both ABC-CA1 and ABC-CA2) are considered a security sensitive credential and must be secured accordingly, for the ABC smart card administration project, the EA private key is stored in the same nCipher nShield HSM that is attached to the CMS server for the purpose of storing GlobalPlatform keys.

The key will use the concept of *module only* protection in the nCipher Security World, i.e. there is no requirement for Operator Card Set (OCS) authorization to activate the key.

A distinct EA certificate will enrol for on each CMS node; additionally, because CMS is required to enrol for certificates from both ABC-CA1 and ABC-CA2, EA certificates must be enrolled from each CA.

4.8 CMS Startup Mode

When a CMS server is started up there are three passwords required for the CMS services to successfully instantiate: the SQL DB owner password, CMS security key (used to encrypt / decrypt sensitive data in the CMS databases) and the HSM PIN (used to authorise access to GPK material).

CMS has the option of unattended startup (where all the passwords are stored in an encrypted file on the CMS servers' file systems) or attended startup (an operator is prompted for all three passwords). For the ABC smart card administration project, the unattended startup mode is specified to reduce the level of operator involvement when restarting CMS servers.

4.9 CMS Binding to Active Directory

A nominated *service account* (AD user account) is specified in CMS for the purpose of binding to LDAP repositories. The CMS service account must have entitlement to read account information for all users in the domain (a separate binding definition is configured for each AD domain in the AD forest where users are enrolled). Because of the elevated privileges associated with the CMS service account (it can issue and manage certificates at the CA servers), it is necessary to protect the binding using LDAP/S such that the password is transmitted inside an SSL secured channel. Existing certificates issued to AD domain controllers are leveraged to facilitate LDAP/S.

4.10 User Attribute for Card Binding

When a user is bound to a smart card, CMS stores card content records associated with that user in the CMS database. A unique user attribute identifier must be specified in CMS for the user, the attribute must be unique within the LDAP repository for which it is defined.

It is important to recognise that any changes made to this attribute in AD will immediate render any smart cards issued to that user orphaned, and therefore unmanageable. The smart card cannot even be recycled, since the user associated with a smart card must be terminated prior to a recycle operation. The Universal Principal Name (UPN) attribute might seem a good candidate for ensuring uniqueness with an AD forest, not just the AD domain. However, experience has shown that this is can often be a relatively frequently updated attribute (following change of marital status, etc.) and therefore unsuitable for use as the card binding attribute. It is therefore stipulated that the sAMAccountName attribute is used by CMS for card binding.

4.11 CMS Portal TCP/IP Listening Ports

The CMS IIS web portals must have TCP/IP listening ports defined for inbound SSL communications, it is proposed that the default ports setup by the installation routine are used:

- **User (MDIDC) Portal: 49153**
- **Operator Portal: 49153**

4.12 CMS Card Policy

4.12.1 Physical Attributes

For CMS to issue a smart card, it must have a card policy associated with it which defines the necessary layout and characteristics associated with the physical smart card.

The smart cards prescribed for use on the ABC smart card administration project are:

1. **Oberthur 64k Secure Channel Cosmo v5.2 Java Card**
2. **GemAlto 64K Secure Channel CyberFlex Access v2c Java Card**

Both smart cards use the same ActivIdentity Card Profile XML specification based upon the ActivIdentity v2 applet framework which provides the card with the following *layout definition*:

- **One PIN application slot**
- **Three Secure Channel (CMS) protected PKI slots**
- **Three PIN protected PKI slots**
- **One Secure Chanel protected SKI application slot (for AAA OTP applet)**

It should be noted that cards procured for the ABC smart card administration project are shipped with no layout defined and therefore no PIN applet instantiated, etc.

A representation of the layout of the smart card profile is illustrated in Figure 6.

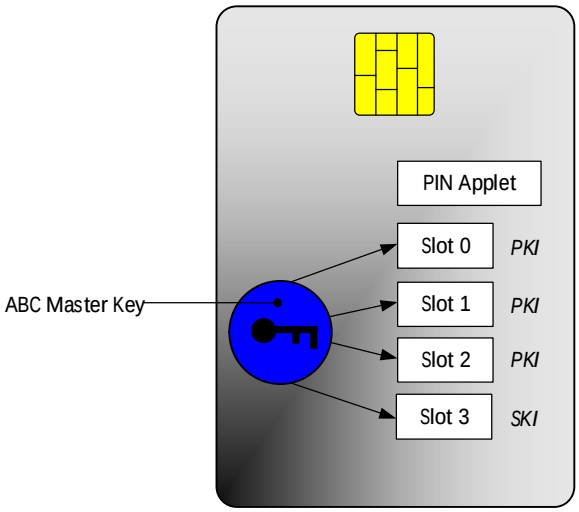

Figure 6: Smart Card Profile (Layout)

It is important to understand that in practice, there is no requirement for a user when completing self-service issuance to be aware of the smart card's manufacturer, i.e. the process is smart card agnostic. This is due to the fact both smart cards support the same applet framework and can therefore be configured with the same card policy based upon a common smart card profile specification. From an application standpoint, any smart card that is personalized with the same ActivIdentity applet framework will have the same "card edge" and will therefore look the same for the calling application regardless of what model or version the smart card is.

4.12.2 Policy Balancing

Due to the use of two separate CA servers in the solution, there needs to be two card policies defined, these would be presented to a CMS operator in a *drop-down list box*. Due to the requirement to balance certificate requests between the two CA servers, the default card policy selection should be randomized such that an operator does not ordinarily need to explicitly select a card policy.

This *balancing* functionality could feasibly be provided by using CMS groups to limit what card policies selected CMS operators are entitled to, however, this has two drawbacks which mean it has not been proposed in this design:

a) There are no *limited operators* defined in the solution, meaning that it would not be possible to distinguish assignments between CMS operators

b) In a disaster scenario whereby one of the CA servers was unavailable, CMS groups would need to be reconfigured to assign all CMS operators to the functional CA; whereas with the proposed solution it would only be necessary to inform CMS operators to explicitly select the correct card policy (relating to the functioning CA) when performing a smart card binding

In the event of a disaster making one of the CA servers unavailable, the card policy relating to the failed CA will be deselected from its card policy assignment.

4.12.3 PIN Policy

As described previously, the smart cards are shipped "blank", without even a PIN applet configured. The PIN applet is deployed as part of the smart card personalisation process and has a PIN policy defined as shown in Table 1.

Setting	Value
Minimum PIN Length	**6**
Maximum PIN Length	**12**
Maximum Number of Wrong PIN Tries Before Locking the Card	**3**
Maximum Number of Wrong Unlock Tries Before Blocking the Card	**12**
Force PIN to be Changed on First Usage	**No**
Allow Weak PIN	**No**
Force PIN to Contain Letters	**No**
Force PIN to Contain Digits	**Yes**

Table 1: PIN Policy Definition