SMConsole
Version 1.01

User Guide

Nanometrics Inc.
Kanata, Ontario
Canada

SMConsole Version 1.01 User Guide

The information in this document has been carefully reviewed and is believed to be reliable for Version 1.01.*xx*. Nanometrics, Inc. reserves the right to make changes at any time without notice to improve the reliability and function of the product.

Nanometrics, Inc.
250 Herzberg Road
Kanata, Ontario, Canada K2K 2A1
Tel (613)592-6776
Fax (613)592-5929
Email info@nanometrics.ca

Part number 15160R2
Release date 2005-10-28

Contents

Figures

Tables

SMConsole

1 About SMConsole

Security Management Console (SMConsole) provides command-line options for managing PKCS11-compliant cryptographic tokens on both digitisers and workstations. Tokens can be managed locally, and remotely via SSH. For workstation tokens SMConsole also provides commands to manage certificates, certificate revocation lists (CRLs), and the AutoDRM access control list (or authorization model).

1.1 Typical operation

Tokens on Europa digitisers are used to sign CD-1.0 and CD-1.1 frames. Workstation tokens are used to sign CD-1.1 frames and to provide S/MIME email signing and verification services using X.509 certificates, and to check command request authorization using the stored authorization model.

Objects such as keys, certificates, and CRLs may be stored on a token:

- Keys are used primarily for signing data and email.
- Certificates are used for email verification at the workstation token (for example, by the AutoDRM program, which requires that a valid certificate be included in any signed received message and which includes a certificate in outgoing signed emails).
- CRLs are used for verifying certificates.

Many objects may be stored on the token. Each stored object is assigned an ID number. The object type and ID number are used to specify a particular object on the token (for example, to specify which private key should be used for signing data). Related objects will have the same object ID. For example, if you generate a public/private key pair, then obtain and store a certificate containing the public key, all three objects (both of the keys and the certificate) will have the same object ID.

SMConsole provides functions for using a token with the NmxToCD1, NmxToCD11, and AutoDRM programs. The requisite management function determines how you should configure the token initially; see Section 3.3.1, “Managing digitiser tokens” and Section 3.3.2, “Managing workstation tokens” for token setup by function. See also documentation for PKCS11 Cryptography Token Version 2.01.

1.2 Summary of inputs and outputs

1.2.1 Inputs

- Solaris or Linux program files – `smconsole.jar, smconsole, bcprov-jdk14-122.jar, libcrystoki2.so, libpkcs11nmx.so`.
- Windows program files – `smconsole.jar, SMConsole.bat, bcprov-jdk14-122.jar, cryst201.dll, pkcs11nmx.dll`.
- Digitiser token:
  - DSA parameters file – The program accepts both Windows and Unix path names. The file specified must adhere to the Java Properties file syntax and contain the *p*, *q*, and *g* DSA parameters. The parameters are hex representations of Big Integers and can be up to a maximum size of 256, 40, and 256 hex characters respectively for *p*, *q*, and *g*. The parameter identifiers in the file are case-sensitive.
- Workstation token:
  - Certificate files in PEM format.
  - CRL files in PEM format.
  - DSA parameters file (with characteristics as described above).
- The appropriate Luna card drivers for your operating system and card reader.
- The appropriate security token access libraries for your operating system.

1.2.2 Outputs

- `SMConsole_`*yyyymmdd*`.log` – The log file contains timestamped messages for every user action or attempted action that can cause a change to the system (for example, changing the mapping of a user to a role).
- Digitiser token, in PEM or DER format:
  - Certificate requests.
- Workstation token, in PEM or DER format:
  - Certificate and CRL files.
  - Certificate requests.

2 Installing SMConsole

2.1 Dependencies

- SMConsole must be run from the same directory as NaqsServer. It will look in the working directory for the `naqsaddr.ini` file created by NaqsServer.
- For the workstation and authorization model functions:
  - It must be installed on the same machine as the workstation token, with that token being in the first slot if more than one are installed.
  - It must be run from the same directory as AutoDRM. (It uses `AutoDRM.ini` to get the workstation token PIN so that the authorization model can be retrieved from the token without requiring the user to log in first.)
- For the digitiser functions, the Comms Controller firmware must be version 5.81.01 or higher.

2.2 Install SMConsole

- See the installation instructions for the acquisition system workstation.

3 Using SMConsole

SMConsole commands are accessible through a hierarchy of menus (see Appendix A for an overview). These are divided into three general task areas:

- Digitizers – Manage the digitiser key pairs.
- Workstation – Manage key pairs and other information on the workstation token.
- User Authorization – Manage user access permissions.

3.1 Levels of access to menus and commands

The SMConsole menus and commands that are available depend on your level of access. For example, the User Authorization section of SMConsole is visible only to users with the MaintainAuthorizationModel permission (which defaults to be the nmx userID only). The default role and permission mappings are shown in Appendix B, “Access Level Defaults”. See also Section 3.3.3, “Managing user access and permissions,” on page 12.

3.2 Starting and stopping SMConsole

- To start SMConsole, enter `smconsole` in any terminal window.
- To stop SMConsole, enter `0` as required from each sub-menu until you have exited from the Main menu.

3.3 Using the SMConsole run-time commands

This section provides a summary of initial token configuration and the command options for each of the management functions (digitiser keys, workstation keys and token information, and authorization model).

- To run a command, enter the number for the corresponding menu option, and then (for some commands) enter parameter values as prompted.

All SMConsole menus have an Exit or Logout option `0`:

- To exit or log out from the current menu, enter `0`. This will open the menu that is one higher up in the hierarchy.
- To exit from SMConsole, enter `0` from the Main menu.

3.3.1 Managing digitiser tokens

Digitiser tokens are only used to sign CD-1.0 and CD-1.1 sub-frames. The digitiser token can store up to 2 key pairs.

3.3.1.1 Set up a digitiser token

Note When a digitiser is rebooted and there are no key pairs on the token, the digitiser will automatically create a key pair and make it active. It will not clear off any other existing data from the token.

1. Choose the digitiser via Digitizers > *digitiser number*, and then choose Initialize Token. When the digitiser initializes a new token, it clears any existing data off the token, then creates a new key pair and makes it active.
2. Choose Get KeyList to see what Key ID was assigned to the key pair (the default value is the instrument serial number).
3. Create a certificate request with this Key ID:
   a) Choose Generate CertReq, and then enter distinguished name parameters as appropriate for the certificate request (for Subject Name, Organization, Organization Unit, Locality, and Country).
   b) Save the file in `/nmx/user/`*digitiserID*`.crq`.
4. Email the certificate request to the CTBTO officer in charge of the system.

The digitiser will use the private key to sign data (in NmxToCD1 and NmxToCD11). Data receivers will use the public key to verify the digitiser signature.

3.3.1.2 Digitiser management commands

SMConsole provides the encryption key management functions listed in Table 3-1.

- To choose a digitiser from the list of all Europa digitisers connected to the system, enter the corresponding menu number from the list of digitisers:

  Main Menu – Digitizers > Available Digitizers – *digitiser menu number*

Table 3-1 Digitiser key management commands

Command	Description

Generate KeyPair	Generate a new DSA key pair on the security token using specified DSA parameters and either an automatically generated or a specified unique key ID. The key ID must be an integer from 1 to 4294967295 (unsigned 4 byte Integer value). The default key ID is the instrument serial number if that ID is not currently in use by either of the key pairs, otherwise some other default value will be assigned. A maximum of 2 key pairs can be stored on the token. A previously-inactive key pair will be removed if required to accommodate the new key pair. For example (commands are summarized):
	• Get KeyList
	`**********`
	`Number of Keypairs: 1`
	`Key IDs:  414`
	`Active Key Id:  414`
	`**********`
	• Generate KeyPair
	`DSA parameters file:  getKeyPair`
	`Key ID (0 for auto):  0`
	`Destination key information file:  generateKeyPair`
	• Get KeyList
	`**********`
	`Number of Keypairs: 2`
	`Key IDs:  414 5414`
	`Active Key Id:  414`
	`**********`
	Typically after generating a new key pair you would set the key pair to active and set the active key ID to the instrument serial number.
	▸ Use Set ActiveKey to make the new key pair active.
	▸ Use Change KeyID to set the new active key pair ID to the instrument serial number.

Get PublicKey	Retrieve the DSA public key value along with its parameters, for the key specified by the key ID. Specify the filename for the stored key information; there is no default filename. For example:
	`Key ID: 5414`
	`Destination key information file: getKeyPair`

Table 3-1 Digitiser key management commands (Continued)

Command	Description

Change KeyID	Change the ID of either of the 2 key pairs. A key pair ID can be changed to the ID of an existing key pair; the ID of the other key pair is then assigned an available default value. For example (commands are summarized):
	• Get KeyList
	`**********`
	`Number of Keypairs: 2`
	`Key IDs:  414 5414`
	`Active Key Id:  5414`
	`**********`
	• Change KeyID
	`Key ID to change:  5414`
	`New Key ID:  414`
	• Get KeyList
	`**********`
	`Number of Keypairs: 2`
	`Key IDs:  5414 414`
	`Active Key Id:  414`
	`**********`

Get KeyList	Retrieve the list of all key pair IDs currently defined and show which is active.

Set ActiveKey	Set the key to be used for data authentication to the specified key pair. For example (commands are summarized):
	• Get KeyList
	`**********`
	`Number of Keypairs: 2`
	`Key IDs:  414 5414`
	`Active Key Id:  414`
	`**********`
	• Set ActiveKey
	`Key ID:  5414`
	• Get KeyList
	`**********`
	`Number of Keypairs: 2`
	`Key IDs:  414 5414`
	`Active Key Id:  5414`
	`**********`

Delete KeyPair	Delete the specified non-active key pair, after prompting for confirmation.
Generate CertReq	Create a certificate request for the specified key and store it in a file. Encoding options include DER (default filename is `certreq.crq`) and PEM (default filename is `certreq.txt`). ▸ Specify the public key ID for which the certificate is requested, enter the distinguished name information, and select the storage format.
Change RSAKey	Change the key that is used to encrypt protected commands. Use this function when you suspect the current key may have been compromised.

Table 3-1 Digitiser key management commands (Continued)

Command	Description
Initialize Token	Clear (reinitialize) the digitiser token. This deletes all keys, certificates, and other data that are currently stored on the token. Initialization will automatically create a single key pair and make it active, with the instrument serial number as the key pair ID. It may take a minute or two for the action to complete.

3.3.2 Managing workstation tokens

Workstation tokens are used to provide S/MIME email signing and verification services using X.509 certificates, and can be used to store and export CRLs. These tokens may also be used by NmxToCD11 for signing data. The authorization model for assigning permissions to users (for example, to send commands via email) is stored on workstation tokens.

3.3.2.1 Set up a workstation token

1. Insert the token into the card reader.
2. Launch SMConsole.
3. Choose Workstation > Initialize Token. A confirmation prompt will display.
4. Enter `y` to initialize the token.
5. Set all PINs to a password, and leave the token label blank.
6. Choose Login User, and enter the *PIN*.
7. Choose Generate KeyPair and then set the key pair to active (Set ActiveKey).

   The system will use the private key to sign data (in NmxToCD11) or email (in AutoDRM). Data receivers will use the public key to verify the signature.
8. Export the public key as a certificate request:
   a) Choose Generate CertReq.

      Use the following guidelines for parameters in the certificate request:
      - Subject Name (CN): *CF-nn-mm* (The computer name, where *nn* denotes the Central Facility number and *mm* denotes the workstation number; for example, CF-01-02 for ws02 in the first central facility of station *Locality.*)
      - Organization (O): CTBTO
      - Organization Unit (OU): IMS
      - Locality (L): *station code* (for example, JMIC)
      - Country (C): (not used; press Enter)

   b) Save the file in `/nmx/user/`*computer name*`.crq`.
   c) Obtain a certificate containing this public key by sending the certificate request to an appropriate certificate authority (CA). For example, email this file to CTBTO, requesting in return the workstation certificate as well as the certificate of the Certificate Authority (CA) that issued the workstation certificate.
9. When the certificates are received, log in again as User.
10. Choose Load Certificate.

11. Load the CA certificate (it will probably be called `cacert.pem`).
12. If a CRL is available for the CA certificate, choose Load CRL, and then load the CRL.
13. Choose Load Certificate.
14. Load the workstation certificate (it will probably be called *computer name*`.pem`.)
15. (Optional) Store other trusted certificates on the token using the Load Certificate option. These certificates are used to identify users authorized to request data via AutoDRM.
16. Log out.

3.3.2.2 Workstation management commands

Note Commands used to alter data on the workstation token (commands in the Workstation and User Authorization sections) will update the first token that is found on the workstation.

The workstation menu provides access to sub-menus for token, encryption key, certificate, and CRL management (see Table 3-2, Table 3-3, and Table 3-4). The menus and options available depend on whether you are logged in as a Security Officer or as a User (see also Section A.2, “Workstation menus,” on page 16).

Table 3-2 Workstation token menu commands

Command	Description
Get TokenInfo	Get information about the token through options in the Token Information menu: • Get Version – Get the token version information, such as manufacturer and library version. • Get MechanismList – Get a list of all encryption mechanisms (or algorithms) supported for the token in the specified slot. • Get SlotInfo – Get information about a slot on this workstation. • Get TokenInfo – Get detailed information about a token. • Get SlotList – Get a list of the slots on this workstation, either all slots or those slots that have a token.
Login SO	Log in to the token with Security Officer (SO) access. See Table 3-3 for SO-level token management options.
Login User	Log in to the token with User access. See Table 3-4 for User-level token management options.

Table 3-3 Workstation Security Officer commands

Command	Description
Change PIN	Change the login PIN for this Security Officer account.
Initialize Token	Clear (reinitialize) the workstation token. This deletes all keys, certificates, and other data that are currently stored on the token. ▸ After initializing the token, you must set the Security Officer (SO) PIN, token label, and User PIN.

Table 3-3 Workstation Security Officer commands (Continued)

Command	Description
Initialize UserPIN	Initialize the User PIN for this token. This command will delete all User data from the token (including keys and certificates) as it initializes the User area of the token.

Table 3-4 Workstation User commands

Command	Description
Get SessionInfo	Get information about this session.
Change PIN	Change the login PIN for this User account. It prompts for the current User PIN and new User PIN. If the current PIN is entered correctly, the User PIN will be set to the new value.
Delete Object	Delete a user object (for example, a key pair) from the token. It prompts for the Object ID, and lists the types if multiple objects have the same ID. The Object properties will be shown, and you will be prompted to confirm the deletion.
List Objects	View summary information for all objects stored on the token: • The List Objects sub-menu provides options to display keys, certificates, or all objects.
Display Object	View detailed information for a single object on the token. It prompts for the Object ID (and lists the types if multiple objects have the same ID).

Generate KeyPair	Generate a DSA private / public key pair for data and email signing and verification. You can use either default or specified DSA parameters, and either an automatically generated or a specified unique key ID. The key ID must be an integer from 1 to 4294967295 (unsigned 4 byte Integer value). The key labels are generated automatically.
	A maximum of 2 key pairs can be stored on the token. An inactive key pair will be removed if required to accommodate the new key pair. For example (commands are summarized):
	• List Keys
	`**********`
	`Keys found on the Token:`
	`PUBLIC_KEY ID: 100 DSA LABEL: NmxActiveKey`
	`PUBLIC_KEY ID: 3 DSA LABEL: NmxNotActive`
	`PRIVATE_KEY ID: 100 DSA LABEL: NmxActiveKey`
	`PRIVATE_KEY ID: 3 DSA LABEL: NmxNotActive`
	`**********`
	• Generate KeyPair
	`DSA parameters file (optional):`
	`Key ID (0 for auto): 0`
	`Key Pair with ID: 2 generated.`
	• List Keys
	`**********`
	`Keys found on the Token:`
	`PUBLIC_KEY ID: 100 DSA LABEL: NmxActiveKey`
	`PUBLIC_KEY ID: 2 DSA LABEL: NmxNotActive`
	`PRIVATE_KEY ID: 100 DSA LABEL: NmxActiveKey`
	`PRIVATE_KEY ID: 2 DSA LABEL: NmxNotActive`
	`**********`
	Typically after generating a new key pair you would set the key pair to active.

Table 3-4 Workstation User commands (Continued)

Command	Description

Set ActiveKey	Specify the key pair to be used for signing data and emails. For example (commands are summarized):
	• List Keys
	`**********`
	`Keys found on the Token:`
	`PUBLIC_KEY ID: 100 DSA LABEL: NmxActiveKey`
	`PUBLIC_KEY ID: 2 DSA LABEL: NmxNotActive`
	`PRIVATE_KEY ID: 100 DSA LABEL: NmxActiveKey`
	`PRIVATE_KEY ID: 2 DSA LABEL: NmxNotActive`
	`**********`
	• Generate KeyPair
	`DSA parameters file (optional):`
	`Key ID (0 for auto): 0`
	`Key Pair with ID: 3 generated.`
	• List Keys
	`**********`
	`Keys found on the Token:`
	`PUBLIC_KEY ID: 100 DSA LABEL: NmxActiveKey`
	`PUBLIC_KEY ID: 3 DSA LABEL: NmxNotActive`
	`PRIVATE_KEY ID: 100 DSA LABEL: NmxActiveKey`
	`PRIVATE_KEY ID: 3 DSA LABEL: NmxNotActive`
	`**********`
	• Set ActiveKey
	`Key ID: 3`
	`Key Pair with ID: 3 set to Active.`
	• List Keys
	`**********`
	`Keys found on the Token:`
	`PUBLIC_KEY ID: 100 DSA LABEL: NmxNotActive`
	`PUBLIC_KEY ID: 3 DSA LABEL: NmxActiveKey`
	`PRIVATE_KEY ID: 100 DSA LABEL: NmxNotActive`
	`PRIVATE_KEY ID: 3 DSA LABEL: NmxActiveKey`
	`**********`

Table 3-4 Workstation User commands (Continued)

Command	Description

Change KeyId	Change the ID of either of the 2 key pairs to any valid ID (any integer from 1 to 4294967295). If you change a key pair ID to the ID of an existing key pair, the ID of the other key pair is automatically assigned an available default value. For example (commands are summarized):
	• List Keys
	`**********`
	`Keys found on the Token:`
	`PUBLIC_KEY ID: 2 DSA LABEL: NmxNotActive`
	`PUBLIC_KEY ID: 100 DSA LABEL: NmxActiveKey`
	`PRIVATE_KEY ID: 2 DSA LABEL: NmxNotActive`
	`PRIVATE_KEY ID: 100 DSA LABEL: NmxActiveKey`
	`**********`
	• Change KeyId
	`Key ID: 2`
	`New Key ID: 100`
	`Key Pair ID changed to: 100.`
	• List Keys
	`**********`
	`Keys found on the Token:`
	`PUBLIC_KEY ID: 100 DSA LABEL: NmxNotActive`
	`PUBLIC_KEY ID: 3 DSA LABEL: NmxActiveKey`
	`PRIVATE_KEY ID: 100 DSA LABEL: NmxNotActive`
	`PRIVATE_KEY ID: 3 DSA LABEL: NmxActiveKey`
	`**********`

Load Certificate	Load an X.509 certificate from file and store it on the token. ▸ Specify the relative or absolute filename for the certificate. The certificate may be DER or PEM encoded. • If the certificate is self-signed, a warning will display. Choose whether or not to trust (and store) the certificate. • If the certificate is not self-signed, the program will search the token for the issuer's self-signed certificate, and verify the new certificate using the issuer's public key. If the new certificate can be verified it will be stored automatically on the token, otherwise it will not be stored. • Unverified certificates will not be stored on the token. Each certificate stored on the token will be given an object ID. If the public key contained in a certificate matches a public key on the token, it will get the same ID number as the key.
Export Certificate	Export an X.509 certificate from the token and store it as a file. You must specify the object ID for the certificate to export, and a format in which to store the certificate. Encoding options include DER (default filename is `certf.cer`) and PEM (default filename is `certf.txt`). You can specify different filenames.
Load CRL	Load a certificate revocation list in X.509 format onto this token. The self-signed certificate that issued the CRL must be present. The CRL is stored with the certificate, and is deleted automatically when the certificate is deleted.
Export CRL	Export a certificate revocation list in X.509 format from this token. Encoding options include DER (default filename is `crlst.crl`) and PEM (default filename is `crlst.txt`). You can specify different filenames.

Table 3-4 Workstation User commands (Continued)

Command	Description
Generate CertReq	Generate a certificate request for a key pair. This exports a public key in a standard format as a PKCS10 Certificate Request. ▸ Specify the public key ID for which the certificate is requested, and the country, organization, organization unit, and subject for the certificate. The program will save the certificate request in either DER or PEM encoding. Default filenames are `certreq.crq` for DER, and `certreq.txt` for PEM. You can specify different filenames.

3.3.3 Managing user access and permissions

Note The User Authorization section is visible only to users with the Maintain-AuthorizationModel permission, which defaults to be only the nmx userID.

Users are assigned roles, and roles are assigned permissions. The stored mapping of users, their roles, and role permissions is the authorization model. This determines both the level of access to menus and commands for using and managing tokens, and permissions for sending requests via email (using AutoDRM):

- User level is determined when you log on to the workstation. Users can be added and deleted.
- One or more roles can be mapped to a user (for example, a system administrator might be assigned the role SecurityAdministrator). In the current release, the list of possible roles is fixed.
- Different permissions can be mapped to a role (for example, a SecurityAdministrator has permission to add or remove users and to modify other user roles, and to send various requests; a User has permission to request data). In the current release, the list of possible permissions is fixed.

See Appendix B for a list of the default roles and permissions.

3.3.3.1 Authorization model management commands

The User Authorization menu provides authorization model management functions (Table 3-5). Updating of users and roles is managed through the sub-menus.

Notes:

1) Commands that are used to alter data on the workstation token (commands in the Workstation and User Authorization sections) will update the first token that is found on the workstation.

2) Authorization model changes will not be seen by AutoDRM until the changes are saved and AutoDRM is restarted.

Table 3-5 User authorization commands

Command	Description
Display User	View the user name and the assigned role(s) for that user.

Table 3-5 User authorization commands (Continued)

Command	Description
Update User	Add/remove a user's roles. The sub-menu Update User: *user name* lists the options to add or remove roles, as applicable to the current user configuration. See Appendix B for the default role and permission values.
Add User	Add a new user. You can then use Update User to assign roles to the new user to grant them access to the appropriate permissions. When adding a new AutoDRM user, their user name must be the Subject Distinguished Name of the certificate used to sign their emails.
Delete User	Delete an existing user. You will be prompted for confirmation before the user is deleted.
Display Role	Display the role name and the assigned permissions. See Appendix B for the default values.
Update Role	Add/remove a role's permissions. The sub-menu Update Role: *role name* lists the options to add or remove permissions, as applicable to the current configuration. All permissions except MaintainAuthorizationModel are for email (AutoDRM) requests. Permissions include: • CenterMass – Can send mass centre commands. • GenerateKeypair – Can generate key pairs and store them on the digitiser token. • MaintainAuthorizationModel – Can change the user authorization model, such as adding a user and changing the user roles. • StartCalibration – Can send calibration commands. • StartContinuous – Can set the CD-1.1 sender to continuous on. • StartKeypair – Can set a digitiser key pair to active. • StopContinuous – Can set the CD-1.1 sender to continuous off. • UpdateCRL – Can update the certificate revocation list.
Load Defaults	Load the default authorization values (Appendix B). This will overwrite all changes made to the authorization model subsequent to the initial system configuration. ▸ If you want to keep the changes (either the default values, or any changes you have since made to the values that were loaded from default), use Save Changes before exiting to the Main menu.
Save Changes	Save changes made to authorization values before exiting the User Authorization menu. (Exit without saving will discard all of the changes.)

3.4 Updating the token configuration for AutoDRM

AutoDRM caches certificates and CRLs for 10 minutes, and access control lists are cached indefinitely. Any changes to these items made using SMConsole will not be seen by AutoDRM for that period of time unless AutoDRM is restarted.

▸ Restart AutoDRM after making changes to certificates, CRLs, or the access control list (authorization model) to ensure that the cache reflects the current token configuration.

3.5 Monitoring SMConsole operation

Log messages generated by SMConsole list all user actions and attempted actions that can change the state of the system. A new log file is created each day, with the name `SMConsole_`*`yyyymmdd`*`.log`; all applicable session messages are appended to this

file. The logs are stored in the directory `/nmx/log/SMConsole` (Linux and Solaris) or `c:\nmx\log\SMConsole` (Windows).

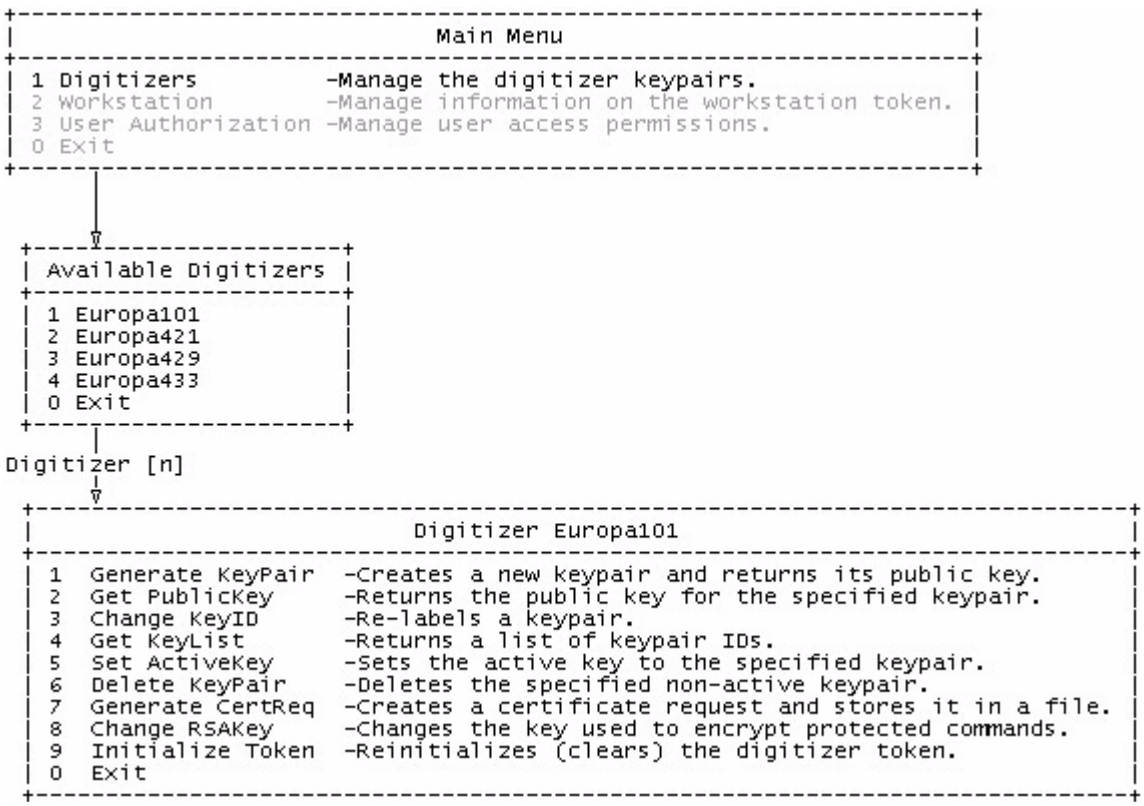
If SMConsole encounters an error, the console window will display an error message to help with solving the problem.

Appendix A Overview of Menus

This section provides a graphic overview of the SMConsole menus by each task area (digitiser token management, workstation token management, and authorization model).

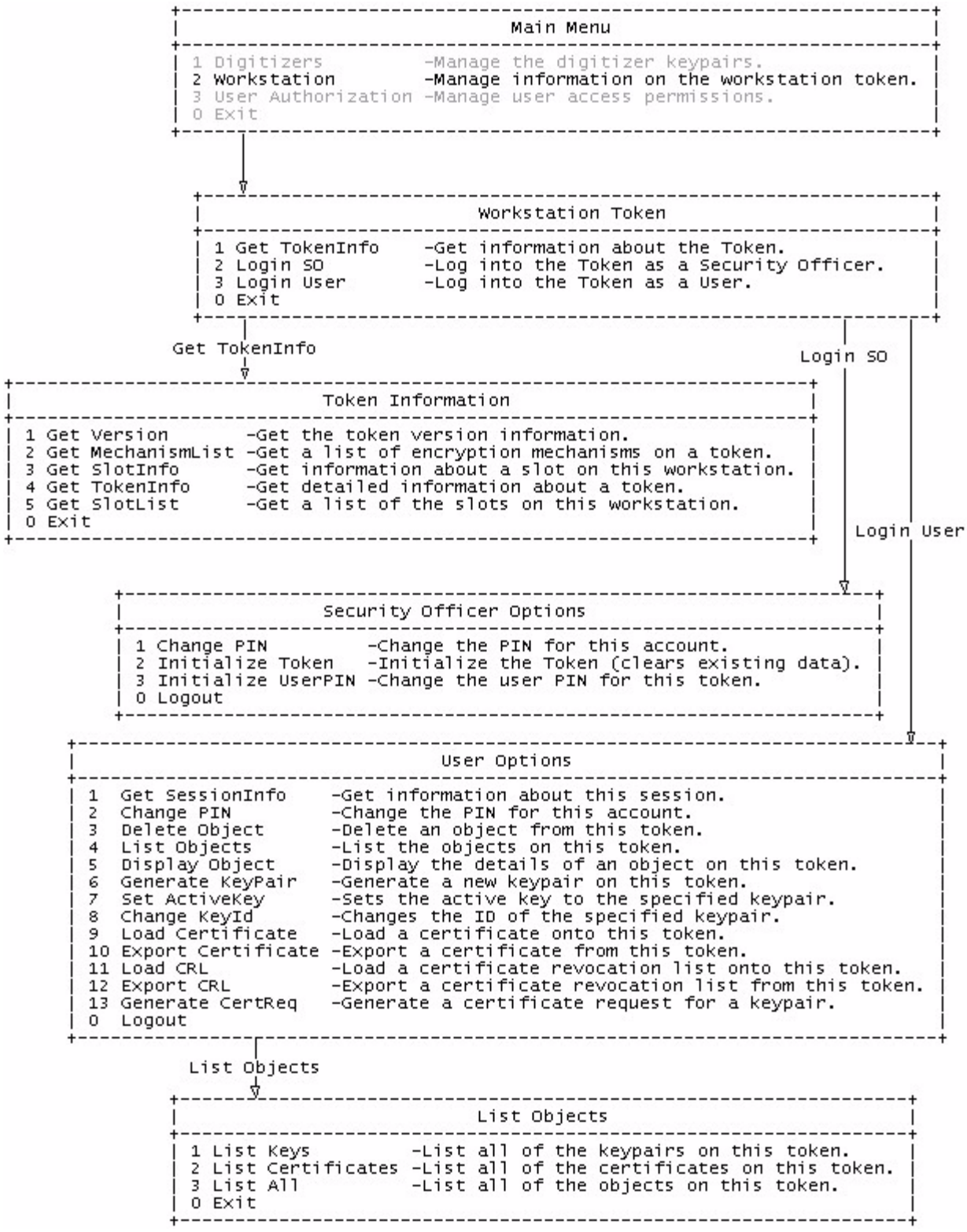
A.1 Digitiser menus

Figure A-1 Digitiser key management menus



A.2 Workstation menus

Figure A-2 Workstation token management menus



A.3 User Authorization menus

Figure A-3 User authorization management menus

Appendix B Access Level Defaults

Table B-1 shows the default role and permission mappings for the authorization model.

Table B-1 Default role and permission mappings*

User	Role	Permissions	Permission description
operator	Operator	CenterMass	Can send mass centre commands.
		StartCalibration	Can send calibration commands.
		StartContinuous	Can set the CD-1.1 sender to continuous on.
		StopContinuous	Can set the CD-1.1 sender to continuous off.
nmx	SecurityAdministrator	CenterMass	Can send mass centre commands.
		GenerateKeypair	Can generate key pairs and store them on the token.
		MaintainAuthorizationModel	Can change the user authorization model, such as adding a user and changing the user roles.
		StartCalibration	Can send calibration commands.
		StartContinuous	Can set the CD-1.1 sender to continuous on.
		StartKeypair	Can set a digitiser key pair to active.
		StopContinuous	Can set the CD-1.1 sender to continuous off.
		UpdateCRL	Can update the certificate revocation list.

Table B-1 Default role and permission mappings* (Continued)

User	Role	Permissions	Permission description
—	SecurityUser	CenterMass	Can send mass centre commands.
		GenerateKeypair	Can generate key pairs and store them on the token.
		StartCalibration	Can send calibration commands.
		StartContinuous	Can set the CD-1.1 sender to continuous on.
		StartKeypair	Can set a digitiser key pair to active.
		StopContinuous	Can set the CD-1.1 sender to continuous off.
unsigned	User	RequestData	Can request data (such as waveform data).

* All permissions except MaintainAuthorizationModel are for email (AutoDRM) requests.