Draft Utility Program Requirements

1. A utility program shall be provided that allows for independent validation of electronically signed data by taking an input file specified by the user and, using the electronic signature information generated by the standardized electronic signature system contained in the file, validate that the data has not be altered.

    1.1. The user shall be able to specify the delimiter used to separate the electronic signature information from the signed data.

    1.2. The file data may be either ASCII or binary, depending on the user's needs, with each record being separated by a carriage return.

    1.3. The signature validation process may use either the certificate revocation list (CRL) used for the production system or a CRL cached on the device performing the validation. (When using the cached CRL capability, the utility user should ensure that the cached CRL is current).

    1.4. The program shall clearly identify the records that failed validation and, using the standard error codes provided elsewhere in this document, provide the first reason that a record failed the verification test. The system is not responsible for determining whether additional problems would cause verification errors after the system has detected a failure.

    1.5. The program shall produce summary information that can be used to help ensure that all transactions were processed. The summary information must include such items as (1) number of transactions processed, (2) number of transactions validated, and (3) number of transaction failing verification.

2. A utility program shall be provided that allows for analysis of the critical data characteristics associated with a given signature transaction. It may be assumed that the transaction data used by this program has already been validated using the utility program described in Requirement 1 above or that additional validation is not needed. The result of this program is an output file that can be further manipulated by the user. How the output file is used is outside the scope of this requirement.

    2.1. The user shall be able to specify the delimiter used to separate the electronic signature information from the signed data and support from 0 to 10 transaction-identifying elements. The responsibility of determining the number and type of transaction identifying elements, e.g., transaction type, transaction identification number, etc., is the responsibility of the user and outside the scope of this requirement. However, the user information may be up to 250 alphanumeric characters for each identifying element.

2.2. The file data may be either ASCII or binary with each record being separated by a carriage return.

2.3. The utility program shall, using the electronic signature information that has been signed by the entity generating the electronic signature, provide the following information:

2.3.1. Information adequate to determine the entity that is bound to a given signature.

2.3.2. The algorithm that was used to generate the message digest.

2.3.3. The algorithm that was used to generate the electronic signature.

2.3.4. The date and time that a signature was generated.

2.3.5. The assurance level associated with the signer's certificate.

2.4. The utility program shall, using the information obtained in requirement 2.3 above and the data file provided by the user, produce an output file that can be used for additional analysis by the user. How the output file is used is outside the scope of this requirement.

2.4.1. The user shall be able to specify the output file name and destination.

2.4.2. The user shall be able to specify the delimiter used to separate the data elements in the file.

2.4.3. The file data may be either ASCII or binary, as specified by the user, with each record being separated by a carriage return.

2.4.4. The following data elements must be provided in the output file using the delimiter specified by the user in requirement 2.4.2 above: (1) entity signing the transaction, (2) the algorithm that was used to generate the message digest, (3) the algorithm that was used to generate the electronic signature, (4) the date and time that a signature was generated, (5) the assurance level associated with the signer's certificate, and (6) each of the transaction identifying elements that was provided in the user's data file separated by the delimiter specified by the user in requirement 2.4.2. The data contained in elements 1 through 5 must come from the process outlined in requirement 2.3 above.

3. A program shall be provided that allows for the verification that critical files associated with the electronic signature system or other systems have not been altered after they are approved for production.

3.1. The program shall allow a user to input a file name and directory path and, using the standardized electronic signature system, electronically sign the file and maintain the critical information necessary to identify the file and its corresponding electronic signature. The data elements retained by the system include (1) the file name, (2) the version (as shown in the configuration management system), (3) the date the program was released into production, and (4) an electronic signature that complies with the standardized electronic signature system.

3.2. The program shall allow for independent validation of files signed by the program in requirement 3.1.

3.2.1. The program shall allow a user to input a file name and directory path and, using the standardized electronic signature system, validate that the file identified by the user has an identical electronic signature to the file that was signed in requirement 3.1.

3.2.2. The program shall clearly identify the records that failed validation and, using the standard error codes provided elsewhere in the document, provide the first reason that a record failed the verification test. The system is not responsible for determining whether additional problems would cause verification errors after the system has detected a failure.

3.3. The program shall produce a report showing the items contained in its database and the data maintained about each file that is being maintained. The report shall be sorted by file name.

4. A utility program shall be provided that can be used to determine the time necessary for the electronic signature system to complete its operations.

4.1. The program shall allow a user to specify a file name, including the directory path, that should be timed, the name of the user performing the test, and at least 10 other data elements determined by the user that should be included in the test report. The 10 data elements that can be entered by the user are outside the scope of this requirement. However, the user should be allowed to enter at least 250 characters of alphanumeric data for each element.

4.2. The program shall allow the user to specify the standardized electronic signature function that should be tested, e.g., sign the file identified in requirement 4.1. This program must support each function, commonly referred to as a high level call, that is considered available in the standardized electronic signature system.

4.3. The program shall use the standardized electronic signature system, and determine the time necessary for the standardized electronic signature system to return its results to this program.

4.4. The program shall produce a report showing the results of its operation. The data included shall include (1) the function being tested, (2) the file that was used for the testing, (3) date and time the test was executed, and (4) the data elements entered by the user under requirement 4.1.

4.5. The program shall maintain a log of its testing results.

4.5.1. The testing log shall contain all the items necessary to regenerate the testing report in described in requirement 4.4.

4.5.2. The program shall allow the user to delete one or more log entries by either (1) selecting a specific log item, (2) selecting log entries by specifying a beginning and ending date/time range, or (3) selecting all items.

4.5.3. The program shall allow the user to print one or more log entries by either (1) selecting a specific log item, (2) selecting log entries by specifying a beginning and ending date/time range, or (3) selecting all items. The report produced by the system shall contain the data obtained in requirement 4.4.