|  |  |
| --- | --- |
| **CROMERR System Checklist** | |
| **Item** | **GENERAL NOTE:** This checklist is provided as a template to authorized programs implementing the Windsor nForm system. Local control authorities implementing this system require review and approval by EPA under CROMERR Part 3. An indirect regulated utility may require approval by the state regulatory agency. This checklist describes implementation of the nForm system. The control authority must provide additional detail where indicated in blue text. Also, where a control authority chooses to deviate from the system as described in this application, those deviations must be described in the checklist below. It is generally helpful in EPA’s review of your application if you mark deviations from the template in text that is of a different color. |
| **Registration (e-signature cases only)** | |
| **1. Identity-proofing of registrant** | |
|  | **Business Practices:**  Per CROMERR 3.2000(b)(5)(vii)(C), the receipt of a signed Subscriber Agreement is sufficient proof of a user's identity. Reference Item 1 b-alt for more information on the Subscriber Agreement. The **(Insert Local/Agency Acronym)** will review the information provided in the Subscriber Agreement and perform additional identity proofing.  The information included in the Subscriber Agreement is compared to that of the facility's contact information, and the Subscriber Agreement includes documentation supporting proof of authority for the facility's responsible official. If the **(Insert Local/Agency Acronym)** has any questions about the persons identified as potential users of the **(Insert System Name)** System, the facility is contacted and the information is verified. **(Revise or insert any additional processes as needed).** |
|  | **System Functions:**  For identity proofing registrants, the **(Insert Local/Agency Acronym)** will be using a Subscriber  Agreement approach. Reference Item 1 b-alt for more information. |
| **Supporting Documentation (list attachments):**  N/A |
| **1a. (priority reports only) Identity-proofing *before* accepting e- signatures** | |
|  | **Business Practices:**  Reference Item 1 for how identity proofing will be performed using a Subscriber Agreement. Reference Item 1 b-alt for more information about the data contained in the Subscriber Agreement, how the user will provide the information, and the identity proofing processes used by **(Insert Local/Agency Acronym)**. |
|  | **System Functions:**  The **(Insert System Name)** System will not allow a user to electronically sign reports until the Subscriber Agreement has been received and verified by the appropriate regulating authority. Reference Item 1 b-alt for more information about the data contained in the Subscriber Agreement and how the user will provide the information. |
| **Supporting Documentation (list attachments):**  - CROMERR Reports Inventory (Attachment A) |
| **1b. (priority reports only) Identity-proofing method (See 1bi, 1bii, and**  **1b-alt)** | |
| **1bi. (priority reports only) Verification by attestation of disinterested individuals** | |
|  | **Business Practices:**  N/A - Reference Item 1 b-alt. |
| **System Functions:**  N/A - Reference Item 1 b-alt. |
| **Supporting Documentation (list attachments):**  N/A - Reference Item 1 b-alt. |
| **1bii. (priority reports only) Information or objects of independent origin** | |
|  | **Business Practices:**  N/A - Reference Item 1 b-alt. |
| **System Functions:**  N/A - Reference Item 1 b-alt. |
| **Supporting Documentation (list attachments):**  N/A - Reference Item 1 b-alt. |
| **1b-alt. (priority reports only) Subscriber agreement alternative** | |
|  | **Business Practices:**  Per CROMERR, Subscriber Agreements with wet-ink signature will be stored at **(Insert Local/Agency Acronym),** per Federal or State regulations, for at least five years after the associated signature credential has been deactivated. Reference Item 1 and 1 a for how the Subscriber Agreement meets the identity proofing requirement, and reference Item 2 for how the Subscriber Agreement is used by the **(Insert Local/Agency Acronym)** to determine the requestor's signing authority. |
|  | The **(Insert Local/Agency Acronym)** will keep signed Subscriber Agreements in a secured room or locked file cabinet. **(Insert additional agency due diligence for physical security)**. As stated above, at the minimum, **(Insert Local/Agency Acronym)** will store the Subscriber Agreement per Federal or State regulations, whichever is longer. **(Insert Local/Agency Acronym)** will utilize 10 years for their minimum record retention period. |
| **System Functions:**  Per CROMERR, a Subscriber Agreement is "an electronic signature agreement signed by an individual with a handwritten signature". The **(Insert System Name)** System will allow the user to download a Subscriber Agreement form from the application web site. Refer to: Electronic Subscriber Agreement Form, Attachment 1 for an example. If the **(Insert Local/Agency Acronym)** prefers, the registration forms download capability can be disabled, and the user would be required to contact the **(Insert Local/Agency Acronym)** directly to request the Subscriber Agreement forms from the **(Insert Local/Agency Acronym)**.  The Subscriber Agreement also contains language requiring the user to protect their signing credentials, not to share signing credentials with anyone else, and immediately report any compromise of the credentials to the **(Insert Local/Agency Acronym)**. Reference Item 4 for more information about the Subscriber Agreement content.  After printing, completing, and signing, the Subscriber Agreement form will be mailed to the **(Insert Local/Agency Acronym)**. A Submitter will not be able to sign electronic documents until the Subscriber Agreement has been received by the **(Insert Local/Agency Acronym)** which has then verified the information, and set the application's Subscriber Agreement flag/indicator, associated with the user account, for the user requesting the Submitter role.  When the paper copy of the Subscriber Agreement is received, a designated **(Insert Local/Agency Acronym)** staff will confirm specific information such as Name, Address, Phone Number, Email address match the information entered on the electronic registration page within the **(Insert System Name)** System. When verified, the subscriber will be given access to continue with an electronic application submission via their email address. The **(Insert Local/Agency Acronym)** will retain a paper copy of the Subscriber Agreement on file as specified in Item b-alt.  Reference Item 3 for a description of how user accounts are created. |
| **Supporting Documentation (list attachments):**  - Electronic Subscriber Agreement Form (Attachment 1). |
| **2. Determination of registrant's signing authority** | |
|  | **Business Practices:**  **(Insert Local/Agency Acronym)** uses the following procedures to determine the signing authority of users who will sign submissions to the **(Insert System Name)**:  The permit application (new and renewal) process, which includes the following steps to establish the signature authority of corporate officers:  1. Receive Permit Application. This Application must include a completed **(Insert Local/Agency Acronym)** Disclosure Statement. **(Revise or insert any additional processes as needed).**  2. The application is reviewed for Administrative Completeness, which includes a review of the signatures on all of the submitted application forms. All application forms must be signed by a Responsible Official as defined in 40 CFR 122.22(a).  3. Any signatures on the Permit Application forms are compared. **(Revise or insert any processes as needed)**  4. The facility is notified that their application is Administratively Complete, signifying that the signatures are acceptable. **(Revise or insert any processes as needed)**  For users who are not corporate officers, **(Insert Local/Agency Acronym)** adds the following procedures to establish signatory authority for a permitted facility:  1. The facility must submit documentation, in accordance with corporate procedures (a letter signed by corporate officer or resolution for the board of directors), designating a person or position with signatory authority for the facility.  2. The individuals authorizing the designation are compared to the listing of corporate officers on the Secretary of State website or the board of directors as listed on the corporate website or SEC filings. **(Revise or insert any processes as needed)**  3. The facility is sent a letter confirming that **(Insert Local/Agency Acronym)** has accepted the signatory authority designation. **(Revise or insert any additional processes as needed)**  In addition, the **(Insert Local/Agency Acronym)** must receive a signed Subscriber Agreement from each user that is requesting the ability to sign and submit electronic reports. The **(Insert Local/Agency Acronym)** will validate the information provided in the Subscriber Agreement to assure accuracy and deem that it is appropriate for the requestor to be granted signatory authority.    Reference Item 3 for a description of how user accounts are created. **(Revise or insert any additional processes as needed)** |
|  | **System Functions:**  Reference Item 1 b-alt for more information about the data contained in the  Subscriber Agreement. |
| **Supporting Documentation (list attachments):**  N/A |
| **3. Issuance (or registration) of a signing credential in a way that protects it from compromise** | |
|  | **Business Practices:**  Reference Item 1 b-alt, Item 2 and Item 3 for business processes followed to process received Subscriber Agreements. |
| **System Functions:**  The **(Insert System Name)** System provides the following functions to create and securely issue signing credentials. Signing credentials essentially are comprised of the unique user account login, the active password associated with the user account, and the active challenge questions and answers associated with the user account.   1. Users of the **(Insert System Name)** System create a user account via a registration process. Attachment 2: Create User Profile for an example. This registration page where users provide information about themselves such as their name, phone number, physical address and a unique identifier, in the form of the user's email address, in the system for the account login. No two users can have the identical user account logins, and account logins cannot be re-used if the original account login is no longer needed or used by a user. There is a unique database index on the account login field within the database, guaranteeing unique account logins within the system. If a duplicate login is attempted to be entered, the system will reject the duplicate account login and generates an error. 2. When registering a user account, the user will enter their desired system password. Refer to Attachment 2A: Create User Profile for an example. Passwords that do not meet the password format/strength requirements will be rejected, and the user will be prompted to re-enter a valid password. The settings used by the **(Insert Local/Agency Acronym)** for password strength parameters are:  * Must be at least 8 alpha-numeric characters * Must include at least one lower case letter * Must include at least one upper case letter * Must include at least one numeric digit * Must include at least one special character * Must not have been used by the user before.   A history of all passwords used by the unique user account login is maintained in the database. At a minimum, **(Insert Local/Agency Acronym)** will store the passwords per the same retention requirements of the CoR with which the password is associated. See item 20 (last paragraph) for CoR retention details.  Note: The password is not unique to the database, only to the user. The signing credential is unique due to the fact the each user account login is unique, and each user specifies their own password, which is then hashed in the database. So the combination of user account login and password is always unique. Only one password can be "active" at any time for the user. Password uniqueness is enforced through a unique database index on the account and password (the combination of the two properties must be unique) within the database, guaranteeing unique passwords per user. If a duplicate password is attempted to be entered, the system will reject the duplicate password and generate an error. In addition, the date-time that the password was created, and the date-time that the password expired (the end date) is also recorded in the database.   1. If the entered password adheres to the **(Insert Local/Agency Acronym)** password requirements, the user's password will be stored in encrypted format in the database and the user will be prompted to sign into the system. The system will also send a Password Reset email confirmation to the user. Refer to Attachment 5 & 6: Password Reset and eMail Notification for an example.   The password is stored as one way, "salted", hash, using the SHA- 2 512 bit algorithm. If needed, the algorithm can utilize the Bcrypt or SHA-3 algorithms, via confirmation setting. Passwords are protected from deletion or alternation through hashing, with the only two database accounts with read- write access to the table being the application's database user account and the **(Insert Local/Agency Acronym)** Database Administrator of the database.  A history of password sets/changes are written to a database event log, available only to the application's database user account (READ, INSERT access only) and the **(Insert Local/Agency Acronym)** Database Administrator of the database. Since this information is not part of the signing credentials or CoR, but simply a history of system function use, it can be purged at the discretion of the **(Insert Local/Agency Acronym)**.  The identical event information is also written to a text-based log file on the web server available only to **(Insert Local/Agency Acronym** personnel who are not **(Insert Local/Agency Acronym)** Database Administrators on the **(Insert System Name)** database server. If a password was improperly or maliciously changed in the database, an audit of the text-based log file would reveal no entry for its change, so its absence would indicate improper use or malicious intent on the part of the **(Insert Local/Agency Acronym)** Database Administrator. Since this information is not part of the signing credentials or CoR, but simply a history of system function use, it can be purged at the discretion of the **(Insert Local/Agency Acronym)**.   1. The **(Insert System Name)** System will send, via automated email, a notification to the user of their account creation, along with a hyperlink used to confirm their email address. The user will need to click on the link to confirm their account in order to enable their user account. Until the user account is confirmed, the user will not be able to login to the account. 2. If a Subscriber Agreement has been received for the user, and verified by the **(Insert Local/Agency Acronym)**, **(Insert Local/Agency Acronym)** will use the **(Insert System Name)** System to specify that the user is approved to electronically sign form submissions and that their Subscriber Agreement is complete and accepted. The system will be assigning the user to the Electronic Signature rights. Refer to Attachment 3: User Registration complete and eMail Notification for an example of how user accounts are created and a Subscriber Agreement is acknowledged as received and verified. Refer to Attachment 4: Account Confirmation and First Time Login Screen. 3. Following a successful login to the system, if the agency has accepted and approved their Subscriber Agreement and the user has Electronic Signature rights, the user is prompted to define their challenge questions and answers. The user is not forced to perform this step immediately; however, the system will not allow the user to certify and submit a form (which requires an electronic signature) until this step is performed. 4. If the user does choose to setup their challenge questions and answers after their login, they will access their user profile, available as a hyperlink from within the application. The user is presented with approximately 22 possible challenge questions, of which they are required to provide answers to five (5) unique questions, with answers of at least five (5) characters in length. Refer to Attachment 7 & 7A: Challenge Questions List and Selected Questions for example. A unique question check and length check is implemented when the user attempts to save their challenge questions and answers. In addition, the answer to each challenge question must be unique (e.g., a user could not enter "ABC DE" to every question). If it does not pass these checks, the save will be canceled and descriptive error text displayed to the user. The date-time that the challenge question answer was created, and the date-time that the challenge question answer expired (the end date) is also recorded in the database. A history of all challenge questions and answers used by the user account, along with their effective dates, is retained in the database. At a minimum, **(Insert Local/Agency Acronym)** will store the challenge questions/answers per the same retention requirements of the CoR that the challenge questions/answers are associated with. See item 20 (last paragraph) for CoR retention details.   Once the challenge questions have been selected, answered, and successfully saved, the user is unable to view or change the questions and answers. They may contact the **(Insert Local/Agency Acronym)** to have their challenge questions reset, which will "expire" all the user account's current challenge questions and answers, requiring them to repeat steps 6 and 7. The challenge question reset function is available to **(Insert System Name)** Administrator accounts from the User Management module of the application. The **(Insert System Name)** Administrator will first verify the identity proofing requirements outlined in item 1,1b -alt, and 2 are satisfied before performing a reset operation.  Challenge questions are stored as one way, "salted", hash, using the SHA-2 512 bit algorithm. If needed, the algorithm can utilize the Bcrypt or SHA-3 algorithms, via confirmation setting. Challenge questions are protected from deletion or alteration through hashing, with the only two database accounts with read- write access to the table being the application's database user account and the **(Insert Local/Agency Acronym)** Database Administrator of the database.  Access to the configuration file is limited to the application server's local Administrator account. Only the **(Insert Local/Agency Acronym)** System Administrator for the application server will have the credentials to this account. No **(Insert Local/Agency Acronym)** Database Administrator will have access to this account, and thus will have no access to the configuration file. Likewise, the **(Insert Local/Agency Acronym)** System Administrator will have no access to the **(Insert System Name)** database where the challenge questions/answers, passwords, and CoRs are stored.  Only two database accounts (the application's database user account and the **(Insert Local/Agency Acronym)** Database Administrator for the database) have access to the table containing encrypted challenge question answers.  Setting challenge question answers can only be initiated by a user from within the application. This is the only mechanism to set challenge question answers.   1. A password can be changed in three ways:  * A password can be reset by a user from within the application by entering their user email address and requesting that the password be reset at which time **(Insert System Name)** System will randomly choose, using a random number generator, one of the five (5) challenge questions answered in the database for the user's account (for specific details see C# Random class; range used is 0 to Number of Security Questions answered - 1). The user must enter the answer to the presented question to reset their password. If a challenge question is incorrectly answered, **(Insert System Name)** System will automatically cycle to the next challenge question.   The system will provide an automatic lockout mechanism based on a configurable maximum number of reset password (challenge question) attempts, with 5 being minimum setting.  Correct and incorrect challenge question response attempts (user, date-time, success or fail indicator) are logged to the application audit table as well as the text-based log file.  If a challenge question is correctly answered, a new password is randomly generated by the system and emailed to the user's email address.  Note: For users who are not Electronic Signatories, they will not be required to enter· a challenge question answer to request a password reset.   * + An Administrator or Organization Manager may reset a user's password, when required. When an Administrator or Organization Manager initiates the password reset, a new password is randomly generated by the system and emailed to the user's email address.   After a password is reset, the user will be required to change their password when attempting to login into the system for the first time before proceeding with system use. The user will be required to enter their current password as well as their new compliant password.   * + A user can change their password. When attempting to change the password, the user will be required to enter their current password as well as their new compliant password.   A history of password and challenge question-answer set/change are written to a database event log, available only to the application's database user account (READ, INSERT access only) and the **(Insert Local/Agency Acronym)** Database Administrator of the database. The user who changes the value, the time the value was changed as well as the changed value itself will be saved to this log. These data elements for any historical password or challenge question-answer will be maintained in the log for as long as the system maintains the CoRs that were signed using that password and challenge question-answer – so that these historical signing credentials can be made available to revalidate the signatures in the associated CoRs as needed.  The identical event and historical signing credential information is also written to a text-based log file on the web server available only to **(Insert Local/Agency Acronym)** personnel who are not **(Insert Local/Agency Acronym)** Database Administrators on the **(Insert System Name)** database server. This information is maintained in the text-based log for as long as it is maintained in the database event log. Access to both these logs is limited to a small number of **(Insert Local/Agency Acronym)** personnel, and no user will have access to both logs. The two logs help ensure the integrity of the data. For example, if a password was improperly or maliciously changed in the database event log, an audit of the text-based log file would reveal no entry for its change, so its absence would indicate improper use or malicious intent on the part of the **(Insert Local/Agency Acronym)** Database Administrator.  Note: All user session communication is protected through SSL, including during account set up, and where the password and challenge questions and answers are established. The **(Insert Local/Agency Acronym)** currently uses an SSL certificate from GeoTrust called True BusinesslD that provides business identity authentication, strong 256-bit encryption 2048-bit root. Refer to item 5 for additional details regarding the SSL Certificate used by the **(Insert Local/Agency Acronym)**. |
|  | **Supporting Documentation (list attachments):**   1. Creating user account (Attachments 2 & 2A) 2. User Registration complete and New User Registration email notification (Attachment 3) 3. Account Confirmation, First Time Login and Welcome Screen (Attachment 4 & 4A) 4. Password Reset Request (Attachment 5) 5. Password Reset eMail Notification (Attachment 6) 6. Challenge Question List and Selected Questions example (Attachment 7 & 7 A) |
| **4. Electronic Signature Agreement** | |
|  | **Business Practices:**  N/A |
| **System Functions:**  The **(Insert System Name)** system will use a Subscriber Agreement, defined as "an electronic signature agreement signed by an individual with a handwritten signature". The content of the Subscriber Agreement is described in Item 1 b-alt.  The **(Insert Local/Agency Acronym)** will keep signed electronic signature agreements in a secured room or locked file cabinet. **(Revise or insert any additional language describing retention period for ESAs)**  **(Describe how ESAs are stored in a way that protects them from tampering, destruction, and unauthorized access)**  If the user has the Submitter role, the Subscriber Agreement acknowledgement has been indicated, and the user has set up their challenge questions and answers, they can proceed with an electronic submission. Assuming the form submission data passes all applicable business/QA rules checked when initiating the submission, the **(Insert System Name)** system will present a Certify and Submit page.  The Certify and Submit page consists of agency-defined electronic signature agreement criteria that each submitter must agree to before they can proceed with the form submission process. The user must individually acknowledge each agreement on the screen before they are allowed to continue. |
| **Supporting Documentation (list attachments):**   1. **(Insert System Name)** will require an RO to print out and mail in the Subscriber Agreement to **(Insert Local/Agency Acronym)**. 2. The standard ESA language in the Subscriber Agreement states:   **(Insert standard ESA language)**   1. By signing the agreement, user agrees to:   **(Insert relevant ESA language)**   1. For the ESA templates for **(Insert System Name)**, please refer to **(Insert Attachment Number)**. |
| **5. Binding of signatures to document content** | |
|  | **Business Practices:**  N/A |
| **System Functions:**  The **(Insert System Name)** electronic signature process involves multiple steps. **(Insert System Name)** allows users to submit only one electronic form within a single transaction. The **(Insert System Name)** System also generates a unique Copy of Record (CoR) for each electronic form that is submitted via the system.  Signing Process  If the user has the Electronic Signature role, the Subscriber Agreement acknowledgement has been indicated, and the user has set up their challenge questions and answers, they can search for and initiate a form submission. Assuming the form data passes all applicable business/QA rules checked when initiating the form submission, and the user has agreed to all conditions on the Certify and Submit page, the user is prompted to enter their electronic signature.  **(Insert System Name)** will randomly choose, using a random number generator, one of the five (5) challenge questions answered in the database for the user's account (for specific details see C# Random class; range used is o to Number of Security Questions answered - 1). The user must enter the answer to the presented question to sign the form submission. If a challenge question is incorrectly answered, **(Insert System Name)** System will automatically cycle to the next challenge question.  In addition to the challenge question, the user must also re-enter their **(Insert System Name)** account password at the time of signing (see Attachment 8).  The system will provide an automatic lockout mechanism based on a configurable maximum number of electronic signature (challenge question + password) attempts, with 5 being minimum setting.  Correct and incorrect challenge question and password response attempts (user, date-time, success or fail indicator) are logged to the application audit table as well as the text-based log file discussed in Item 3.  Electronic Signature  The **(Insert System Name)** digital signature process will support any XS09-compatible certificate. X509-compatible certificates may be obtained through a 3rd party certificate authority (CA). Certificates can be configured to start/expire on specific dates, and can also be revoked by the issuer of the certificate. **(Insert System Name)** System currently uses the PKCS#12 (PFX) type.  Changing to another/additional types is not currently planned.  Once configured, the certificate must be securely stored, as possession of the certificate by un-authorized, malicious sources could allow the signed documents to be modified. Certificates are unique to the application install itself, and not users. The certificate is stored on the same server as the **(Insert System Name)** application. Access to the certificate is limited to the **(Insert Local/Agency Acronym)** System Administrator(s).  The **(Insert Local/Agency Acronym)** currently uses an SSL certificate from GeoTrust called True BusinesslD that provides business identity authentication, strong 256-bit encryption 2048-bit root.  Note that the electronic signature certificate is different from that used by the **(Insert System Name)** solution to secure its communication (which uses the SSL certificate).  Certificates are not given to "users". A single certificate is used by the application to electronically sign submitted forms (see Item 9b for additional details on file formats). No check for expiration date of the certificate is currently performed at the time of signature. It is assumed to be valid since the certificate is unique to the **(Insert System Name)** application and not users.  The unique user account login, password, challenge question and challenge question response is used as the electronic signature device. **(Insert System Name)** application will use its private certificate key to digitally sign the hash of the signature device and the CoR to bind the electronic signature to the submitted form. |
| **Supporting Documentation (list attachments):**  **(Insert System Name)** Subscriber Agreement Template  **(Insert System Name)** Account Registration Instructions  **(Insert State Name)** State Government IT Policies, Standards, and Guidelines |
| **6. Opportunity to review document content** | |
|  | **Business Practices:**  N/A |
| **System Functions:**  During the form submission process, the system will generate a read- only representation of the form submission and present it on the screen for the Submitter can view the form submission before signing. The read-only representation of the form submission includes all data contained within the form submission as well as the ability to download and/or open any related attachments that the Submitter included in their submission. The Submitter must acknowledge that they have reviewed the form submission prior to completing the form submission process. |
| **Supporting Documentation (list attachments):**  - Review Form Submission Screen (Attachment 8) |
| **7. Opportunity to review certification statements and warnings** | |
|  | **Business Practices:**  N/A |
| **System Functions:**  During the form submission process, the Submitter will be required to select all checkboxes indicating that they understand and agree to the electronic signature and form submission conditions. The submitter cannot continue the form submission process and sign the form submission until all checkboxes are selected. This function is further described in Item 4.  The following agreements must be met:   * I am the owner of the account used to perform the electronic submission and signature. * I have the authority to submit the data on behalf of the facility I am representing. * I agree that providing the account credentials to sign the submission document constitutes an electronic signature equivalent to my written signature. * I have reviewed the electronic form being submitted in its entirety, and agree to the validity and accuracy of the information contained within it to the best of my knowledge.   **(Insert additional information regarding ability to add additional agreements or data)**.  The following certification statement will also be presented to the  Submitter:  "I certify that I have not violated any term in my Electronic Subscriber Agreement and that I am otherwise without any reason to believe that the confidentiality of my user ID and/or password have been compromised now or at any time prior to this submission. I understand that this attestation of fact pertains to the implementation, oversight, and enforcement of a federal environmental program and must be true to the best of my knowledge.  I certify under penalty of law that this document was prepared under my direction or supervision in a manner to assure that qualified personnel gathered and evaluated the information submitted. Based on my inquiry of the person(s) who gathered and evaluated the information and of the person(s) responsible for managing the regulated activity, the information submitted is, to the best of my knowledge and belief, true and accurate and complete. I am aware that submitting false information is punishable by up to 5 years in prison and a fine of $50,000."  The certification statement presented to the signer, including warning of penalties for false certification, is incorporated into the copy of record for the signed submission. |
|  | **Supporting Documentation (list attachments):**  - eSignature Form and Agreements and Completed Form (Attachment 9) |
| **8. Transmission error checking and documentation** | |
|  | **Business Practices:**  N/A |
| **System Functions:**  The integrity of the form submission is protected by the following:  1. No alteration of the form submission content is expected during transmission or after it is received by the **(Insert Local/Agency Acronym)**.  2. **(Insert System Name)** utilizes SSL for the entire form submission process, protecting the system and submission against man-in-the-middle attacks. The **(Insert Local/Agency Acronym)** currently uses an SSL certificate from GeoTrust called True BusinesslD that provides business identity authentication, strong 256-bit encryption 2048-bit root. Refer to item 5 for additional details regarding the SSL Certificate used by the **(Insert Local/Agency Acronym)**.  3. The submitter is sent an email notification after each form submission. This email contains an electronically signed CoR attachment, as well as the unique submission number. If the Submitter does not receive this confirmation email and CoR attachment, that is an indication that the form was not received, and the **(Insert Local/Agency Acronym)** can investigate.  4. The information used to populate the read-only representation of the form submission, reviewed by the Submitter during a form submission, is the exact information used to complete the form submission. No updates to that data previewed can be made after the submission process begins.  5. The CoR contains the exact data used to populate the read-only view of the form submission, reviewed by the Submitter during a form submission.  6. The CoR contains a header page with meta-data from the submission process, including date and time of submission, submission number and submitter name. A watermark indicating the certificate authority used and fingerprint (a unique certificate number) for the electronic signature is also displayed. No passwords, challenge questions/answers, or any other sensitive information is displayed on this header page. The header page is included in the CoR strictly as a clear way of visibly indicating to any viewer of the CoR that the CoR has been successfully electronically signed. The meta-data recorded on the header page is retrieved from the database, so it's not the sole source of this information.  Once a form submission has been signed, it is protected from transmission errors or other changes by the electronic signature itself. |
| **Supporting Documentation (list attachments):**  **(Insert System Name)** Subscriber Agreement Template  **(Insert System Name)** Account Registration Instructions |
| **9. Opportunity to review copy of record** | |
| **9a. Notification that copy of record is available** | |
|  | **Business Practices:**  N/A |
|  | **System Functions:**  Submitters are informed about the CoR in several of the following ways:  1. **(Insert System Name)** System presents the Submitter with a confirmation page and a unique Submission Number, following the form submission.  2. The submitter is sent an email notification after each form submission. This email contains the unique confirmation number and a description of where to download the CoR within the system.  3. Submitters have the ability to view or download the electronically signed copy of the CoR at any time for any form submission (where they are assigned as a contributor to the form submission) from the Submission View page of the **(Insert System Name)** system. |
| **Supporting Documentation (list attachments):**  - Submission Confirmation (Attachment 10) |
| **9b. Creation of copy of record in a human-readable format** | |
|  | **Business Practices:**  N/A |
| **System Functions:**  The electronically signed CoR file created for each submitted form will contain the reported data, header page, related attachments (if applicable) and bound electronic signature. The electronically signed version of the CoR is also used for verification of signature authenticity, and that no modification to the CoR has been made since initial creation.  The electronically signed CoR file created for each submitted form will be in the form of a ZIP or PDF file, depending on whether attachments are included in the submission. If no attachments are included in the submission, the submitted form will include one PDF file representing the reported data. This PDF file will include the certificate. If attachments are included in the submission, the submitted form will include one ZIP file which will include one PDF file representing the reported data and all attachments included in the submission. This ZIP file will include the certificate.  The CoR (i.e., PDF and associated attachments) is stored as a two-way hash, using the SHA-2 512 bit algorithm. If needed, the algorithm can utilize the Bcrypt or SHA-3 algorithms, via confirmation setting. CoR's are protected from deletion or alternation through hashing, with the only two database accounts with read-write access to the table being the application's database user account and the **(Insert Local/Agency Acronym)** Database Administrator of the database.  When providing the human readable CoR to a user for download and access, the CoR is first decrypted, using the decryption key. The decryption key is stored in the application configuration file. Access to the configuration file is limited to the application server's local Administrator account. Only the **(Insert Local/Agency Acronym)** System Administrator for the application server will have the credentials to this account. No **(Insert Local/Agency Acronym)** Database Administrator will have access to this account, and thus will have no access to the configuration file.  There is no separate step required to make the CoR human readable, except that the browser or local user's computer must be capable of opening a ZIP files (if applicable) and rendering PDF documents as well as well as any applicable attachments provided by the Submitter. |
| **Supporting Documentation (list attachments):**  N/A |
| **9c. Providing the copy of record** | |
|  | **Business Practices:**  N/A |
| **System Functions:**  Submitters are notified about the creation and storage of the CoR in an email acknowledgement. Any user with access to the form submission can access the electronically signed CoR via the button on the form submission's view page, which can be accessed from the User's submission history page. |
| **Supporting Documentation (list attachments):**  N/A |
| **10. Procedures to address submitter/signatory repudiation of a copy of record** | |
|  | **Business Practices:**  It is not possible or desired to change a signed CoR. Instead, if the Submitter needs to modify/revise their submission, they can submit a revision to their form data and re-submit the form. The most recent form submission then becomes the official CoR for the form, but does not replace or delete any existing report CoR. See item 20 (last paragraph) for CoR retention details.  If the **(Insert Local/Agency Acronym)** finds an error or omission in a submitted form, they will contact the Submitter and request that they re-submit the form as described above.  There is also the ability to request that a signed CoR be marked as "rescinded". The **(Insert Local/Agency Acronym)** users can approve or deny a rescind request. If it is approved, the submitted form CoR is "rescinded" or repudiated, and the Submitter is notified via email with details of the report CoR that was marked as repudiated. If the repudiation request is denied, the  Submitter is also notified via email, with the reason for the denial.  For instance, if the submission of the document was in error/accidental, or credentials were determined to be compromised, the Submitter can contact **(Insert Local/Agency Acronym)** to request that the form submission CoR be rescinded. Note that only the **(Insert System Name)** internal staff members, not submitters, will be able to mark the form submission as rescinded after reviewing the request made by the Submitter, and either denying the request, or granting the request which results in the CoR being rescinded. |
| **System Functions:**  The signed CoR is stored in a dedicated database table. There is no function in **(Insert System Name)** that allows the CoR to be modified or deleted, and CoRs will be retained indefinitely. Refer to item 20 for details regarding the maintenance of the CoR.  Submitters can contact **(Insert Local/Agency Acronym)** to request, along with specifying the reason for the request, that any previously submitted form submission be rescinded. If it is approved, the submitted form CoR status is changed to 'Rescinded', and the Submitter is notified via email that the submission was marked as "rescinded".  The user may use the View Submission History function (refer to Item  19 System Functions), to view all submissions and their status, including any forms submissions that were rescinded.  Note: A history of rescinded requests is written to a database event log, available only to the application's database user account (READ, INSERT access only) and the **(Insert Local/Agency Acronym)** Database Administrator.  The identical event information is also written to a text-based log file on the application server available only to agency personnel who are not **(Insert Local/Agency Acronym)** Database Administrators on the **(Insert System Name)** database server. If a rescind of a CoR was improperly or maliciously performed in the database, an audit of the text-based log file would reveal no entry for this change, so its absence would indicate improper use or malicious intent on the part of agency personnel with Database Administrator rights in the database. |
| **Supporting Documentation (list attachments):**  N/A |
| **11. Procedures to flag accidental submissions** | |
|  | **Business Practices:**  Accidental submissions such as duplicate forms and deviations from normal content or procedure would be detected through normal **(Insert Local/Agency Acronym)** quality assurance checks/processes performed by the **(Insert Local/Agency Acronym)** personnel for each submitted form.  If an accidental submission is detected, **(Insert Local/Agency Acronym)** personnel will investigate the cause of the accidental submission, typically involving a telephone call to the responsible official, using the telephone number on record with the **(Insert Local/Agency Acronym)**. Also refer to item 15 for additional details on how **(Insert Local/Agency Acronym)** may identify and investigate suspect submissions, and the resulting actions they may take.  If a user determines that they accidentally submitted a form, the submission will be rescinded through the rescind request process. See Item 10 for the rescind process and system functions.  The rescinded form submission will still be accessible using the View Submission History (described in item 10 and 19) system function however the submission status will be indicated as "Rescinded" and the user will be unable to revise the form submission. |
| **System Functions:**  Refer to item 10 System Functions. |
| **Supporting Documentation (list attachments):**  N/A |
| **12. (e-signature cases only) Automatic acknowledgment of submission** | |
|  | **Business Practices:**  If the **(Insert Local/Agency Acronym)** receives a message from **(Insert System Name)** or their internal email system that there was an undeliverable email message, then the agency will call the phone number associated with the user account that made the submission. If they cannot contact the person directly, then the agency will contact the responsible official, using the telephone number on record at the **(Insert Local/Agency Acronym)**, and determine if they are still employed at the company. |
| **System Functions:**  **(Insert System Name)** sends an acknowledgement email to the email address of the Submitter after every submission. The email will contain the Submitter's name, date and time of submission, subject of email, as well as a unique Submission Number, so that the Submitter can further identify the form submission in question.  The event log (both database and file system based that has been previously described) is inserted with an entry from each step of the submission process, including that the acknowledgement email was sent. The log contains information such as user name (Submitter name), user ID, date and timestamp, a link to the CoR, as well the unique confirmation number of a successful submission.  A user can change the email address associated with their user account, if desired. If the email address is changed, an email is sent to the old email address notifying that email account that the change was made. |
| **Supporting Documentation (list attachments):**  N/A |
| **CROMERR System Checklist** | |
| **Signature Validation (e-signature cases only)** | |
| **13. Credential validation (See 13a through 13c)** | |
| **13a. Determination that credential is authentic** | |
|  | **Business Practices:**  N/A |
| **System Functions:**  **(Insert System Name)** System will compare the encrypted form of the user-supplied password and the encrypted form of the challenge question answer provided during the signing process to the encrypted form of the user's password and the encrypted form of the user's response to the challenge question stored in the database.  **(Insert System Name)** System will randomly choose, using a random number generator, one of the five (5) active challenge questions answered in the database for the user's account. The user must enter the answer to the presented question to sign the report. If a challenge question is incorrectly answered, **(Insert System Name)** will automatically cycle to the next challenge question. The system will provide an automatic lockout mechanism after 5 failed electronic signature (challenge question + password) attempts within the current session.  Correct and incorrect challenge question response attempts (user, date- time, success or fail indicator) are logged to the audit table as well as the text-based log file discussed in Item 3. |
| **Supporting Documentation (list attachments):**  N/A |
| **13b. Determination of credential ownership** | |
|  | **Business Practices:**  N/A |
| **System Functions:**  **(Insert System Name)** System will compare the encrypted form of the user-supplied password and the encrypted form of the challenge question answer provided during the signing process to the encrypted form of the user's password and the encrypted form of the user's response to the challenge question stored in the database.  There is no provided system mechanism that could be used to determine what the user's password is after it set by the user, nor the challenge questions selected, or answers to the selected challenge questions. |
| **Supporting Documentation (list attachments):**  N/A |
| **13c. Determination that credential is not compromised** | |
|  | **Business Practices:**  The **(Insert Local/Agency Acronym)** System Administrator access and utilize/share the event log to aid the **(Insert Local/Agency Acronym)** in determining if an account has been compromised, or not. If it is determined that an account has been compromised, the account will be locked, preventing the user from logging into **(Insert System Name)** System, and signing electronic reports, and the user will be contacted to address the situation. In addition, since email acknowledgements are sent after an account is created, a password reset, a password is changed and following the form submission process, the users themselves will be in the best position to determine if their account has been compromised, either through the lack of email notifications, or email notifications received that were results of actions not initiated by them.  Item 11 and 15 further describe this process. |
|  | **System Functions:**  The **(Insert System Name)** System event log can be used to help detect potential credential compromises. The electronic signature process includes answering a challenge question with an answer that is only known to the Submitter. This provides evidence that the Submitter's password remains within the control of the user who created and defined it.  System administrators can lock a user's account using the **(Insert System Name)** System. If a user account is locked, not only can they not login to the system, but they also cannot sign electronic form submissions. The fact that the account was not locked at the time a form was signed provides evidence that neither the user nor administrators believed the credential was compromised at that time. |
| **Supporting Documentation (list attachments):**  N/A |
| **14. Signatory authorization** | |
|  | **Business Practices:**  Reference Item 2 for the process that the **(Insert Local/Agency Acronym)** uses to grant signatory authority to **(Insert System Name)** Portal users and Submitters. |
| **System Functions:**  The **(Insert System Name)** System includes an Electronic Signature role that grants permission for a user to sign and submit forms which require an electronic signature. This role provides signatory authorization. Using the **(Insert System Name)** application, the Electronic Signature role is assigned to a user that he/she has signatory authority. This role can be revoked by the **(Insert Local/Agency Acronym)** at any time or at the request of an organization responsible official.  In addition, in order to even grant the Electronic Signature role to a user, the **(Insert Local/Agency Acronym)** must also have indicated and acknowledged that the Subscriber Agreement has been received and verified by setting the Subscriber Agreement flag in the application for the user, as mentioned in Item 1 b-alt. |
| **Supporting Documentation (list attachments):**  N/A |
| **15. Procedures to flag spurious credential use** | |
|  | **Business Practices:**  Spurious credential use, identified through such events as duplicate forms, off-schedule submissions, false or incorrect data, and deviations from normal content or procedure, would be detected through normal **(Insert Local/Agency Acronym)** quality assurance checks/processes performed by the **(Insert Local/Agency Acronym)** personnel for each submitted report.  If spurious credential use is detected, **(Insert Local/Agency Acronym)** personnel will investigate, typically involving a telephone call to the responsible official, using the telephone number on record with the **(Insert Local/Agency Acronym)**. In addition, the **(Insert Local/Agency Acronym)** may check event logs when there are indications of suspicious activity.  In the event the account has been compromised, the **(Insert Local/Agency Acronym)** will use the event log to investigate all activities performed by that user, including submissions, if applicable, during the timeframe that the **(Insert Local/Agency Acronym)** suspects the account credentials may have been compromised.  If the **(Insert Local/Agency Acronym)** suspects that an authorized user has maliciously submitted false data, then they will call other facility users, including the responsible official, using the telephone number(s) on record with the **(Insert Local/Agency Acronym)** to determine if the user is no longer working at the organization and should have their authorization revoked. The **(Insert Local/Agency Acronym)** will use the event log, and repudiate any reports that were submitted with known compromised credentials. |
| **System Functions:**  The **(Insert Local/Agency Acronym)** will use the User Management functionality of the system to immediately lock out an account, and also revoke signing credentials, once a compromised account has been detected or reported by a user. The **(Insert Local/Agency Acronym)** will rescind any reports that were submitted with compromised credentials using the repudiation process described in Item  10.  Also reference Item 13c. |
| **Supporting Documentation (list attachments):**  N/A - Reference Item 13c. |
| **16. Procedures to revoke/reject compromised credentials** | |
|  | **Business Practices:**  When a request, by email or telephone, is received by the **(Insert Local/Agency Acronym)** for account locks, password resets, or challenge question rests, the **(Insert Local/Agency Acronym)** will collect the person's information making the request and telephone the responsible official, using the telephone number on record at the **(Insert Local/Agency Acronym)**, to determine the validity of the request. |
| **System Functions:**  The **(Insert System Name)** Administrator can use the User Management functionality of the system to immediately lock out an account and also revoke signing credentials. The User Management functionality can also be used to remove the lock from an account.  The **(Insert System Name)** application will send, via automated email, a notification to the user with any change of their account status (e.g., from locked back to active). In addition, a user whose account is unlocked will be required to reset their password, as described in item 3. |
| **Supporting Documentation (list attachments):**  N/A |
| **17. Confirmation of signature binding to document content** | |
|  | **Business Practices:**  N/A |
| **System Functions:**  Reference item 5 for a full description of binding electronic signatures to the CoR.  Any user who is considered a contributor to a submission, as well as Agency staff, will have the ability to confirm the bound signature and verify the document integrity by clicking the Verify Authenticity feature on the Submission View page. This function will recalculate the signature of the CoR and compare it to the signature generated at the time of submission. If any part of the CoR was altered or the electronic signature improperly bound, the new signature would differ from the original, allowing the system to detect the change and the invalid signature binding. |
| **Supporting Documentation (list attachments):**  N/A |
| **Copy of Record** | |
| **18. Creation of copy of record (See 18a through 18e)** | |
| **18a. True and correct copy of document received** | |
|  | **Business Practices:**  N/A |
| **System Functions:**  Reference Item 9 for a description of the CoR content and the process used to assure it is a true and correct copy of reported data. |
| **Supporting Documentation (list attachments):**  N/A |
| **18b. Inclusion of electronic signatures** | |
|  | **Business Practices:**  N/A |
| **System Functions:**  Reference Item 9 for a description of the CoR content and information on how the electronic signature is included in the CoR. |
| **Supporting Documentation (list attachments):**  N/A |
| **18c. Inclusion of date and time of receipt** | |
|  | **Business Practices:**  N/A |
| **System Functions:**  **(Insert System Name)** includes the date and time of the submission in the CoR. Reference Item 9 for a description of the CoR content. |
| **Supporting Documentation (list attachments):**  N/A |
| **18d. Inclusion of other information necessary to record meaning of document** | |
|  | **Business Practices:**  N/A |
| **System Functions:**  Reference Item 9 for a description of the CoR content. |
| **Supporting Documentation (list attachments):**  N/A |
| **18e. Ability to be viewed in human-readable format** | |
|  | **Business Practices:**  N/A |
| **System Functions:**  Reference Item 9 for a description of how the CoR is provided in a human-readable format. |
| **Supporting Documentation (list attachments):**  N/A |
| **19. Timely availability of copy of record as needed** | |
|  | **Business Practices:**  N/A |
| **System Functions:**  **(Insert System Name)** generates the CoR immediately following the form submission process. Following the completion of the signing, PDFing and packaging process, the CoR is available for review by the Submitter. The **(Insert Local/Agency Acronym)** Agency users with access to the **(Insert System Name)** System are also able to view a form submission CoR.  Users can view the CoR online through the **(Insert System Name)** application, can print the CoR, or can download the CoR to their computer/network. The CoR is available and viewable via **(Insert System Name)** per the retention period specified by the **(Insert Local/Agency Acronym)**, which currently is indefinitely. Reference Item 20 for retention period details.  The CoR can be obtained through various lookup methods. An **(Insert System Name)** Agency user would use one of the submission search methods available to select a particular form submission where the CoR is being sought. The Submission Details and Submission View pages are used to obtain the submission details, which include date/time of submission, the Submitter's name, the submission status, the reported data, related attachments, and provides the ability to download/view the CoR for the particular form submission.  Refer to Attachment 11: View Submission History. |
| **Supporting Documentation (list attachments):**  - Attachment 11: View Submission History |
| **20. Maintenance of copy of record** | |
|  | **Business Practices:**  N/A |
| **System Functions:**  **(Insert System Name)** CoR are stored and retained in the **(Insert System Name)** database on a database server in the **(Insert Local/Agency Acronym)** environment. Form submissions are stored in the database as Binary Large Objects (BLOB). A BLOB has no structure that can be interpreted by the database management system, but is known only by its size and location. **(Insert System Name)** assigns each CoR a unique document identifier and is related to the Submission.  Using the **(Insert System Name)** system, there is no modification to the CoR possible through any existing functionality (only selecUview and insert capability). Nor does the application's database user account have the ability to modify or delete a CoR.  The **(Insert Local/Agency Acronym)** Database Administrator of the database would have the ability to delete a CoR due to the nature of these account types and their privileges within the DBMS (e.g., a database administrator could remove their access to the table containing the CoR, but cannot remove their ability to grant themselves access again; the very nature of the database administrator account prevents this).  In addition, the bound signature can be confirmed and the document integrity verified by recalculating the signature of the CoR and comparing it to the signature generated at the time of submission, as described in Item 17. If any part of the CoR was altered, including the signature binding information, the new signature would differ from the original, allowing the system to detect the change.  The CoR can be searched, viewed, and downloaded as specified in Item 19.  The **(Insert Local/Agency Acronym)** will backup both database data and log files on a nightly basis. The **(Insert Local/Agency Acronym)** backups will be replicated and stored in an off-site, secure location.  The application and database server will be behind a firewall in a DMZ. Security is in place to only allow access via HTTP, HTTPS and FTP protocols. The firewall is monitored daily. The system also has virus protection installed on the server.  System and Network Administrators are the only staff that have permissions to access the server both physically and remotely through network credentials set with authorized approval. The computer room is locked at all times. Only authorized personnel have access to this room. **(Revise or insert any additional processes as needed)**  At the minimum, **(Insert Local/Agency Acronym)** will store the CoR per Federal or State regulations, whichever is longer. **(Insert Local/Agency Acronym)** will utilize 5 years for their minimum record retention time period. |
|  | **Supporting Documentation (list attachments):**  N/A |