IDEM CROMERR Checklist Submittal (September 2008)

40 CFR, Part 3 – *CROMERR* Checklist

Item 1: Identity-Proofing Of Registrant

Business Practices

With regard to identity-proofing of a registrant, eAuth will use a Subscriber Agreement. Per Cross-Media Electronic Reporting Requirements (CROMERR) 3.2000(b)(5)(vii)(C), the receipt of a signed Subscriber Agreement is sufficient proof of the user's identity. See Chapter 6 for more information on the Indiana Department of Environmental Management Electronic Subscriber Agreement (IESA). Also, see Appendix 2 for a sample of the IESA.

A user of an eAuth Enabled Application (eA-EApp) is required to complete an on-line registration process specific to that particular eA-EApp. As part of the registration process a registrant record is created in the eAuth Identity (eA-Identity) system in which information supplied by the user during the registration process is stored. Additionally, a number of other pieces of information specific to the eAuth Application Administrator (eA-AppAdmin) follow-on activities are created. Most of the follow-on activities are presented to an eA-AppAdmin as a series of on-line checklists which must be completed in their entirety before the eA-AppAdmin can grant user access to any given eA-EApp and grant signatory ability (if appropriate).

During the user registration process, the eAuth system provides the user with a downloadable/printable IESA. The appropriate eAuth Application Administrator (eA-AppAdmin) must receive via standard mail an eA-EApp -specific IESA prior to user account or role activation.

Every IESA requires a company's authorizing official to acknowledge that s/he or the company has performed the following duties noted in NIST 800-63 for assuring that the applicant is in possession of a valid Government ID:

1. The authorizing official or company has validated that the prospective user is in possession of a valid current primary Government Picture ID that contains the applicant's picture, and either address of record or nationality (e.g. driver s license or passport).
2. The authorizing official or company has inspected said ID to compare picture to applicant, have recorded the ID number and issuer of the ID as well as the address and Date of Birth, as so indicated on the Government Picture ID.
3. The authorizing official or company will keep a record of the above information for a minimum of five years after user employment termination or change in position.

An IESA submitted to an eA-AppAdmin must meet the following conditions:

1. No alterations that materially change the meaning and/or substance of the document have been accomplished beyond those necessary for the registrant, the company's authorizing official and/or other company personnel to complete the document.
2. Completed in its entirety (excluding sections reserved for official IDEM use).
3. Contain submitting user's handwritten signature.
4. Contain the company's authorizing official's handwritten signature.

The eA-AppAdmin will contact the company's authorizing official and/or employer by telephone, per standard eAuth user identity-proofing procedures, review/validate the information provided on the submitted IESA and perform identity proofing.

The eA-AppAdmin, at a minimum, validates/confirms:

1. IESA completeness
2. IESA contains applicant's handwritten signature
3. IESA contains company's authorizing official's handwritten signature
4. Applicant's business employment
5. Applicant's authorization level
6. Applicant's ability to serve as a signatory, and, if so, if authority has been delegated

eA-Identity provides the vehicle by which consistent identity-proofing procedures are performed and documented. The eA-AppAdmin will store the received and verified IESA in a paper-based filing system until such time as the received and verified IESA is scanned and the scanned image is stored in the IDEM's Virtual File Cabinet (VFC) document management system. IDEM currently retains IESA's for all signing credentials for a minimum of five years after account deactivation.

System Functions

During the registration process for each distinct eA-EApp, the eAuth system presents each registrant with:

1. a web-based link to:
    a. download and/or print the IESA
    b. instructions for its completion
    c. follow-on actions the eA-AppAdmin will undertake upon receipt of the IESA
2. an opportunity to review all of the above on-line.

The registrant signifies his/her understanding of these instructions/processing actions by clicking an on-screen "I Agree" button.

Identity-proofing steps performed by an eA-AppAdmin are documented via checklists and additional data structures created and maintained within the eA-Identity system.

Supporting Documentation (list attachments)

Attachment 1 - eAuth.doc

Item 1a: (priority reports only) Identity-proofing *before* accepting e-signatures

Business Practices

Before eAuth registrants are issued security/signatory credentials, s/he must complete the on-line registration process (discussed in Item 1, above), complete an IESA and have his/her identity verified through rigorous identity-proofing procedures conducted by an eAuth Application Administrator (eA-AppAdmin).

Under no circumstances is an eA-AppAdmin to grant signatory authority to any registrant for an eA-EApp without:

1. **receiving an IESA completed by the registrant for that specific eA-EApp and**
2. **the IESA for that specific eA-EApp verified by an eA-AppAdmin through telephone interviews with company officials.**

See Chapter 6 for a detailed discussion of the IESA, including:

1. contents
2. how the user will provide information
3. adequacy for proof of identity
4. role in the identity-proofing process
5. registrant's authorizations
6. registrant's signatory authority
7. storage
8. retention schedule

System Functions

eAuth will not allow a user's electronic signature device to sign electronic documents until an IESA has been received and verified by the appropriate eA-AppAdmin and the eA-AppAdmin signifies same by the appropriate setting of flags in the eAuth security and eA-EApp databases.

eAuth Identity (eA-Identity) provides the vehicle by which consistent identity-proofing procedures are performed and documented. As part of the identity-proofing measures performed by eA-AppAdmin's, s/he verifies authorization levels and signatory rights of new registrants through telephone interviews with company officials. The registrant's highest allowed authorization level and signatory right for a given facility are explicitly stored in the eA-Identity database.

An eA-AppAdmin's ability to grant eA-EApp authorization level(s) and/or signatory right is subject to the following conditions existing in the eA-Identity database:

1. successful completion of the identity-proofing process (i.e. checklists)
2. appropriate eA-EApp enabling flags set
3. authorization no higher than the level indicated in the database
4. signatory right indicated in the database

Likewise, each user's login attempt triggers the same checks

See Business Practices above and Item 1b-alt for more information on:

1. the information contained in the IESA,
2. how the user will provide the information, and
3. the business processes used by an eA-AppAdmin to verify and assure the requested access is appropriate for the user.

Supporting Documentation (list attachments)

Attachment 1 - eAuth.doc

Item 1b: (priority reports only) Identity-proofing method (See 1bi, 1bii and 1b-alt)

Item 1bi: (priority reports only) Verification by attestation of disinterested individuals

Business Practices

N/A – user 1b-alt Subscriber Agreement alternative

System Functions

N/A – user 1b-alt Subscriber Agreement alternative

Supporting Documentation (list attachments)

N/A – user 1b-alt Subscriber Agreement alternative

Item 1bii: (priority reports only) Information or objects of independent origin

Business Practices

N/A – user 1b-alt Subscriber Agreement alternative

System Functions

N/A – user 1b-alt Subscriber Agreement alternative

Supporting Documentation (list attachments)

N/A – user 1b-alt Subscriber Agreement alternative

Item 1b-alt: (priority reports only) Subscriber Agreement alternative

Business Practices

Per Cross-Media Electronic Reporting Requirements (CROMERR) 3.2000(b)(5)(vii)(C), the receipt of a signed Subscriber Agreement is sufficient proof of the user's identity.

eAuth will use a Subscriber Agreement referred to as the Indiana Department of Environmental Management Subscriber Agreement (IESA). Before eAuth registrants are issued security/signatory credentials, s/he must complete the on-line registration process (discussed in Item 1 above), complete an IESA and have his/her identity verified through rigorous identity-proofing procedures conducted by an eAuth Application Administrator (eA-AppAdmin).

Under no circumstances is an eA-AppAdmin to grant signatory authority to any registrant for an eA-EApp without:

1. **receiving an IESA completed by the registrant for that specific eA-EApp and**
2. **the IESA for that specific eA-EApp verified by an eA-AppAdmin through telephone interviews with company officials.**

See Chapter 6 for a detailed discussion of the IESA, including:

1. contents
2. how the user will provide information
3. adequacy for proof of identity
4. role in the identity-proofing process
5. registrant's authorizations
6. registrant's signatory authority
7. storage
8. retention schedule

System Functions

The applicant will complete portions of the IESA in an online eAuth Enabled Application (eA-EApp) registration form specifically tied to each eA-EApp. All eA-EApp registration forms collect a standard set of information and optionally eA-EApp-specific information.

eAuth automatically determines to which eA-AppAdmin the signed IESA should be mailed. The applicant must print, sign, and mail the agreement to the specified eA-AppAdmin. If the signing authority is being delegated to the applicant, the delegating authority must also sign the IESA.

The applicant's electronic signature device will not be able to sign electronic documents until the IESA has been received by the appropriate eA-AppAdmin and the eA-AppAdmin has verified the information through telephone interviews with company officials (see Business Practices above).

The online eA-EApp registration form requires, at a minimum, the applicant to enter the following data:

1. Full name (First Name, Middle Initial, Last Name)
2. Email address.
3. Mailing address (Street, City, State, Zip)
4. Daytime Telephone
5. Company for which access is desired.
6. Desired eA-EApp authorization level(s).

    Note: Authorizations are not automatically granted, but rather verified during the identity-proofing process by the eA-AppAdmin with the company's authorizing official(s). Authorizations which give the applicant the ability to submit and sign reports are determined by the company's authorizing official and must be explicitly communicated to the eA-AppAdmin via the IESA (see Item 4, below) and confirmed during the follow-on procedures conducted by the eA-AppAdmin.

7. Whether the applicant has direct authority under the rules to sign the report(s) for the company or the authority is being delegated to him/her.
8. If the authority is delegated, the name and title of the person delegating the authority.

The agreement includes language, in the first person, stating that the requestor:

1. Agrees to:

    a. Protect their account password from compromise, not allow anyone else to use the account, and not share the password with any other person.
    b. Promptly report to the eA-AppAdmin(s) any evidence of the loss, theft, or other compromise of the user account password.
    c. Notify eA-AppAdmin(s) if the user ceases to represent any of the requested companies as the submitter for the organization's electronic reports as soon as this change in relationship occurs.
    d. Review, in a timely manner, the acknowledgements (email and onscreen) and copies of submitted documents using their account.
    e. Report any evidence of discrepancy between documents submitted, and those received by the eA-AppAdmin.
    f. By affixing his/her signature, the user explicitly provides he/she will adhere to all eAuth/eA-EApp policies, terms and conditions listed in the agreement.

2. Understands that he/she will be held as legally bound, obligated, and responsible by the use of his/her electronic signature as he/she would be by a handwritten signature and that legal action can be taken against him/her based on his/her use of the electronic signature in submitting electronic documents.

See Item 3 for information on how the user account is created.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 2: Determination of registrant's signing authority

Business Practices

IDEM is responsible for establishing and documenting policies and procedures to:

1. Determine the accuracy of applicant information
2. Verification of applicant information
3. Determine, for the each organization associated with the applicant, the following:
    a. Organization(s) the applicant will be authorized to represent
    b. Appropriate eAuth Enabled Application (eA-EApp) authorizations
    c. Appropriateness of applicant's signatory authority
    d. Granting, if appropriate, applicant signatory authority

The IDEM Electronic Subscriber Agreement (IESA) contains a section wherein a company's authorizing official (at the level defined by the appropriate IDEM Program Office (PO)) attests that the registering applicant is employed by and/or is authorized to represent the company and, if appropriate, the registering applicant is authorized to submit and electronically sign reports on behalf of the company. Additionally, a Sponsor Letter is a required attachment to the IESA. The Sponsor Letter must be on company letterhead and must include, at a minimum, the identical attestations contained in the IESA and the handwritten signature of the same authorizing official signing the IESA.

The applicant must mail the completed IESA and Sponsor Letter to the appropriate eAuth Application Administrator (eA-AppAdmin).

The eA-AppAdmin ensures standardized verification and identity-proofing procedures are followed and properly documented. eA-AppAdmin's, according to policy and in order to provide evidence of identity-proofing to relying parties, must maintain a record of applicants whose identity have been verified and the procedure followed to verify their identities. eAuth Identity (eA-Identity) provides the vehicle by which consistent identity-proofing procedures are performed and documented.

The eA-AppAdmin also ensures the IESA, the Sponsor Letter and verification/identity-proofing procedure documentation (if appropriate) are stored in a paper-based filing system until such time as all documents are scanned and the images stored in IDEM's Virtual File Cabinet (VFC) document management system. The IESA, Sponsor Letter and verification/identity-proofing procedure documentation shall be retained for a period of 5 years after being notified of the applicant's departure from his/her sponsoring organization by a company official.

Each eA-EApp employs a role-based user authorization system. The applicant's role-based access to eA-EApp's and, if appropriate, signatory permissions shall not be activated until verification and identity-proofing requirements are completed and, in accordance with established

role authorization approval processes, access and, if appropriate, signatory authorization is granted by the appropriate eA-AppAdmin

Each PO is responsible for assigning an eA-AppAdmin for each eA-EApp. The PO is responsible for ensuring eA-AppAdmin's are thoroughly versed and satisfactorily complete all appropriate training to execute the details of all established policies and procedures before assuming the duties of an eA-AppAdmin. The IDEM Office of Information Technology assists each PO in carrying out these actions by identifying the appropriate eA-EApp applicant registration and authorization programmatic functions to support the eA-AppAdmin's signatory approval process (e.g. eA-EApp specific administrative interfaces, links to appropriate established policies, procedures, forms, etc.).

See Supporting Documentation for information related to verification, authorization procedures and IESA storage and retention.

System Functions

Registration processes for an eAuth Enabled Application (eA-EApp) are accomplished via an on-line web-based interface and follow-on activities by the eAuth Application Administrator with company officials. The registration process presents the applicant for an eA-EApp all necessary instructions and forms to:

1. register for access to an eA-EApp
2. obtain organizational authority
3. signatory authority, if appropriate, on behalf of a organization
4. mail evidentiary details to the appropriate eAuth Application Administrator (eA-AppAdmin)

eA-EApp reporting capabilities are tightly bound to user roles in the eAuth system and each distinct eA-EApp. Thus, authorizations which grant applicants the ability to submit and sign reports are strictly controlled by an eAuth Application Administrator (eA-AppAdmin). eA-AppAdmin's have the ability to grant, deny or revoke access to an eA-EApp and, if necessary, signatory authorization. An applicant's electronic signature device is not able to sign electronic documents until granted explicit permission by the appropriate eA-AppAdmin (i.e. mere account creation does not automatically confer any application or signatory authorizations).

See Appendix 2 for additional information.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 3: Issuance (or registration) of a signing credential in a way that protects it from compromise

Business Practices

eAuth supports e-signature credentials. e-signature credentials are devices authorized for creation and use via the real-time input and validation of a user's eAuth Enabled Application (eA-EApp) identity credential (i.e. PIN/password-based approach). To assist protection of the identity credentials, a set of user-provided secret(s) is associated with the user's account profile upon creation of the credential.

Users must adhere and must explicitly acknowledge adherence to strict guidelines when selecting a password for use with an eA-EApp. Upon activation of the credential the user must:

1. Select a password that will not be easily guessed (e.g., names, children's names, birthdays, etc.).
2. Choose a password that is at least eight characters long and contain a mix of numbers and upper/lower-case letters. Compliance with this guideline is automatically enforced by the eAuth system.

Users must adhere and must explicitly acknowledge adherence to strict policies governing access to an eA-EApp, namely:

1. Protect their password by:
    a. not divulging the password to any other individual
    b. not storing it in an unprotected location
    c. not allowing it to be written into computer scripts for automated login purposes.
2. Take appropriate actions if they believe their eA-EApp User account has been compromised
3. Will notify the appropriate eAuth Application Administrator (eA-AppAdmin) within ten working days if their duties change and they no longer need to interact with the eA-EApp on behalf of their organization.

See Item 1b-alt for the business processes used to process received IDEM eAuth Subscriber Agreements (IESA).

See Item 4 for information pertaining to IESA's, their use/verification and account activation.

See Supporting Documentation for additional information related to user registration and this item.

System Functions

eAuth and eAuth Enabled Applications (eA-EApp) provides multiple mechanisms to securely issue signing credentials and use of them for on-line electronic signature processes. Namely:

User access and information exchanged with the eAuth system and all eA-EApp's is performed over Secure Socket Layer (SSL) connections between the user's web browser and the eAuth Web/Application/Database Servers. Thus, third parties are prevented from deciphering and/or viewing any sensitive information exchanged with eAuth and eA-EApp's during a user's active web browser session. Negotiation of the version of SSL used for secure sessions is controlled through server configuration files.

Users specify their selection of an eAuth/eA-EApp User ID and Password as part of each eA-EApp user registration process. The eAuth/eA-EApp User ID is automatically entered on the IESA submitted by the user. The User password must be at least 8 characters long and contain a mixture of numbers and upper/lower-case letters. Upon entry, the user's selected ID is stored in the eA-EApp registration/security database, and the password is stored in a protected manner, namely by:

1. Applying a one-way hash (SHA-256) to the password
2. Storing the resulting Hex value in the registration/security database
3. Storing a creation date timestamp in the registration/security database

Upon subsequent logins, user authentication is accomplished through a comparison of the one-way hash value of the session-specific user-supplied password with the hash value of the most recently established password.

The eAuth systems expire passwords every 90 days. Users are sent an out-of-band email notifying them of this occurrence along with instructions on how to reset their password. eAuth retains all previously entered passwords for the user in order to prevent password re-use.

Users are allowed to alter their account profiles; however, the system requires the user answer a secret question (established at registration time, see below) prior to gaining access to their account profile. This helps ensure that the user established the original account in question and was properly vetted. All user changes to an account profile for a facility immediately disable access to that facility. The eA-AppAdmin is notified via system function(s) of the change. The account owner is also notified, via email, of all account profile changes. Thus the original owner of the registered email address will continue to receive account profile modification notifications, even in the event the account is compromised. The account owner must undergo another round of identity-proofing before the eA-AppAdmin will reactivate the account.

A change to the user's registered email is handled as a special case. The new email address is not used for notifications (i.e. the vetted email continues to be the email address utilized for user notifications), until the eA-AppAdmin completes the new identity-proofing and re-enables the user's account. At that time, the new email address becomes the email of record.

In addition to the business and system processes outlined in Items 1b-alt and Item 4 pertaining to IESA's, IESA usage/verification and account activation, the eAuth/eA-EApp system displays a list of twenty questions at registration time, from which registering applicants are required to select five questions and provide a secret answer for each selected question (referred to as 20-5-1 security question/answer technique). Each of the answers is independently secured as follows:

1. The system retrieves the user's ID
2. The system concatenates the user ID, question # and user-supplied answer
3. The concatenated value is SHA-256 hashed
4. The Hex value of the resulting hash is stored in the registration/security database
5. The creation date timestamp is stored in the registration/security database

Use of the eAuth/eA-EApp User ID in the hash computation ensures that the supplied answers are tied to a particular eAuth/eA-EApp account.

To apply electronic signatures to submission documents, eAuth/eA-EApp's use a combination of the account holder's eAuth/eA-EApp User ID, hashed Password and, from the set of five user-selected questions, a randomly selected question and its associated user-specified hashed secret answer, as well as session-specific information. Use of a randomly selected user-specified secret is known as the 20-5-1 security question/answer technique.

Further use of User ID/Password information or the 20-5-1 secret(s) in the e-signature process are described in Item 5.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 4: Electronic Signature Agreement

Business Practices

See Item 1b-alt

See Chapter 6 and Appendix 2 for more information, a detailed description and sample of the IDEM Electronic Subscriber Agreement (IESA).

System Functions

The eAuth web-based registration system provides user's the opportunity to obtain and print the IESA either during the new user registration process or during existing user account profile maintenance. Subsequent processing of the IESA is handled by business practices outlined in Item 1b-alt.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 5: Binding of signatures to document content

Business Practices

eAuth/eA-EApp's support use of eAuth/eA-EApp-specific account information and user-proprietary secrets (known as the 20-5-1 method) to authorize the creation/use of an electronic signing credential. Creation of the electronic signing credential is accomplished at document submission time.

System Functions

Document submission is accomplished in multiple steps in a straightforward, linear fashion. See Chapter 7 for a detailed discussion of the submission process. eAuth/eA-EApp systems do not, by design, support multiple document submission in one submission session, thus the systems create a unique Copy of Record COR) for each data submission.

A few pertinent definitions are listed below:

Data Document

The Data Document (DD) is an XML document containing user's client web-browser content at the time of submission, thus ensuring the user has the opportunity to view the document being submitted in human readable format. One portion of the XML document contains an exact representation of the user's browser content at the time of signing. Thus, the user or appropriate system function is able to retrieve and display the signed document in human-readable format. At a minimum, a data document contains those items listed in Item 6 – System Functions. It should be noted that the user's web-browser content always include appropriate user attestations/certifications pertaining to the user's acceptance, understanding and explicit acknowledgement of their responsibilities and legal implications associated with use of their electronic signature credential.

Submission Receipt

A Submission Receipt (SR) is created for each submission at the time of the submission. The SR is an XML document containing additional data/metadata related to the data document and is tightly-coupled to the COR submission. The SR includes:

1. Document Id
2. submission source; (i.e. the specific eAuth/eA-EApp system which mediated the submission)
3. submission type (e.g. eDMR)
4. submission document type (e.g. XML, MS-Word, etc.)
5. submission date and time
6. submitter account login id (username)
7. submitter full name
8. submitter email address
9. submitter operating system and version of submitting computer
10. submitter browser and version of submitting computer
11. IP of submitting computer
12. certificate public key
13. SHA-256 hash of the submission
14. signed Data Document (DD) with embedded signature credential (i.e. user's encrypted password)
15. placeholder for image type (empty)
16. placeholder for other information necessary to establish pertinent COR data associated with content stored in the BLOB fields (empty), such as:
    a. permit #
    b. outfall #
    c. facility name

Copy of Record

The Copy of Record (COR) contains:

1. Submission Receipt (SR), primary COR
2. An XSL style sheet to apply to the DD and/or SR content (if either varies from standard well-know XSL styles)

Standard submission documents include the COR and a standalone signed DD

Note: eA-Sign utilizes an application utility, developed collaboratively by IDEM and US EPA to validate electronic signatures and bind them to documents.

See http://test.epacdxnode.net/cromerr/site/download.html for more information concerning this utility.

During the submission process, users are informed of the implications of their review/certification/signing of submission documents as per the mechanisms described in Items 6 and 7. After their acknowledgement of these conditions, the eAuth system downloads a client side control to the user's workstation and prompts the user for their current account password. This password, along with the current known User ID from the eAuth session management table is then hashed as per the procedure noted in Item 3 and compared to the current User ID/Password combination. If this combination is valid, it is immediately used to authorize access to a randomly selected question/answer pair from the list of five selected 20-5-1 questions. This re-establishment of the password ensures that the user has not walked away from their workstation while the submission action is in progress, thereby allowing others to select submission files or perform other actions while the account owner is not present.

When a valid User ID/Password combination is provided, the eAuth system will randomly select one of the five questions selected by the user during the 20-5-1 registration process for the

application and request that the user provide the correct response to that question. The current user-supplied answer is then hashed as per the procedure noted in Item 3 and then compared with the answer as originally recorded.

If the user-supplied answer to the 20-5-1 challenge is correct, eAuth uses the client side control to create a 1024-bit public/private key pair using the properly hashed User ID and Password hash. The public key from this process is stored in a temporary X.509 signing certificate on the user workstation that also includes current user/session information. This temporary X.509 certificate is signed by an US EPA Central Data Exchange (CDX) server process call using a CDX server private certificate.

A message digest for each submission document is created on the client by the client side control using an SHA-1 algorithm. The document is subsequently signed utilizing an RSA key pair (public key/user's temporary private key). An SHA-256 hash of the signed document is then calculated. The temporary X.509 certificate, signed document, signature (encrypted document message digest) AND the resulting SHA-256 value are submitted to the eAuth/eA-EApp server. The eAuth system, prior to insertion into the appropriate Copy of Record (COR) data store, calculates an SHA-256 hash of the received signed document and compares the resulting value to the SHA-256 hash received from the client. Matching SHA-256 hash values essentially ensures the signed document arrived intact. Non-matching SHA-256 hash values are treated as submission failures. Successful transmissions are inserted into the COR database with a unique document ID.

Storage of the SHA-256 hash in the COR repositories facilitates quick, on-demand verification (via recalculation of the SHA-256 hash) that the COR has not been altered subsequent to initial insertion.

See Chapter 7, Appendix 1 and Appendix 2 for a detailed description of the eA-Sign system

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 6: Opportunity to review document content

Business Practices

None

System Functions

eAuth/eA-EApp systems present submitters with a verification/confirmation page or dialog consisting of one or more read-only pages (except for confirmation/affirmation checkboxes which require the submitters to check). All certifications and affirmations will be in the first person and are included in user's submission. Minimum content is indicated below:

1. Certification/warning statement outlining the proper use of their signing credential.
2. Certification/warning statement (in the first person) outlining the legal implications of attaching their electronic credential device to submission materials.

3. Certification statement that the submitter is the owner of the account he/she is using and s/he has protected the account and password and is in compliance with the IDEM eAuth Subscriber Agreement (IESA).
4. Affirmation that the signatory is not aware of any compromise to his/her signature credential.
5. Agrees that providing the account password to sign the document constitutes an electronic signature equivalent to his/her written signature.
6. Certification the submitter has the authority to submit the data on behalf of the represented facility.
7. Certification statement that the submitter understands this attestation of fact pertains to the implementation, oversight, and enforcement of a federal environmental program and must be true to the best of the submitter's knowledge.
8. A checkbox indicating the submitter has read, understood and acknowledged the certification statements and non-compromise affirmation. The submitter must place a check in the checkbox before the submission can be made (i.e. activate SUBMIT button).
9. A checkbox indicating the submitter has been given an opportunity to review all pertinent data associated with the submission. The submitter must place a check in the checkbox before the submission can be made (i.e. activate SUBMIT button).
10. The data being submitted and/or a button control, the selection of which (though not required), provides the submitter the opportunity review detailed data (read-only) associated with the submission.
11. Links and/or other controls to allowing the submitter to view and download any additional information associated with the submission.
12. A textbox for the eA-EApp system utilized for the submission (pre-populated).
13. A textbox for the submission date and time.
14. A textbox for the user id of the submitter.
15. A textbox for the name associated with the submitting user id.
16. A text box for the user's one-way hashed password (credential)
17. A textbox for the secret question answered during the submission process.
18. A textbox for the response (one-way hashed) to the secret question.
19. A textbox for the SHA-256 hash of the submission.

Submitter's acknowledgements/affirmations as indicated by the placement of checks in all above-mentioned checkboxes will activate an on-screen button which when clicked initiates the signing and upload process. It should be noted that the user's web-browser content always include appropriate user attestations/certifications pertaining to the user's acceptance, understanding and explicit acknowledgement of their responsibilities and legal implications associated with use of their electronic signature credential. See Attachment 20 for examples with application certification statements. The XML representation of the user's client web-browser content at the time of signature process initiation is the document the user signs and thus ensures the user sees the document being submitted.

Since the submitter's acknowledgment of the certification/warning statements and various affirmations are part and parcel of the submission, there is no need to separately capture them in audit tables.

During the signing process and prior to submission the textboxes for submission date/time, user id, user name, and secret question answered are transparently populated. The resultant XML document is SHA-256 hashed and the hash stored in the appropriate textbox. The XML document is then transparently uploaded.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc
Attachment 20 – Examples.doc

~~Attachment 18 – Example – DMR Submission Browser Session~~
~~Attachment 19 – Example - DMR Copy of Record~~.

Item 7: Opportunity to review certification statements and warnings

Business Practices

The verification/confirmation pages and/or dialogs used by individual eAuth Enabled Applications (eA-EApp) minimally include content outlined in Item 6; however, the specific text displayed by the system in the signature certification and warning statement(s) used by an application is specified by individual IDEM Program Offices (PO).

System Functions

Verification/confirmation pages or dialogs are presented to a user prior to affixing of electronic signature credentials, system acceptance of signed submissions. See Item 6 for the content that will, at a minimum, appear on verification/confirmation pages or dialogs and example verification/certification statements.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 8: Transmission error checking and documentation

Business Practices

None

System Functions

eAuth/eA-EApp systems use only SSL-secured HTTP sessions (HTTPS). eAuth/eA-EApp systems support SSL v3.0 128 bits (IOT Standard) and TLS v1.0 256 bits, thus preventing man-in-the-middle attacks. These protocols ensure encrypted application messages are exchanged between the client and server. Every exchanged data record must be successfully decrypted on the server using the negotiated key to maintain a viable connection status. If data is found to be corrupted during transmission (i.e. the server decryption fails) the protocol automatically retransmits.

To determine whether submission documents are faithfully received by eAuth/eA-EApp systems, an SHA-256 hash of submission documents is computed on the client machine (post signature-binding) and concurrently submitted with submission documents. Prior to insertion into the appropriate Copy of Record data stores, the system re-computes the SHA-256 of the submission and compares it to the SHA-256 hash submitted from the client side. Identical values essentially ensure the submission was received intact. Non-matching values causes the system to fail the submission. In either case, both the user and appropriate eA-AppAdmin are notified via email of the status of the submission. Additionally, the user is notified in real-time by an appropriate dialog displayed at the end of the submission session.

Each submission and its success/failure is tracked in system audit logs. Logs contain, at a minimum:

1. eA-EApp system
2. Submission document type (i.e. report)
3. Submitting User Id
4. Timestamp
5. Submission outcome
6. Doc ID (if successful)

The integrity of submission files and/or data is additionally protected as follows:

1. Submitted documents are presented in XML format on the user's client browser at the time of signing.
2. The XML representation of the user's client web-browser content at the time of submission is the document the user signs, thus ensuring the user sees the document being submitted,
3. The information in the data XML document used for the verification page (see Item 5) comes from data already stored in the eAuth/eA-EApp databases. No updates to this data are performed at any time during or after the submission process. With the protection in place from man-in-the-middle attacks, this provides a high level of assurance that the user is seeing the data as it is stored in the database.
4. No alteration of the document content is made during transmission or after it is received.
5. The data XML document is included, without alteration, in the COR. This assures that the COR contains the same data, in the same format, the user was given the opportunity to review (see Item 6).
6. For successful transmissions the COR signature (see Item 5) is provided to the user in an email acknowledgement along with instructions to access the COR. The email allows the user to detect modifications to the submission. See Item 5 for more information.
7. It is computationally infeasible for the user to create a valid COR signature without the eAuth generated one-off public and private keys. This protects against users modifying the COR and attempting to claim the data were altered in eAuth
8. The validity of the signed COR can be determined using the eAuth 'one-off' public key. This assures that the eAuth 'one-off' private key was used to sign the COR.
9. The post submission data hash can be recomputed, if needed, to compare against the original values and thus determine whether stored binary submission objects have been altered.
10. The submitter has the opportunity to review the data during data entry, the submission process, and the COR review process.

See Item 5 for the submission process and more detail on how the submission process protects against alterations once it has been received by eAuth and the appropriate eA-EApp.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 9: Opportunity to review copy of record

Item 9a: Notification that copy of record is available

Business Practices

See Item 5 and 6 for Copy of Record (COR) content.

An XSL style sheet (to apply against XML submission file(s) or metadata documents) may be included if XML-style documents are included as part of, or generated during, the submission process.

Certificate public keys, if necessary for review, are also included.

Users are notified of the availability or submission failure of their eAuth/eA-EApp Copy of Record (COR) through the System Functions described below. Notifications always contain at least a standard set of information; however, additional content can be specified for inclusion by each IDEM Program Office (PO).

System Functions

eAuth/eA-EApp systems inform submitters of COR availability in the following ways:

1. Automatic out-of-band notification to the user's registered email address after each submission with:
    a. the success or failure of the submission
    b. if successful, instructions on how to access the COR
    c. eA-EApp specific information

2. The status of COR submission is immediately available from the appropriate eA-EApp. Each eA-EApp has the ability to:
    a. determine status of COR submissions
    b. search for successfully submitted COR's
    c. list all submitted COR's.

The comprehensive details related to each eA-EApp's notification process/content and COR viewing/search/download capabilities are eA-EApp specific.

For information on how a user would view the COR see Item 9c.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 9b: Creation of Copy of Record in Human-Readable Format

Business Practices

eAuth/eA-EApp systems receive submission files as XML formatted documents.

System Functions

COR submissions are XML documents and include any necessary XSL style sheets to be applied to the XML documents. The XML tags used in the documents inherently provide data context for document data. One portion of the XML document contains an exact copy of the user's browser content (i.e. human-readable plain text source used to display document content) at the time of signing. Source XML and HTML tags inherently provide data context. Thus, the user or appropriate system function is able to retrieve and display (re-render in a web browser or other compatible viewer) the signed document at any time, and viewed exactly as it was presented to the user at the time of signing – again, obviously, a human-readable format.

After a user has acknowledged all necessary certifications/affirmations and explicitly elected to sign and submit, the COR is created and stored/retained in the eAuth/eA-EApp archival databases in their native format(s). Submitted documents are additionally assigned a document id.

eAuth/eA-EApp systems have provisions for converting the COR into static image formats and storing the resultant image side-by-side within the same database record as the original submission.

Additionally, images are system-compatible with the searchable IDEM Virtual File Cabinet (VFC) document management system. The images are submitted to/stored in the VFC and appropriately indexed/cross-referenced sufficiently to be easily identified and/or located.

See Items 5 and 6 for information on what comprises a COR and the COR creation and submission processes.

See Item 9c for how users can search for, view and/or download the COR.

See Item 20 for more information related to the VFC.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 9c: Providing a copy of record

Business Practices

None

System Functions

All eAuth/eA-EApp systems provide functionality to view any given Copy of Record (COR) in a variety of fashions:

1. The COR can be viewed on-line (read-only) via the user's web-browser in the same format and content presented during the submitter's submission activity.
2. An image of the COR can be displayed if the user's client workstation has the appropriate image viewer to display .pdf or .tiff formats. The format of the stored image is determinable at run-time.

An image of the COR can also be viewed in a web-browser via the IDEM Virtual File Cabinet (VFC) document management system.

All of the above systems also provide functionality to search for and/or download the COR in their associated storage formats.

Submitters receive email notifications regarding their submissions with sufficient information to easily locate COR's in the appropriate eA-EApp or VFC.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 10: Procedures to address submitter/signatory repudiation of a copy of record

Business Practices

Common reasons a signatory may want to repudiate a Copy of Record (COR) and appropriate measure to be undertake follow. In no case is a submission ever deleted (i.e. remain in COR datastores).

An authorized signatory did not submit the COR

If an authorized signatory disputes a COR submission, s/he must contact the appropriate eAuth Application Administrator (eA-AppAdmin). The eA-AppAdmin will obtain the submission data and associated metadata, the date/time of submission, the submitting User ID, any appropriate audit logs, the public key, and the signature hash that were stored with the submission (the last two items as applicable). That information, along with items from any IDEM Electronic Signature Agreement (IESA) and Sponsor Letters would be used to establish the identity and authority of the submitter with respect to a particular submission.

If it is found that the authorized signatory did not submit the COR, the user's signature credential has been compromised. The user's account will be immediately locked and the disputed COR flagged as repudiated. See below for appropriate further actions the eA-AppAdmin and user must undertake.

An authorized signatory claims his/her signing credential is/was compromised or used inappropriately OR a determination that a authorized signatory's signature device has been compromised

If an authorized signatory claims his/her signing credential is or has been compromised or an eA-AppAdmin has verified compromise, the eA-AppAdmin will immediately lock the user's account to prevent further potential compromise. The eA-AppAdmin will undertake actions similar to those outlined above to determine the extent of the compromise and whether any submissions need to be repudiated. All compromised COR's will be flagged as repudiated. The user and the eA-AppAdmin will also investigate how the account may have become compromised in order to prevent future occurrences.

A submission is erroneous and/or was submitted erroneously (accidentally)

In general, authorized signatories are allowed to resubmit COR's (see Item 11 below), but strongly discouraged from repudiating erroneous or accidental submissions. If however, the signatory insists upon repudiation for the COR in question, the COR and signatory's account will be treated in a similar fashion to the procedures outlined above for unauthorized submissions and/or comprised signing credentials.

System Functions

eAuth, by design, has no on-line facility to repudiate submissions. If a user wishes to repudiate a COR, s/he must contact the appropriate eA-Admin by telephone. eA-AppAdmin telephone numbers are available via appropriate Help and/or Contact Us links in each eA-EApp as well as submission notifications. User's rights to repudiate a submission are first verified by the eA-AppAdmin and secondly challenged with one of the user's security questions to ascertain identity. The eA-AppAdmin also logs notes and all actions taken with regard to the repudiation request in appropriate system logs. If rights and identity are verified, the eA-AppAdmin flags the COR as repudiated and an automatic email is dispatched to the user.

Exact rules for handling repudiated submissions are determined at the IDEM Program Office level; however, at the very least, signatories of record are notified of the erroneous submission via an email sent to the signatory's registered email address.

Verified COR repudiations are managed by appropriate eA-AppAdmin via administrative interfaces in the eAuth and eA-EApp systems and on-line documentation of the repudiation process. eA-AppAdmin's have the ability to lock user accounts as well as revoke and/or re-issue signing credentials.

Authorized signatories are given, via the appropriate eA-EApp systems, the ability to view (read-only), search and download COR's, in part, to identify potentially suspect submissions and initiate repudiation-related communications with appropriate eA-AppAdmin's.

Since all submissions, repudiated or not, are retained in COR data stores, submissions are accessible via appropriate COR search/view functions within individual eA-EApp modules. The status of each COR submission is clearly indicated (i.e. active, repudiated, accidental, etc.).

Note: Retention of all COR submissions provides a running history for any given submission period. This facilitates submission review by administrators and registered users alike through non-public interfaces. Public access to COR's is only available through the public-facing side of IDEM's Virtual File Cabinet (VFC). Copies of CORs are stored in VFC as an image. If the associated COR is repudiated, established protocol dictates, at a minimum, changing the accessibility status of the image via VFC to non-public and thus is no longer available for public-viewing. A replacement, clearly marked as repudiated, is an option that may be exercised according Program Office (PO) specific direction.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 11: Procedures to flag accidental submissions

Business Practices

If a user determines an erroneous or accidental submission was received by the eAuth and/or eAuth Enabled Application (eA-EApp), the user, in general, is allowed to resubmit a corrected COR and /or repudiate the COR. Submissions are never deleted (i.e. remain in COR data stores).

If a document is erroneously submitted or contained faulty data, submitting signatories can correct with a follow-up submission. However, the submitter is obligated to contact the appropriate eAuth Application Administrator (eA-AppAdmin) and have the erroneous submission flagged as such. eAuth, by design, has no on-line facility to indicate that submissions were erroneously submitted or contained faulty data. If a user wishes to have a COR appropriately flagged as erroneous and/or contains faulty data, s/he must contact the appropriate eA-Admin by telephone. eA-AppAdmin telephone numbers are available via appropriate Help and/or Contact Us links in each eA-EApp as well as submission notifications. User's rights to request such actions are first verified by the eA-AppAdmin and secondly challenged with one of the user's security questions to ascertain identity. The eA-AppAdmin also logs notes and all actions taken with regard to the repudiation request in appropriate system logs. If rights and identity are verified, the eA-AppAdmin flags the COR as "accidental" and an automatic email is dispatched to the user.

Exact rules for handling erroneous submissions are determined at the IDEM Program Office level; however, at the very least, signatories of record are notified of the erroneous submission via an email sent to the signatory's registered email address.

Since all submissions, erroneous or not, are retained in COR data stores, submissions are accessible via appropriate COR search/view functions within individual eA-EApp modules. The status of each COR submission is clearly indicated (i.e. active, repudiated, accidental, etc.)

Note: Retention of all COR submissions provides a running history for any given submission period. This facilitates submission review by administrators and registered users alike through non-public interfaces. Public access to COR's is only available through the public-facing side of IDEM's Virtual File Cabinet (VFC). Copies of CORs are stored in VFC as an image. If the associated COR is determined to be erroneous, established protocol dictates, at a minimum, changing the accessibility status of the image via VFC to non-public and thus is no longer available for public-viewing. A replacement, clearly marked as "accidental", is an option that may be exercised according Program Office (PO) specific direction.

If the signatory wishes to repudiate a submission (though s/he is strongly advised not to undertake repudiation for erroneous submissions), the user must contact the appropriate eA-AppAdmin. See Item 10 for the repudiation process and system functions.

System Functions

eAuth and eA-EApp's institute several measures to mitigate occurrences of accidental submissions, namely:

1. Submitters are given the opportunity to review submission content prior to submission (read-only).
2. Submitters must confirm their intent to submit in a multi-step process by providing their user id, password and answer a user-specified security question.
3. Submitters are sent an email after every submission.

4. Submitters can review previous submission COR's (subject to retention rules) via built-in eA-EApp mechanisms

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 12: Automatic acknowledgement of submission

Business Practices

None

System Functions

Upon successful submission/upload of documents, the submitter is presented with a dialog containing:

1. Confirmation of successful binding of the signature device.
2. Confirmation of successful uploading of document(s) to the appropriate eAuth Enabled Application (eA-EApp).
3. A short description of follow-on the system will undertake.

Additionally, an immediate email notification is automatically sent by the eAuth/eA-EApp systems to the submitter's registered email address, the body of which minimally contains:

1. User ID utilized in making the submission
2. Date/Time of the submission
3. SHA-256 signing hash (can be used to compare to COR SHA-256 signing hash)
4. Other information related to the submission (e.g. eA-EApp system, Submission document type (i.e. report), Document ID)

Many eA-EApp systems also send the identical email to a list of authorized users (at the discretion of the company's authorizing official).

All email notifications are appropriately logged in eAuth system components. At a minimum, the following information is logged:

1. eA-EApp system
2. Submission document type
3. User Id
4. Timestamp
5. Document ID
6. Sender email address
7. Recipient email address
8. email body text content

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 13: Credential validation

Item 13a: Determination that credential is authentic

Business Practices

None

System Functions

The eAuth/eA-EApp systems compare the hashed forms of the user-supplied password and answer to the secret question provided during the signing process to the hashed forms of the user's password and the user's response to the secret question stored in the appropriate eA-EApp database(s).

Additionally, for 20-5-1 security question/answer data flows, eAuth/eA-EApp systems determines that the certificate issuer signature contained in the temporary submission-signing X.509 certificate matches the official eAuth/eA-EApp signing certificate. If the issuer information is incorrect, then the submission is rejected and an out-of-band email will be sent to the registered email address for the submitter and the condition is also noted in the eAuth/eA-EApp audit logs.

See Item 3 for how eAuth/eA-EApp systems securely issues and protects PIN/Passwords.

See Item 5 for how eAuth creates the temporary X.509 certificate for PIN/Password Enabled Flows.

See Chapter 7 and Appendix 1 for more information pertaining to the above.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 13b: Determination of credential ownership

Business Practices

None

System Functions

eAuth/eA-EApp systems compare the hashed forms of the user-supplied password and answer to the secret question provided during the signing process to the hashed forms of the user's password and the user's response to the secret question stored in the appropriate eA-EApp database(s).

Additionally, for the 20-5-1 security question/answer data flows, eAuth/eA-EApp validates that the User ID and Password hash information contained within the user identity portion of the temporary x.509 certificate matches the submitter's eAuth/eA-EApp User ID and Password hash

as stored in the appropriate eA-EApp system's Identity Management tables. If the information does not match then the submission is rejected and an out-of-band e-mail will be sent to the registered users email address for that certificate and the condition is also noted in the eAuth/eA-EApp audit logs.

See Item 3 for how eAuth/eA-EApp systems securely issues and protects PIN/Passwords.

See Item 5 for how eAuth creates the temporary X.509 certificate for PIN/Password Enabled Flows.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 13c: Determination that credential is not compromised

Business Practices

eAuth Application Administrators (eA-AppAdmin) review appropriate audit logs to ascertain whether any given user account has been compromised. If it is determined a compromise has occurred for a particular account, the eA-AppAdmin follows established policies and procedures to lock the account and all appropriate follow-on steps (see Items 10, 11, 15, 16 and Supporting Documentation).

System Functions

eAuth/eA-EApp systems implement functionality for eAuth Application Administrators (eA-AppAdmin) to detect credential compromise. Additionally, eA-AppAdmin's are trained in established rules of behavior when dealing with compromised credentials. See Items 10, 11, 15, 16 and Supporting Documentation for descriptions of these functions, policies and procedures.

eA-AppAdmin's have the ability to lock any user's account associated with their respective eA-EApp's. Users also have the ability to lock their own accounts should they suspect their credentials have been compromised. Accounts not locked at the time of submission provide evidence that administrators and users believe a credential is not compromised at the time of submission.

In 20-5-1 security question/answer data flows, the signing instrument is a one-time-use-only X.509 certificate. This temporary certificate is only generated after prompting the user for their account PIN/Password and the Secret Answer to one of their five selected security questions. The hashed values of the current user-supplied PIN/Password and the Secret Answer are then compared to the original hash values stored in the user registration database. If the values do not match, the user is not allowed to generate the temporary X.509 certificate and the failed signing attempt is logged in the eAuth/eA-EApp audit logs.

See Item 3 for specifics regarding user account, user PIN/password and secret answer protection.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 14: Signatory authorization

Business Practices

See Item 2 for processes eAuth Application Administrators (eA-AppAdmin) use to grant signatory authority to eAuth/eA-EApp users. IDEM Program Offices (PO) are responsible for specific requirements over and above established eAuth requirements to determine a registrant's signing authority.

System Functions

All eAuth/eA-EApp systems have an eAuth Application Administrator role. These roles are granted the ability for associated user account credentials to be used for signatory purposes on behalf of the companies the user accounts represent. This signatory role can be granted on a per company basis to user accounts.

All eAuth/eA-EApp systems also have an "authorized agent" role. These roles are assigned to authorized signatory accounts and they are used by and eAuth/eA-AppAdmin to determine whether particular user accounts have submission rights.

A user's access to signatory functions is explicitly determined at run time. Availability of signatory functions within an application is enabled or disabled based upon that determination prior to permitting user admittance to the application. Each eA-EApp performs a check against the appropriate user security database (utilizing the user's logon credentials) to determine the user's signatory rights.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 15: Procedures to flag spurious credential use

Business Practices

eAuth-Administartors (eA-Admin) and eAuth Application Administrators (eA-AppAdmin) review appropriate audit logs to ascertain whether any given user account has been compromised. If it is determined a compromise has occurred for a particular account, the eA-AppAdmin follows established policies and procedures to lock the account and all appropriate follow-on steps (see Items 10, 11, 13c, 15, 16 and Supporting Documentation).

Administrators can potentially ascertain spurious credential use by identifying suspicious activities via semi-automated query functions against audit logs. These query functions provide aggregate figures for a variety of metrics such as multiple failed login attempts, repeated credential validation failures, etc. Administrators have the ability to filter output on a variety of criteria (e.g. date range). Currently these query functions are manually executed via the appropriate administrative interface. Established protocol requires administrators to minimally execute these checks on a weekly basis covering the previous week's entries and periodic checks encompassing previous month, quarter and annual time spans; however, exact schedules are

dictated by each Program Office (PO). Administrators, of course, have the ability to asynchronously spot-check audit logs and are encouraged to do so.

Users are required under the conditions of IDEM Electronic Signature Agreements (IESA) to notify the appropriate eA-AppAdmin upon receipt of a notification of submission s/he did not perform and/or any other unauthorized use of his/her account credentials.

Users are allowed to alter their account profiles; however, the system requires the user answer a secret question (established at registration time, see below) prior to gaining access to their account profile. This helps ensure that the user established the original account in question and was properly vetted

All user changes to an account profile for a facility immediately disable access to that facility. The eA-AppAdmin is notified via system function(s) of the change. The account owner is also notified, via email, of all account profile changes. Thus the original owner of the registered email address will continue to receive account profile modification notifications, even in the event that the account is compromised. Since the system notifies the originally registered account owner of all account-related actions using his/her out-of-band email address, spurious use of the user's credentials would be detected by the registered user. The account owner must undergo another round of identity-proofing before the eA-AppAdmin will reactivate the account.

A change to the user's registered email is handled as a special case. The new email address is not used for notifications (i.e. the vetted email continues to be the email address utilized for user notifications), until the eA-AppAdmin completes the new identity-proofing and re-enables the user's account. At that time, the new email address becomes the email of record.

System Functions

eA-AppAdmin's have the ability to lock the user's account associated their eA-EApp. Users also have the ability to lock their own accounts should they suspect their credentials have been compromised. Accounts not locked at the time of submission provide evidence that administrators and users believe a credential is not compromised at the time of submission.

Spurious use of user credentials may also be detected in the following ways:

1. The IP address and date/time of each originating login attempt is entered into eAuth/eA-EApp logs. Many login attempts in a short period of time and/or from different IP address may an indication of spurious use and/or compromise.
2. Irregular and/or unusual submission patterns.
3. Frequent overlapping login attempts. Overlapping logins (i.e. concurrent sessions) are not allowed and results in the original login being invalidated.
4. Submission notifications are automatically sent to the user's registered email address. Receipt of notifications for submissions not initiated by the user would indicate at least spurious use if not outright compromise.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 16: Procedures to revoke/reject compromised credentials

Business Practices

See Supporting Documentation for the guidelines regarding procedures and timeliness of eAuth Application Administrators (eA-AppAdmin) action when account compromise is suspected.

When notified of a compromised user credential the eA-AppAdmin will immediately lock the user account associated with that credential. The user will then have to undergo another round of identity proofing by the eA-AppAdmin in order to reset their pins/passwords and unlock their account.

System Functions

An eA-AppAdmin is able to lock any user's account associated with his/her eA-EApp. Users are able to lock their own accounts. An eA-AppAdmin or user is required to lock the user's account if evidence suggests the account has been compromised. A locked account can not be used to sign an eAuth Enabled Application (eA-EApp) or log in to eAuth or the associated eA-EApp.

See Item 13c for a discussion on the rejection of compromised credentials by the eAuth/eA-EApp systems.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 17: Confirmation of signature binding to document content

Business Practices

None

System Functions

eAuth/eA-EApp submitters sign documents using a temporary X.509 certificate. The user must supply the correct registered user id, password and the correct answer to a randomly selected user-selected secret question associated with the user's account. The submission process is described in Item 5. Identifying account information is inserted into the COR of the submission prior to signing and upload to bind the identity and submitter's signature to the documents content.

Document integrity is verified in the following manner:

1. The current message digest (hash) value of the received document using the standard SHA-256 algorithm is calculated.
2. Decryption of the received signature hash using the supplied public key is used in order to obtain the original document hash value at signing time
3. The current hash value is compared with the original hash value

If any part of the COR was altered, including the signature binding information, the new signature would differ from the original.

eAuth/eA-EApp systems perform this signature validation upon uploading of the signed submission to the eAuth/eA-EApp system web servers. Failure to pass the signature validation results in a “submission failure” out-of-band email being sent to the registered email address for the submitter and the signature validation failure noted in eAuth/eA-EApp audit logs.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 18: Creation of copy of record

Item 18a: True and correct copy of document received

Business Practices

See Items 5 and 9 for a description of the eAuth/eA-EApp Copy of Record (COR).

System Functions

See Items 5 and 9 for the contents of the COR and the process used to assure it is a true and correct copy of the data.

While in transit, the integrity of the submission document is protected through the mechanisms of the SSL HTTPS connection (see Item 8).

The eAuth/eA-EApp systems will validate each user-signed submission document upon receipt (see Item 13 for methods used).

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 18b: Inclusion of electronic signature

Business Practices

None

System Functions

See Items 5 and 9 for the contents of the eAuth/eA-EApp Copy of Record (COR) and information on how the electronic signature is included in the document.

eAuth/eA-EApp systems retain document signature information and related public keys whenever a submission document or any of its related documents (such as those containing submit-time metadata collection items) are stored.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 18c: Inclusion of date and time of receipt

Business Practices

None

System Functions

eAuth/eA-EApp systems include the date and time of the submission in the eAuth/eA-EApp Copy of Record (COR).

See Items 5 and 9 for more information on the contents of the COR.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 18d: Inclusion of other information necessary to record meaning of document

Business Practices

None

System Functions

The eAuth/eA-EApp Copy of Record (COR) is an XML document containing all appropriate information associated with a submission. The XML contains all tags and data necessary to faithfully recreate the content of the submitters browser screen (content) at the time of submission. Further, the XML tags used in these documents relate the user-supplied data to the context in which the data were provided. eAuth/eA-EApp systems also retain XML metadata in the record associated with COR.

See Items 5 and 9 for more information on what the COR contains and associated metadata.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 18e: Ability to be viewed in human-readable format

Business Practices

None

System Functions

See Item 9b, Item 9c and Item 18d for more information on how the eAuth/eA-EApp Copy of Record (COR) is provided in a human-readable format.

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 19: Timely availability of copy of record as needed

Business Practices

IDEM Program Offices determine the rules for granting access to eAuth Enabled Applications (eA-EApp) and a Copy of Record (COR) submitted via the eA-EApp.

Access to eA-EApp's and any given COR submitted via an eA-EApp is granted to:

1. eAuth Application Administrators (eA-AppAdmin) of associated eA-EApp's
2. Other internal users authorized by IDEM Program Offices (PO)
3. eA-EApp COR authorized submitters
4. Other eA-EApp users authorized by both the PO, the user's authorizing official and whose account profile permits access

eA-EApp's have the ability to manually retrieve and/or provide information related to a COR within 1 business day. The eA-EApp will determine whether a request for COR's or related information is allowed according to appropriate PO rules.

COR images are also stored in the IDEM Virtual File Cabinet (VFC) document management system. See Item 20 for a discussion of COR image storage and access. If appropriate flagged, VFC users are able to retrieve COR's in real-time.

System Functions

eAuth/eA-EApp systems generate and store COR's during the submission process. A notification is immediately generated and sent to the email address of the COR submitter.

See Item 9a for information related to submitter notifications.

Each eA-EApp has COR search/viewing functionality by which a COR can be reviewed on-line and/or downloaded for offline review by user's with sufficient authority. The necessary authorizations, search parameters and/or processes to locate a COR are eA-EApp specific.

Additionally COR's can be searched, viewed and downloaded via the VFC (see Business Rules, above and Item 20) again subject to sufficient user authority.

COR's will be searchable, viewable and downloadable for the entire length of time for which they are maintained on in the eAuth/eA-EApp systems (see Item 20).

Supporting Documentation (list attachments)

Attachment 1 – eAuth.doc

Item 20: Maintenance of copy of record

Business Practices

Separation of infrastructure and application administration

Indiana State Information Technology personnel are functionally and physically separated into two groups. The duties of each are well defined for eAuth and eAuth Enabled Applications (eA-EApp). This separation helps prevent unauthorized access to/unauthorized manipulation of systems and/or data.

Infrastructure Personnel

1. The Indiana Office of Technology personnel maintain and service network hardware, operating environments and user access to/security for State information technology resources.

Application Personnel

1. The IDEM Office of Technology personnel develop/maintain Agency applications, maintain/administer Agency application user security and maintain/administer Agency databases.
2. The IDEM Program Offices (PO) assign eAuth Application Administrators (eA-AppAdmin) to perform/mediate eA-EApp-specific identity-proofing, user security and manage retention schedules (established by individual POs)

See Chapter 4 for more information

System Functions

The eAuth/eA-EApp system stores/retains the Copy of Record (COR) associated with user submissions in multiple locations and formats and are created in real-time or near real-time. The distributed nature of the various COR copies makes it extremely unlikely tampering with the COR will go undetected.

Oracle Database

Submissions received via the eAuth system are received as binary files and stored in Oracle databases which natively support XML formats. Each submission is stored in one database record.

The following table provides descriptions of constituent fields and their purpose(s):

Unique Document ID	Reference number for the submission.
XML metadata	Document specific information as follows: 1. document id 2. submission source; (i.e. the specific eAuth/eA-EApp system which mediated the submission) 3. submission type (e.g. eDMR) 4. submission document type (e.g. XML, MS-Word, etc.) 5. submission date and time 6. submitter login 7. submitter name 8. IP of submitting computer 9. post submission hashing certificate public key 10. post submission SHA-256 hash of the submission 11. hashed data document 12. image type 13. other information necessary to establish pertinent COR data associated with content stored in the BLOB fields.
Erroneous Submission flag	Boolean indicator signifying signatory has indicated an erroneous submission.
Repudiation Flag	Boolean indicator signifying signatory has repudiated the submission
*Native file format BLOB**	binary submission
Image BLOB	image representation of the binary submission; supported images types are : pdf and tiff

*BLOB - Binary Large Object; a BLOB field has no structure which can be automatically interpreted by the database management system.

The Unique ID and XML Metadata fields serve as logs and provide convenient search capabilities (as specified in Item 19) directly from the database.

The eAuth/eA-EApp systems maintain various logs that could provide supplemental information to that stored in the COR.

Post Submission Document Hashing

1. At the completion of the submission of the COR, the eAuth/eA-EApp system computes a SHA-256 hash value of all the items that make up the COR. This hash value is then signed using an IDEM server private certificate. This COR signature value (and information regarding it) is saved within the database. This effectively allows detection of COR tampering.

Retention

1. COR's are retained indefinitely, but never less than retention schedules established for the particular type of submission. If no retention schedule is specified for a particular submission type, the default retention period is 5 years. See Supporting Documentation for the additional information related to COR retention schedules.
2. eAuth/eA-EApp logs are retained for 1 year.

Physical Security

1. See the supporting documentation for the more information on the physical security.

Infrastructure

1. Oracle databases are maintained on servers providing storage via a Redundant Array of Independent Disks (RAID). These RAID systems detect and address any hardware-related storage errors. To address DBMS vendor-related errors, eAuth/eA-EApp systems employ automated database backup procedures that allows for rollback/recovery of database objects at nearly any point in time.

Backups

1. All eAuth/eA-EApp system files (including the databases) are automatically backed up on magnetic tape, on a daily (incremental), weekly (full), monthly (full), quarterly (full) and annual (full) schedule for off-line storage. A typical rotation schedule is observed in order to maintain the ability to recover files to any point in time down to a granularity of 1 day. Annual backups are permanently stored off-line.
2. See the Supporting Documentation for more information relating to database backup procedures.
3. All eAuth/eA-EApp databases are included in the Indiana Office of Technology (IOT) Disaster Recover Plan and are subject to a 6 hr / 7 day recovery period.

See Supplemental Documentation for information relating the above.

Virtual File Cabinet (VFC)

IDEM supports and maintains the IDEM Virtual File Cabinet (VFC) document management system. The VFC offers a robust set of on-line and off-line tools to store, index, search and retrieve document images. VFC also supports a public portal for search and retrieval of appropriately flagged documents.

eAuth/eA-EApp COR submissions are converted to an image format - pdf or tiff (see Supplemental Documentation). Creating an image of the COR effectively creates a static snap-shot of the submission. Each image is stored in the Image BLOB field of the Oracle database record associated with the appropriate individual submission. The image is also stored in the VFC system and cross-references created to associate the image with the appropriate Oracle database record and a variety of indices maintained in the VFC (e.g. company, document type, etc.)

VFC has the ability to flag any document, including COR's, as sensitive (restricted to a small group of authorized internal IDEM viewers), internal (IDEM eyes-only) or available for public scrutiny.

See Appendix 6 for more information

Supporting Documentation (list attachments)

Attachment 1
Attachment 9
Attachment 12
Attachment 13
Attachment 14
Attachment 15
Attachment 16