Private Key Protection

Your private keys shall never appear outside of the module in which it was generated in plain text form.

Standards for Cryptographic Modules - For Medium Assurance (software) certificates, Subscribers shall use cryptographic modules that have been validated to meet at least the criteria specified for FIPS 140 Level 1. FIPS 140 Level 2 compliant tokens must be used in the case of a Medium Hardware Assurance certificate.

All ORC ECA certificates will be signed using a hardware cryptographic module that has been validated to meet FIPS 140 Level 3. The ORC ECA private keys are protected by a hardware cryptographic module that meets the criteria specified at FIPS 140-1/2 Level 3 for key storage, as listed on the NIST website.

All cryptographic modules are operated such that the private asymmetric cryptographic keys are never being output in plaintext. No private key shall appear unencrypted outside the ORC ECA, IA or CSA equipment.

No one shall have access to private signing key but the Subscriber. Any private encryption keys held by a CAA shall be held in strictest confidence and controlled as described in the US Government ECA Key Recovery Policy (KRP) and the ORC ECA KRPS. The ORC Key Recovery System only employs cryptographic modules validated to the FIPS 140-1/2, as identified in the ORC CPS.

Private Key Operational Copies - ORC recommends to users that they make operational copies of software based encryption private keys. For help exporting your certificate for operational copies, please go to Instructions on the top navigation bar. Copying of private signature keys for the sole purpose of key recovery shall not be made. Operational copies shall be stored in an encrypted form and shall be protected by a password from unauthorized access. The Subscriber (PKI Sponsor for Component) is responsible for ensuring that all copies of private keys, including those that might be embedded in component copies, are protected, including protecting any workstation on which any of its private keys reside.

Method of Activating Private Key - A password will be used to activate all private keys for IA, RA, LRA, subscriber medium assurance and medium hardware assurance. Passwords shall be generated by the subscriber and entered at the time of key generation (at the RA/LRA workstation in case of medium hardware assurance) and managed according to the FIPS 140-2 (Section 4.3.3) guidance, in accordance with the subscriber obligation agreement. Entry of activation data will be protected from disclosure. The strength of the passwords and the controls used to limit guessing attacks shall have a probability of success of less than 2^{-23} chance (1 chance in 8,338,608) of success over the life of the password. ORC uses the NIST E-Authentication TSM to calculate resistance to online guessing. The strength of the password shall be 8 characters with the following diversity: one upper case alpha, one lower case alpha, one numeric, and one special.

Method of Deactivating Private Key - Cryptographic modules that have been activated shall not be left unattended or otherwise active to unauthorized access. Private keys stored in hardware tokens, excluding end user tokens, shall be removed from the token reader (deactivating access) and stored in a locked container when not in use. End users will protect there token in accordance with the subscriber obligation agreement. The ORC ECA hardware token shall be stored in accordance with Section 5.1.2 of the ORC CPS when not in use.

Private keys stored in software shall be deactivated via a logout procedures. End entities will be advised to also implement a time-out procedure for automatically deactivating private keys after a period of 15 minutes of non-use.

Method of Destroying Private Key - Private keys shall be destroyed when they are no longer needed, or when the certificates to which they correspond expire or are revoked. In order to destroy a Private Key stored on software cryptographic modules, the subscriber can overwrite the data. In order to destroy a Private Key stored on hardware cryptographic modules, the subscriber will need to execute a "zeroize" command. Physical destruction of hardware should not be required.