edu.Library

CERTivity® Release Notes

October 18, 2012

Document Version 1.1.0
CERTivity® 1.1

www.edulib.com

edu.Library

Legal Notice

Copyright

www.edulib.com

Calea Bucuresti, Bl. 27B, Sc. 1, Ap. 2
Craiova, DJ, 200675, Romania
Phone: +40 351 420970
Fax: +40 351 420971
E-mail: office@edulib.ro

Table of Contents

1. Changes in CERTivity 1.1

Release Date: 2012-10-18

1.1 New Features

- Opening the machine's JRE CA TrustStore(s) - Added support for opening the CA TrustStore(s) of the JRE(s) discovered on the current system. We search for the TrustStores in the following locations:
  - The Java property `${java.home}` of the JRE CERTivity started with;
  - The system environment variables `JAVA_HOME` and `JRE_HOME`;
  - For Windows platforms searching the installed Java JDKs and JREs in the Windows registry;
  - For Unix and Mac we are looking for traditional Java instalation directories such as `/usr/java` for Unix, `/usr/lib/jvm` for Linux (Debian, RedHat) and for Mac `/Library/Java/Home/`, `/System/Library/Java/JavaVirtualMachines/`. Varios patterns are then used.

  If the KeyStore persistence settings in `Tools > Options` is set to "`Fully persist (file name & encrypted password)`" the passwords of the TrustStores are saved in the preferences after closing the application.
- Support for transition to secure RSA keys - As Microsoft announces that the use of certificates that have RSA keys that are less than 1024 bits long will be blocked, and in the future other systems and even Java Virtual Machine may do this as well, we have introduced features to easily spot such certificates, and to warn when generating RSA Key Pairs with less than 1024 bits. Also, we have made the minimum size allowed for generating RSA Key Pairs configurable and it can be set in the `Tools > Options`. The minimum size can not be less than 1024.

  The Certificates and Key Pairs which have RSA Keys that are less than the minimum size set are now marked in the KeyStores by having the name and the Key Type / Size colored in red. The Key Type / Size is also colored in red in the certificate panel.

  Another measure is that the default out of the box RSA KeyPair size is now doubled to 2048, too. This value can, at any moment, be increased by users (from `Tools > Options`) and it will remain persistent for future uses.
- Signing APK (Android Application Package) files and verifying signatures on signed APK files - Added support for singing and verifying signatures on APK files in the Sign JAR and Verify JAR actions. By selecting the APK file filter (or by selecting the "All files" file filter), the user is now able to select APK files and sign them in a similar manner as it is done for signing JAR files or for verifying signatures on signed JARs.
- SSL Certificates Retriever details using HTTPS URLs - Added an additional field to the SSL Certificates Retriever for entering an URL (HTTPS protocol) from which the host and port will be parsed. If the URL does not start with a protocol designator the `https://` one will be added automatically. Also, if the URL has no port specified, the default port for HTTPS protocol (443) will be used. The parsed host and port will also be set into the initial fields "Host name" and "Port", which can still be used.
- Importing certificates from the signature verification results of the verify JAR, PDF and XML actions into the active KeyStore (the active KeyStore tab) - Implemented support to allow importing a selected certificate found in the signature of a JAR, PDF or XML file

into the active KeyStore (the active top component), directly from the verification results panels of the verify JAR, PDF and XML actions.

- Basic display of certificate extensions - added support for identifying and displaying the detailed content of the following certificate extensions:
  - Authority Key Identifier;
  - Basic Constraints;
  - CRL Distribution Points;
  - Extended Key Usage;
  - Key Usage;
  - Netscape Cert Type;
  - Private Key Usage Period;
  - Subject Key Identifier.

  For the extensions which are not identified, the OID will be displayed and a rough representation of the extension content.
- ASN.1 display of each certificate extension - added support for displaying the ASN.1 representation of the content of each extension of the certificate (even for the ones for which we display only the rough representation of its content).
- Extensions for certificates when generating Key Pairs - Implemented support for adding certificate extensions to Certificates when generating Key Pairs using an easy to use tree-like structure for adding and representing the extensions. The following extensions can be added:
  - Authority Key Identifier;
  - Basic Constraints;
  - CRL Distribution Points;
  - Extended Key Usage;
  - Key Usage;
  - Netscape Cert Type;
  - Private Key Usage Period;
  - Subject Key Identifier.

  The extension structure is validated in real time, as the extensions and their subitems are being added, showing information about the validity status, and details about the validation errors (if any). Also, some values from some extension fields (such as the Directory Name components from the General Name of Auhority Certificate Issuer component of the Authority Key Identifier extension, and others) are filled automatically using the values provided in the certificate fields (if they are not empty). More than this, when adding extension subitems which require some information from the certificate which was not provided yet, for example if the Authority Certificate Serial Number is required and a Serial Number was not provided yet, the information can be generated at that time, and it will also be set for the corresponding certificate field. Also, the mandatory subitems of the extensions are added automatically in the tree structure, to ease the

job of the user, and for the optional ones, the user can select which one to use from a context menu.

- Handling extensions for CA Reply certificates when signing a CSR file - Implemented support for adding certificate extensions to the CA Reply user certificate as well, which will be obtained after signing a CSR file. The mechanism for creating and adding extensions is the same one described for adding extensions when generating a new Key Pair, with the small difference that the information regarding the issuer is now taken from the signer certificate.
- Viewing extensions structure at creation time as XML - Implemented support for viewing the tree like structure of the extensions in XML format, for easy visualization or for easy copying into other documents. Each extension is a node, and each subitem of an extension is a child node of the extension node.
- Saving extensions structure at creation time as XML templates for later usage and loading them back into the tree structure - Added support for saving the extensions structure as an XML document into a file as an extensions template for later usage. These templates can be loaded later, when generating a new Key Pair, or when creating a CA Reply, which needs for its Certificate a similar or maybe identical extensions structure.
- Reopening the last used files (Key Stores, Certificates) - Added "`Open Recent File`" menu (in the `File` menu), to allow reopening the last used files like KeyStores and Certificates (and the "Readme" file). For the KeyStore files, if the KeyStore persistence option in the `Tools > Options` is set to "`Fully persist (file name & encrypted password)`", the passwords will also be stored, so that the user won't have to re-enter them again when reopening the KeyStores. Otherwise, the user will be prompted to enter the KeyStore password again.
- Renaming an entry by the `F2` key - The `F2` key is the default standard for the Windows Users for renaming, so we also use it in CERTivity for renaming entries, to improve usability, besides the existent shortcut `Ctrl + R`.
- Generating a new Key Pair using the information from an existing Key Pair - Implemented functionality to allow the user to generate a new Key Pair using some information from an existing Key Pair such as:
  - Key Type (RSA or DSA);
  - Key Size;
  - Certificate Version;
  - Signature Algorithm;
  - Certificate Issuer / Subject Distinguished Name (Common Name (CN), Organization Unit (OU), Organization Name (O), Locality Name (L), State Name (ST), Country (C), Email (E)).
- Opening Software Publisher Certificate (SPC) Certificate Files.
- The CA Reply file chooser contains the "All files" filter - This way, the user can open and import a CA Reply even if the file name does not contain the required file extension, if the file really contains a CA Reply.
- Opening the "Readme.txt" file on the first run of the application - Now the "Readme.txt" file is opened and displayed as a new tab in CERTivity on the first run of the application. This file can also be later opened using the `File > Open Recent File` menu (if the file is still in the recent files list). This file contains details, such as passwords, related to the samples provided with CERTivity.

- The "Validity" column from the KeyStore view was renamed to "Validity Status".

1.2 Bug Fixes

- JAR Signature Block compatibility - The Signature Block file (e.g. *.RSA) was bigger than the one produced by JDK's `jarsigner` for the same jar, the same key because there were two equivalences one at logical level, and one at binary (encoding) level. Although the `jarsigner` from Sun (Oracle) JDK was OK with these equivalences we are now using really the same, unequivocal productions by involving the Distinguished Encoding Rules (DER) and by using a PKCS7 with a direct signature, without the default signed attributes. This makes CERTivity compatible with other tools than JDK's, such as the Android SDK tools.
- Focus lost and entry selection issues - Fixed the focus lost and entry selection issues after operations involving KeyStores, including Undo/Redo. Now the focus remains in the Tree Table (allowing navigation through the KeyStore and performing actions using the keyboard) after performing operations such as deleting or adding new entries, converting KeyStore type, Extend Validity period for self signed Key Pairs. Also now the entries remain selected after performing Undo / Redo operations (for example after performing Undo on a delete operation, the entry will be selected). A detailed list of fixes regarding the focus and selection issue can be seen below:
  - After adding entries (by generating, importing or pasting new entries) and using Undo and then Redo, the entries remain now selected (either if there is one entry or more entries), and the focus is in the Tree Table;
  - After deleting or cutting an entry, when performing Undo, the resurrected entry is now selected in the tree, and the tree has the focus allowing navigation;
  - After performing Undo/Redo on a rename operation, the entry which is being renamed remains selected and the focus is in the tree;
  - After changing the password of an entry, the entry remains selected, as well after performing Undo or Redo on the password change action;
  - After extending the validity period of a self signed Key Pair, the entry remains selected and the focus in the tree, as well after performing Undo or Redo on the Extend Validity action;
  - After performing Undo / Redo for Importing the CA Reply, the entry for which the action takes place remains selected and the focus in the tree;
  - After closing modal dialogs invoked through contextual menus, the focus and selection remain in the tree.
- Contextual Help for the `Tools > Options` panel describing the Options from JavaHelp - Now the Help from the Options panel points to the description of the Options in the JavaHelp, and not to the default help anymore.
- Line terminator in "`${certivity.home}/etc/certivity.conf`" configuration file is now platform specific - The line terminator from the file "`${certivity.home}/etc/certivity.conf`" consisted of only one LF (Line Feed), and this could have been problematic on Windows platforms when trying to edit the configuration file using a simple editor such as Notepad. A fix was made to change the line feeds from the mentioned file to the platform specific type at installation time.
- Corrections and uniformization for the File Choosers - For some actions such as opening KeyStores or Certificates the default file filter for the file chooser was "All files". This was

updated so that the default selected filter to be the one in cause (e. g. for Verifying PDF it must be PDF files, for key store files it must be KeyStore Files).

- Clipboard shortcuts on Mac OS X - The clipboard shortcuts (`Meta-C`, `Meta-V`, `Meta-X`) on MAC OS X are now correctly functioning in the Tree Table for KeyStore entries.