Jetico

BCArchive

Help File

Introduction

The BCArchive software is designed to compress group of files or folders to encrypted archive (i.e. a single compressed file).
There are many compressing utilities well known in the world (Zip, Arj, RAR and others) and they are really helpful to save space on hard drives or when we backup data, or send the data over network. From the other hand, when we save our data for a long time, in many cases it is not only an important data, but it is a private data. The question of privacy arises even more when we send the data over insecure Internet connections.

Fortunately, strong encryption standards, algorithms and technologies, developed and evaluated by many individuals and organizations from all over the world, may help people a lot to keep their private data in a secure form. The encryption technologies may help and they really help when people can use the technology for a concrete case with a tool, which:

- Is the most suitable for the concrete situation
- All the steps in securing data use standard (and proven as strong) algorithms
- Not experienced in encryption users could work comfortably

For example, you may wish to create compressed and encrypted archive and store it as a backup copy on your hard drive. In this case you only need to enter a password for the archive when you create it and enter the same password when you want to decrypt and extract the data from the archive. Looks simple and does not require special knowledge in encryption from the user. But the encrypting part of the software must use strong algorithms, for example, IDEA or Blowfish encryption algorithms and PKCS #5 standard for derivation encryption key from the password string. In other case it is possible that the encrypted archive can be easily decrypted by some freeware utility.

If you wish to create encrypted archive and send it to another user, you may use more advanced technology, based on **public/secret key pair** encryption algorithm. In this case you don't need to tell the other user the password for the encrypted archive. Instead, you encrypt the archive file with a public key of the user who is going to receive the file. Public key of the user is not a secret and can be sent to you in any way. At the same time the user keeps a secret key, corresponding to the public key, in a very safe place. Once you have encrypted the archive file with a public key, the file can be decrypted only by corresponding secret key. Hence, only your recipient can decrypt the archive file you send to him, because no one else knows his/her secret key. This technology is used in widely known **Pretty Good Privacy (PGP)** software and a number of specifications and standards were issued to make using the technology as secure as possible.

Public/secret key algorithms are very convenient, because you do not need to discuss with your recipient what password you are going to use for encrypting the data. But you still need your recipient have the same software installed on his/her computer to be able to decrypt the data you send. It is not very convenient in many cases to ask your recipient to install something before he/she starts to access the data. Solution for the case exists – you can create so-called **self-extracted archive**. It means that you not only compress and encrypt the data to a single archive file, but also convert the file to an executable program. Later you or your recipient can run the program on computer without any special software installed and get the data extracted.

Main Functions

Main function of the BCArchive is to provide the user with a tool for compressing and encrypting files and/or folders to a single archive file. From one hand, BCArchive provides a number of ways to create such an archive file and put sensitive data into it, including Windows shell extension commands and support of drag-and-drop operations. From the other hand, BCArchive utilizes strong and proven encryption algorithms and standards to provide high security level for the data stored inside the compressed files.
The section lists main BCArchive functions and gives references on other chapters in the documentation for more detailed information.

Compressing and Encrypting Data

- Create new archive file - To get group of files/folders encrypted and compressed into a single archive file, you should make two steps. First, create new archive file where your data will be stored. At that step you choose public key or enter password, which will be used further to extract the data from the archive. Second step is to place files you have selected into the archive file. Several ways are possible here, including **drag-and-drop** selected files/folders to the BCArchive window or running BCArchive ***Add Files/Add Folders*** commands.

- Open/View/Edit existing archive - Once you have created BCArchive file, you can access the data stored inside it in compressed and encrypted form. To open the archive for access just double-click on the archive file in My Computer window or run the ***Open*** command in the main BCArchive window. The program will ask you to enter password and then show the contents of the encrypted archive. After that you can add new files and folders to the archive, remove some old files from the archive, extract selected files and even open files directly from the BCArchive window without the step of extracting it in decrypted form from the archive.

- Add Public Key/Password to existing archive file - You can create compressed/ encrypted archive for your own use. If so, you can make the archive accessible with a single password (it may be the password for your secret key or just a password for the concrete archive). From the other hand, if you wish other people were able to access the data inside the archive, you can add a number of passwords to the archive. Besides, if some person has his/her own public/secret key pair, you can add public key of the person to the archive. There are many servers in Internet where public keys of thousands of people are stored. BCArchive allows you to run searching for user's public key in Internet and add the key to the archive you have created. Since BCArchive supports PKCS #12 and X.509 standards, you can use existing public key servers to download and use the keys of other people.

- Make Self-Extracted archive file - To use all the functions available in the BCArchive software you should have the program installed on your computer. From the other hand, you may wish to create encrypted/compressed archive file for backup purposes or for sending it to another user. It is possible that the person who will try to extract data from the archive has no the BCArchive program installed. In this case you can transform the archive file to self-extracted executable file (i.e. program file that contains executable code and encrypted/compressed data). As soon as some other person runs the self-extracted program you have created, the program will ask him/her to enter an appropriate password and extract the data in decrypted form on the user's computer.

- Synchronize, Import, Export functions - The contents of the archive can be synchronized with the contents of a non-encrypted folder. As well, ***Import from...*** and ***Export to...*** functions are also available.

- Encode text or clipboard contents - BCArchive includes **BCTextEncoder** utility intended for fast encoding and decoding text data. Plain text data are compressed, encrypted and converted to text format. The result of such conversion may be copied to the clipboard or saved as a text file.

Local Public Key Database and Key Management

- Create Secret/Public Key Pair - A lot of people around the world have their secret/public keys. They make their public keys opened for everyone and keep corresponding private keys in a secure place. Public key is used to encrypt data so that only an owner of corresponding private key can decrypt the data. You may also wish to create your own secret/public key pair so that other people can send you data encrypted by your public key. BCArchive allows you to create the key pair and store/send it in a standard PKCS #12 format as well as issue corresponding public X.509 certificate.

- Add Public Key to Local Public Key Database - BCArchive supports its local database for public keys of other people you may wish to correspond with. For example, you decide to send encrypted and compressed archive to your friend John. John may have his public key created earlier and stored on many Public Key Servers in Internet. If so, you can run BCArchive ***Add Public Key*** command and find John's public key in Internet. Then you can add the public key to BCArchive file and send the file to John. John will be able to decrypt the archive with his secret key. Since you have saved John's key in BCArchive local Key Database, next time you decide to send encrypted archive to John, you won't need to access Internet Key Servers again, you will simply get the key from your local database.

- Backup/Restore Local Public Key Database - BCArchive database of public keys of other users saves your time, because when you add public key of other user to encrypted archive, you do not need in accessing Internet to download the public key again. It is recommended to backup (or export) the database file regularly and save the file on a reliable storage medium. If in future you decide to change your computer or reinstall the software, you can restore (or import) the BCArchive database from the backup copy of the database.

BCArchive Specifications and Limitations

BCArchive utilizes the following encryption algorithms, standards and specifications:

- Symmetric algorithms: Twofish, Blowfish, Blowfish-448, GOST, Rijndael (AES), IDEA, Triple-DES, CAST5, Serpent.
- Secure Hash Algorithms: SHA-1, SHA-256, MD5 and RIPEMD-160.
- Asymmetric (public/secret key pair) algorithms: RSA, ElGamal / Diffie-Hellman.
- Specifications for public/secret key pair format: PKCS #12, X.509.
- PKCS #5 (recommendations for the implementation of password-based cryptography).
- RFC 2440 specifications for session keys encrypted by symmetric or public key encryption algorithms.

BCArchive Limitations:

- Compressed and encrypted archive file created by the BCArchive software can store up to 2 Terabytes not compressed data.
- Self-extracted archive file contains all executable code necessary to extract data stored inside the archive. The code includes corresponding symmetric encryption algorithm, public/secret key algorithm, secure hash algorithm implementations as well as the code to uncompress data. Size of the code is about 150 Kbytes, hence minimum size of self-extracted archive is about 150 Kbytes. The limitation does not concern regular (not self-extracted) archive files.

BCArchive Requirements:

BCArchive requires the following minimum computer configuration:

Hardware

- Minimum 7 MBytes of free HDD space to install and run the BCArchive software.

Software

- Windows 7 32 and 64 bits
- Windows Vista 32 and 64 bits
- Windows XP 32 and 64 bits
- Windows 2008 Server
- Windows 2003 Server
- Windows 2000
- Windows NT 4.0 (Workstation or Server)
- Windows 95, or Windows 98/98SE
- Windows ME

How to install BCArchive

BCArchive Setup program uses standard Windows way to install the software and provides all necessary explanations of the installation details. The only default information the user may want to change during installation is the **Program Folder** name for the BCArchive software and the **Destination Directory** name where to place BCArchive files.
All dialog windows of the Setup program have the following buttons:

Cancel - click this button to abort installation
Next - click this button to proceed with the installation
Back - click this button to return to the previous step

NOTE: BCarchive Setup program also writes data to the **Windows Registry** database, places dynamic load libraries in the system WINDOWS\SYSTEM directory, and prepares a file for the uninstall procedure. Please do not perform any manual manipulations to install or uninstall the BCArchive software in order to prevent appearance of unused garbage software in the system directory or unused strings in Registry database.

Compressing and encrypting

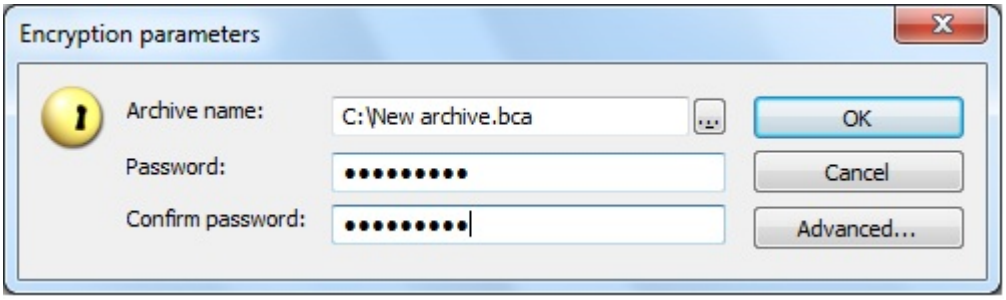
- Create New Archive File
- Open existing archive file
- Editing archive file
- Synchronize, Import, Export functions
- Searching files inside archive
- Automatic opening document stored inside archive
- View file inside archive in binary form
- Add Public Key/Password to existing archive file
- Send archive file as an e-mail attachment
- Make Self-Extracted archive file
- BCArchive Extension for Windows Shell
- How to run BCArchive from the command-line prompt
- Dynamic/manual compacting of archive file
- Text Encoder

Create New Archive File

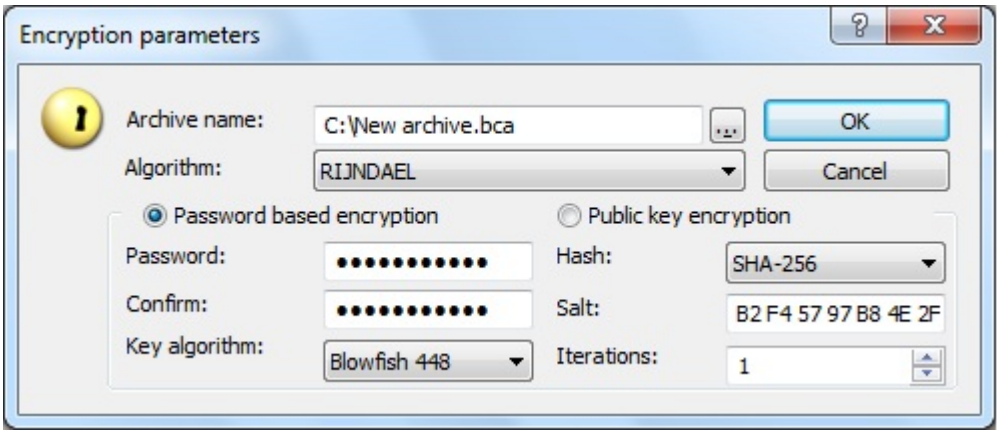
To get your data encrypted and compressed into a single archive file, you should create new archive file where the data will be stored. As soon as you create new archive, you can add regular files and folders to the archive.
There are the following ways to create new archive file:

1. In My Computer window open the folder where you want to create new archive file. In the ***File*** menu of the My Computer program run ***New -> Jetico BestCrypt Archive*** command.
2. Run BCArchive program from the ***Start -> Programs -> BCArchive*** program folder. Then in the BCArchive main window run ***New*** command from ***Archive*** menu.
3. Right-click on a file/folder in My Computer window and run the command ***Add to "file_name.bca"***

As soon as you start creating new archive file, the following window appears:



If you want your new archive to be accessed by entering some password, just enter the password twice in the ***Password*** and ***Confirm password*** edit boxes and click **OK**. Then BCArchive will ask you to make random movements by mouse to generate unique encryption key and the process of creating the archive file will be finished.
The process of creating a new archive can also be more creative work. If you decide to make new archive encrypted by some public key, or you wish to select some concrete encryption options, click **Advanced** . The BCArchive dialog window will start looking like:

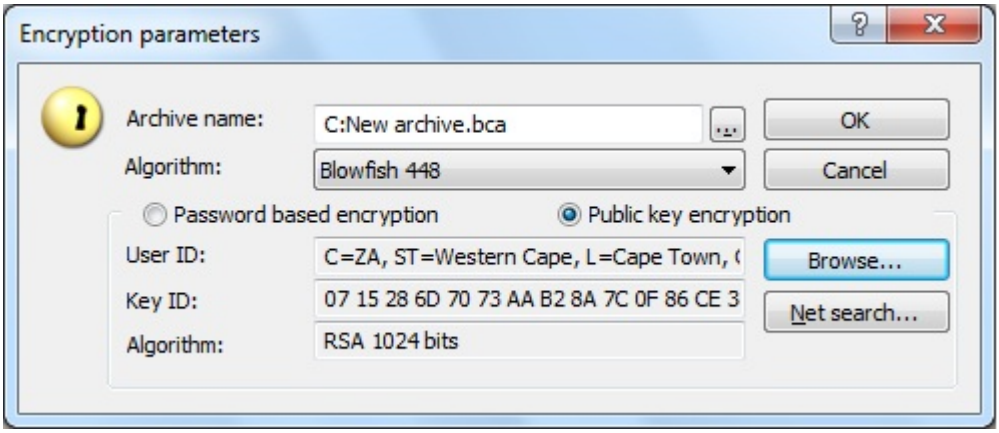


Since ***Password based encryption*** radio button is selected, you continue creating new archive that will be possible to open with a password, but now you can change a number of encryption settings for the archive file:

- Encryption algorithm used to encrypt data inside the archive (Triple-DES in our example);
- Encryption algorithm used to encrypt randomly generated key for the archive (Blowfish in our example);
- Secure hash algorithm for the process of deriving intermediate encryption key from the password (SHA-1 in the example);

- ***Salt*** (random data) and ***Iterations*** parameters are used exactly as it is required by PKCS #5 specification (recommendations for the implementation of password-based cryptography).

If you wish to create new archive so that it will be decrypted by secret key of some user, you should encrypt the data by corresponding public key of the user. Hence, you should select the ***Public key encryption*** radio button as it is shown below:



After that you only need to select public key of the user (probably yourself). You can take the public keys from two places:

1. Find the public key in your **Local Key Database** (click `Browse`).
2. Find the public key in Internet (click `[Net search]`).

As soon as you select public key of some user, BCArchive will display its characteristics in the ***User ID***, ***Key ID*** and ***Algorithms*** boxes so that you can verify validity of the selected key.

See also:

Editing Archive File
BCArchive Extension for Windows Shell
Local Public Key Database and Key Management

Open existing archive file

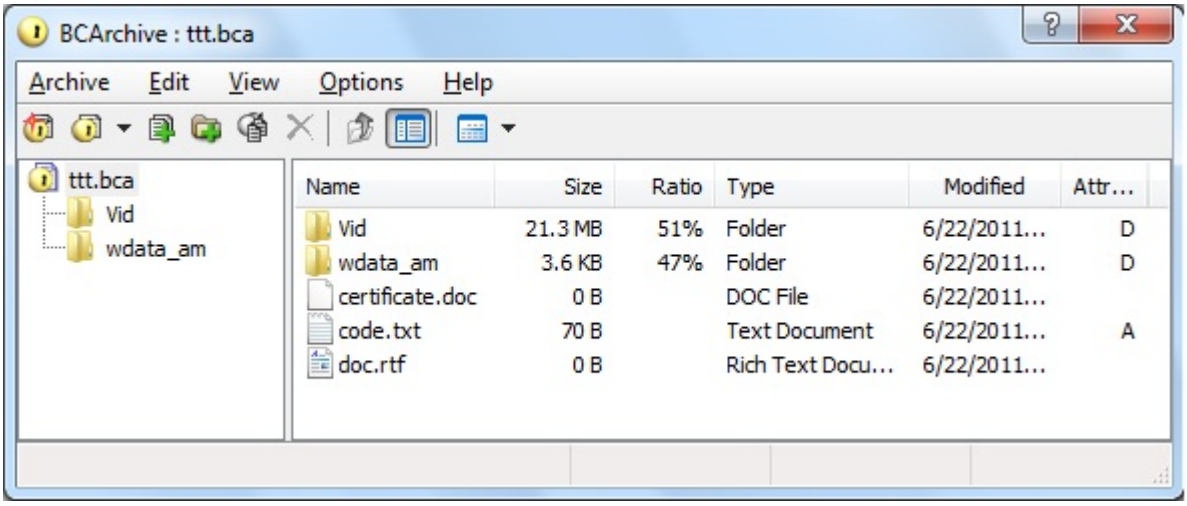
To open existing archive file you can use one of the following ways:

1. Browse the archive file in **My Computer** window and double-click on the file by left mouse button.
2. Run BCArchive program from ***Start -> Programs -> BCArchive*** program folder. You may run BCArchive from Windows **Quick Launch bar** or the icon on your desktop, if you made BCArchive Setup program create the icons during installation.
   Then in the BCArchive main window run ***Open*** command from ***Archive*** menu.
3. If you opened the archive file earlier, you may use ***Open Recent*** command from ***Archive*** menu of BCArchive main window. Optionally, the Recent list may be disabled.

BCArchive will ask you to enter an appropriate password for the archive file and will show you the contents of the archive.

Editing archive file

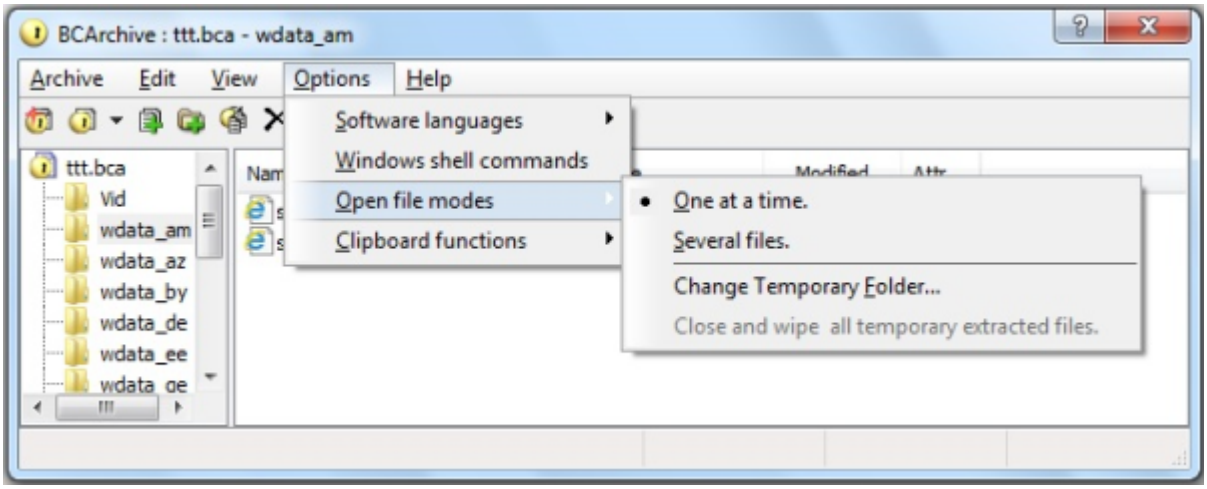
When you open BCArchive encrypted archive file, the program shows you contents of the archive:



You can use a number of ways to edit contents of the archive file:

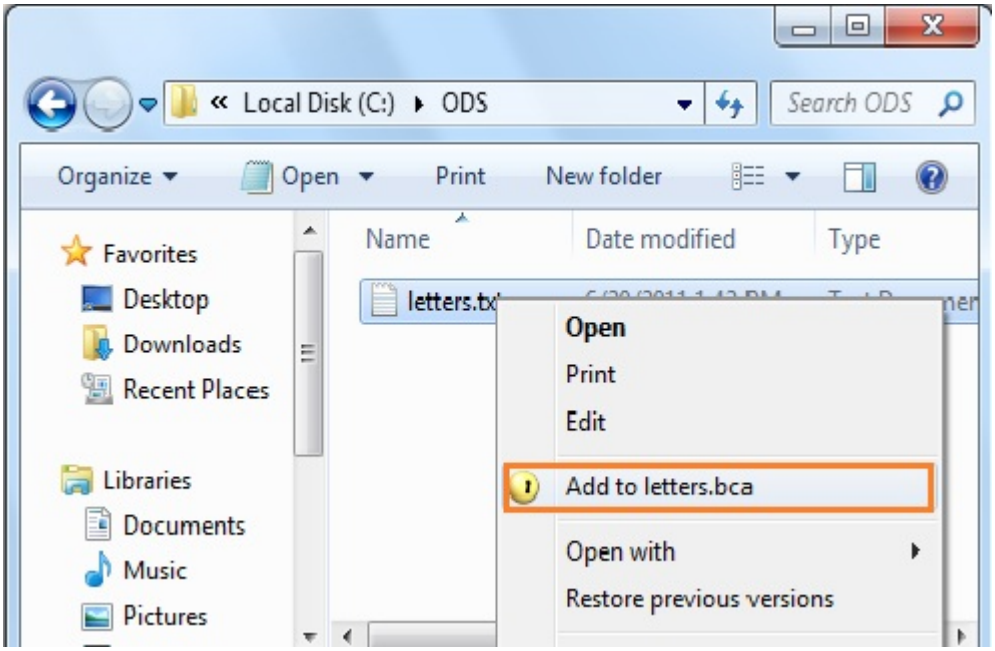
1. **Drag-and-drop** operations - To add files/folders to the archive run **My Computer** program and select group of files/folders you want to add to the archive. Then drag the group of files to the **BCArchive** window and drop it to the window. BCArchive will add the file to the opened archive. You can use the same drag-and-drop way to extract files/folders from the archive – just select the files/folders in the **BCArchive** window and move the files to the **My Computer** window.
2. Use BCArchive commands - You can run ***Add Files*** and ***Add Folders*** commands to add files or folders to the archive, ***Delete*** command to remove files/folders from the archive and the ***Rename*** command to rename some file or folder inside the archive.
3. Run ***Extract All*** command to extract all files/folders from the archive to chosen folder on your computer. If you wish to extract a part of the archive, select the files/folders in the BCArchive window and run the ***Extract*** command from the ***Archive*** menu.
4. **Editing files stored inside the archive** - If you double-click on some file inside archive, BCArchive will automatically extract the file to temporary folder and run corresponding application to open the extracted file stored in the temporary folder. For example: if you open a text (*.txt) file the file will be opened by Notepad. By default, the temporary folder is Windows TEMP folder, but it can be changed using ***Open File Modes --> Change Temporary Folder*** command in ***Options*** menu.

Since BCArchive v2, there are two modes of opening files. You can switch them using the command ***Open file modes*** from BCArchive ***Options*** menu:

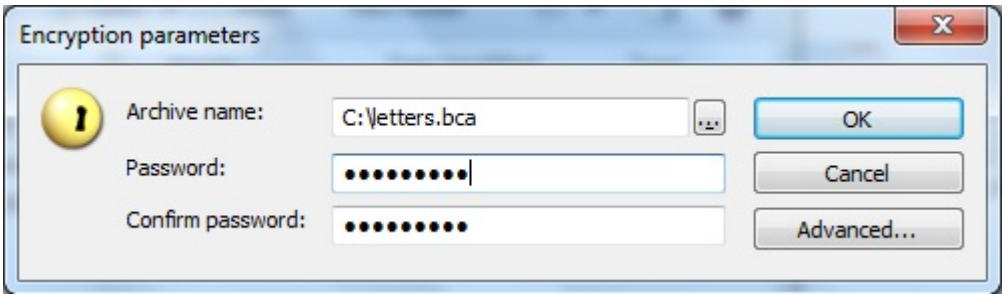


- ***One at a time -*** Only one file can be extracted at a time. You can view and edit the file. When you close the file, BCArchive will ask if you want to update the original file inside archive. After that BCArchive runs a secure deleting procedure (wiping process) to remove temporary copy of the file from the TEMP folder.
- ***Several files -*** You can view and edit several files simultaneously. If a modification is detected, you will be asked if the original file inside archive must be updated or not. Wiping procedure is performed for the whole group of extracted files, after closing the archive, or when the special command (***Open file modes-->Close and wipe all temporary extracted files***) is launched. This mode is less secure, but more convenient.

5. You can add file or group of files/folders to the archive file in **My Computer** window - Select the file(s) and right-click with a mouse. In the appeared context menu run the ***Add to [Filename].bca*** command.



[Filename] is the name of the first file in the group of files you want to add to the archive. If the ***[Filename].bca*** archive does not exist, BCArchive will ask you to enter password for it and create the archive.



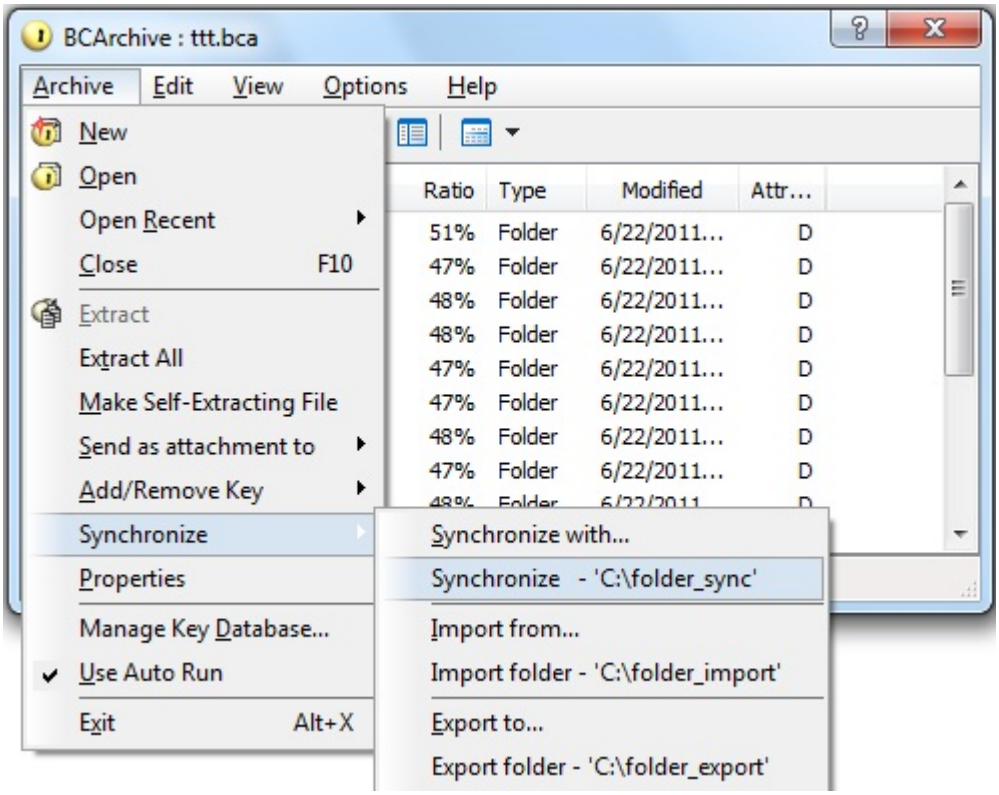
If you want to use existing archive, enter file name of the archive in the ***Archive name*** edit box or browse the archive by clicking
`[...]`.

See also:

Automatic opening document stored inside archive
Searching files inside archive

Synchronize, Import, Export functions

BCArchive provides a set of functions to synchronize the contents of existing archive with a non-encrypted folder: ***Synchronize***, ***Import from ...*** and ***Export to...***
They are available through ***Archive-->Synchronize*** menu:

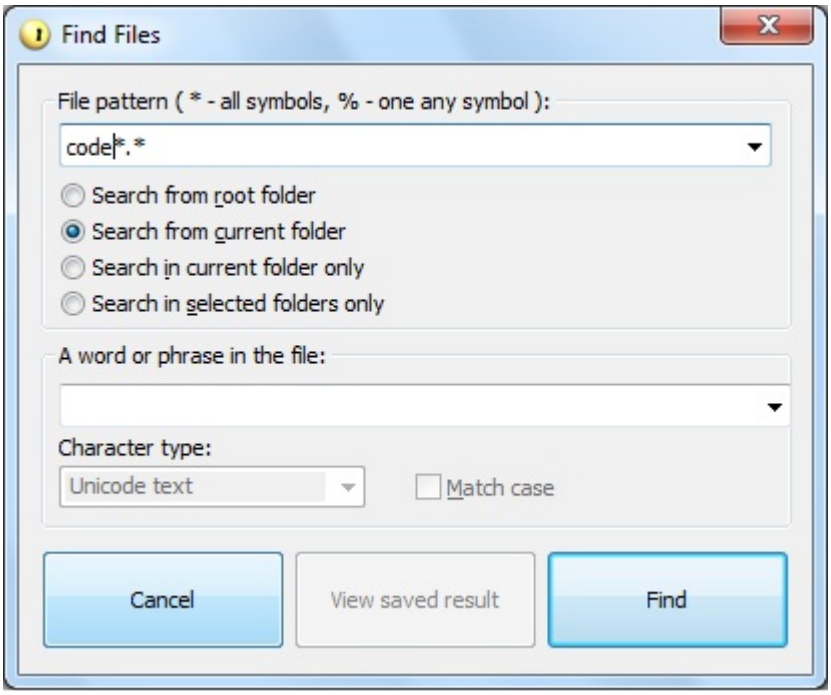


These functions work in different ways:

- ***Synchronize*** function compares the contents of the archive and the folder and keeps the latest versions of all files. It works silently and asks the user no questions. It adds files/folders to keep both locations equal. It does not delete any files or folders.
- ***Import from...*** function modifies the archive to keep it equal to the specified folder. If a file in archive is newer than the file in the folder, it asks the user whether to overwrite the file or not. If a file exists in the archive, but does not exist in the folder, it asks whether to delete the file in archive or not.
- ***Export to...*** function modifies the folder to keep it equal to the archive. If a file in folder is newer than the file in the archive, it asks the user whether to overwrite the file or not. If a file exists in the folder, but does not exist in the archive, it asks whether to delete the file in folder or not.

Searching files inside archive

As soon as you have opened encrypted archive, you can run the ***Find File*** command from the ***Edit*** menu in the main BCArchive window. The following dialog will appear:



Enter a complete file name of the document you want to find or its partial name in the ***File pattern*** edit box. Use * symbol for the sequence of letters that can consist of any number of letters in the file name or % symbol for exactly one letter that may have any possible value.

You can choose the following folders where BCArchive will look for the specified document:

- All folders inside encrypted archive by setting the ***Search from root folder*** option;
- Current folder and all subfolders by using the ***Search from current folder*** option;
- Look for the document in the current folder only by using the ***Search in current folder only*** option;
- Select some folders before running the Find File command, then run the command and choose the ***Search in selected folders only*** option to find the document in the selected folders.

If you want to find document storing specific word or phrase, enter the text in the ***A word or phrase in the file*** edit box. Please note that the document may store the text in different formats, hence, you should choose an appropriate format in the ***Character type*** list: Unicode text (format, which uses 2 bytes per letter), DOS text (format, which uses 1 byte per letter) or, if the file is in a binary format, Hexadecimal option.

Click [**Find**] to start the searching process or **Cancel** to terminate the process.
BCArchive allows users to save results of searching process.
If you click the [**View saved result**] , BCArchive will show you the results.

Automatic opening document stored inside archive

The BCArchive program stores documents inside its encrypted archive in encrypted form. If the user wants to edit the document, he/she should extract it from the archive, edit the document and then place it to the archive again. BCArchive allows complete automation of the process. The following ways are possible:

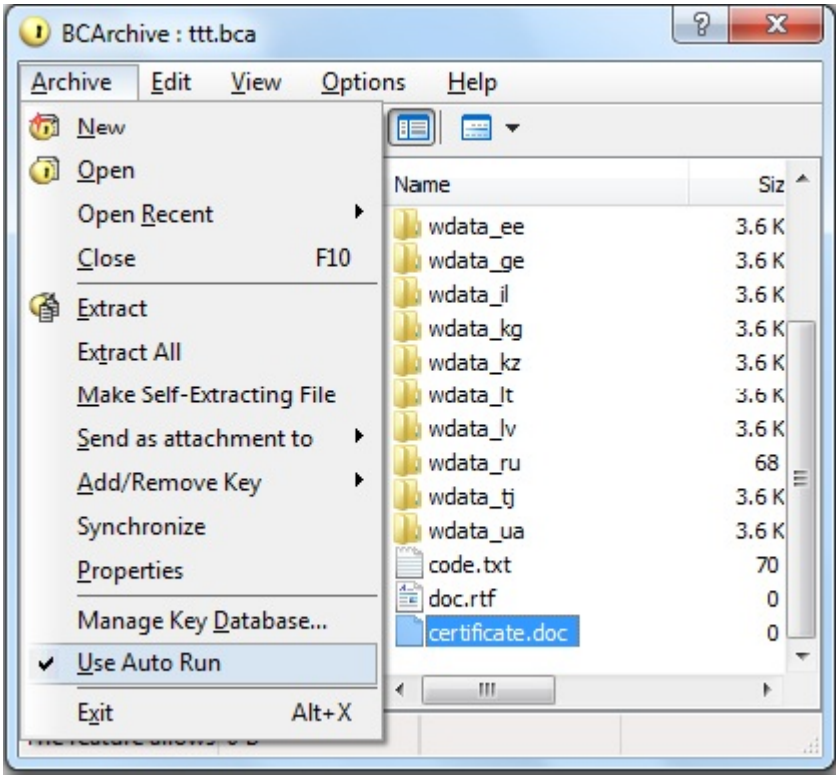
1. The user opens the archive file and gets main BCArchive window, where he/she can browse the document to edit. The user double-clicks on the document by mouse and BCArchive makes all the needed actions automatically:

   - Extracts the document to temporary directory
   - Runs an appropriate program to edit the document
   - Makes the program open the document
   - Will wait while the user is working on the document
   - As soon as the user closes the file, BCArchive will overwrite it in the archive and securely delete (wipe) temporary copy of the document

   The procedure will look for the user like the document was simply opened from the archive and then saved directly to the archive.

2. It is also possible that the user always works with some concrete document inside the archive, for example, with Microsoft Word document Certificate.doc, stored inside the Documents folder inside the archive. In this case the user can automate browsing of the Certificate.doc file inside the archive and automate the process of running Microsoft Word just after opening the archive. The functionality is close to standard **Autorun** function of Windows: when the user inserts CD disk with music files to CD-ROM device, Windows can automatically run Media Player to play the music. BCArchive behaviour is the same: the user enters password for his/her archive and Microsoft Word runs automatically and opens the Certificate.doc document.

To make BCArchive work in **Autorun** mode, open the archive and browse the Certificate.doc document as it shown on the following picture.



After selecting the Certificate.doc file, set the ***Use Auto Run*** option in the ***Archive*** menu. BCArchve automatically creates new **Autorun.inf** file in root directory of the archive and shows it. That's all, then, when the user will open the archive, BCArchive will run Microsoft word and it will open Certificate.doc automatically. Note, if you used **Autorun** feature before, please check off the ***Use Auto Run*** option and remove old **Autorun.inf** file in root directory of the archive. By default, BCArchive will close the archive as soon as the user finishes editing the document. The user can change the default behavior of the program. All the **Autorun** instructions BCArchive uses are stored in the root of the archive. In our case contents of the AutoRun.inf file looks like:

```
[autorun]
open = "Documents\Certificate.doc"
ExitOnClose = true
```

If the user wants BCArchive to keep the archive opened after the user closes Certificate.doc document, the last string in the AutoRun.inf file should be the following:
ExitOnClose =false

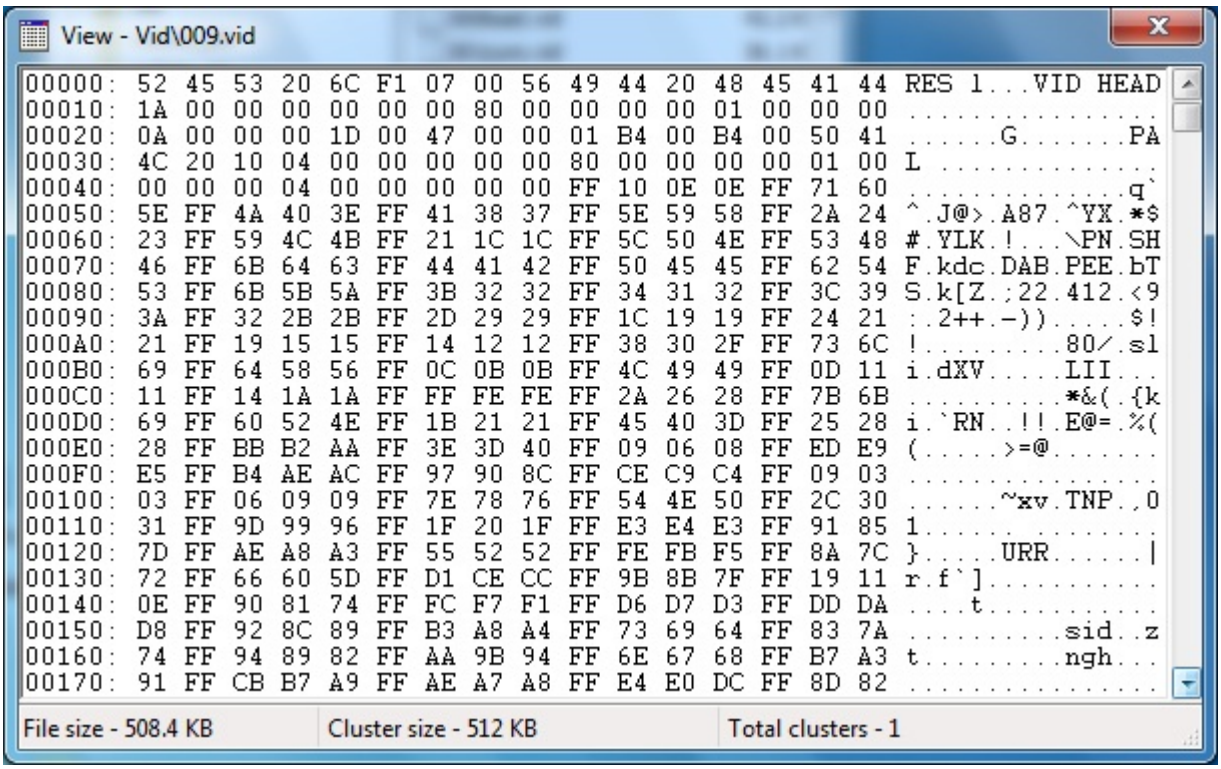
To turn off the **Autorun** functionality, run BCArchive and reset ***Use Auto Run*** option just before opening the archive.

View file inside archive in binary form

As it is mentioned in the Automatic opening document stored inside archive, BCArchive allows users to open documents stored inside encrypted archive so that the following steps are performed automatically:

- Extracting the document to temporary directory
- Running an appropriate program to edit the document
- Making the program opening the document
- Waiting while the user works with the document
- As soon as the user closes the file, BCArchive overwrites it in the archive and securely deletes (wipe) temporary copy of the document

If a user wants just to take a quick look at some file stored inside the archive, he/she can avoid the procedure described above. BCArchive has an embedded **binary viewer**, which allows users to view contents of any file stored inside archive. To run the viewer, browse the file inside archive, press right mouse button and run the ***Hex view*** command from appeared context menu. The following window demonstrates how text document looks in the BCArchive binary viewer.



Add Public Key/Password to existing archive file

To get access to the data stored inside encrypted BCArchive file (i.e. open the archive), you must enter the appropriate password for the archive. Hence, if you want other person to be able to open the archive too, you have to reveal the password to the person . In many cases it is not very good practice: for example, if you have other encrypted archives encrypted with the same password. To overcome the problem you can add new password for your encrypted archive. There are two kinds of passwords you can add:

1. **Additional password for the archive -** After opening archive run the ***Archive -> Add/Remove Key -> Add Password*** command and enter additional password for the archive.

2. **Public key of another person** - The first way of adding password is very simple, but sometimes it is difficult to tell or send the password to another person in a secure way. Besides, your correspondent may receive many archives from many people, and it will be difficult for him/her to remember a lot of passwords for every encrypted archive.In this case **public/secret key technology** can be used. Your correspondent should have his/her own public/private **key pair** generated. The public key of the user is not a secret and it is available for everyone. As for the corresponding secret key, the user stores it in a very secure place. If you encrypt some data with a public key, the data can be decrypted only by the person who knows the corresponding secret key.
Hence, if you encrypt BCArchive file with public key of another person, only that person will be able to decrypt the data. To make it possible, you should add public key of the person to existing archive file by running the ***Archive -> Add/Remove Key -> Add Public Key User*** command.

You can find the public key of another user in two places:

1. **In Local Public Key Database** - If you have received the public key from the user earlier, or found it in Internet, you could place it to the **Local Public Key Database**. If so, run the ***Add/Remove Key -> Add Public Key User*** command from the ***Archive*** menu and add the corresponding key.

2. **In Internet** - BCArchive allows searching of the user's public key in Internet and adding the key to the archive you have created. Since BCArchive supports PKCS #12 and X.509 standards, you can use existing public key servers to download and use the keys of other people.If you run the ***Archive -> Add/Remove Key -> Add Public Key User*** command, BCArchive will show the **BC Key Manager dialog**. In the ***Key*** menu of the dialog run the ***Add Existing Public Key -> Search in The Internet*** command.
Dialog window appears where you should select some predefined Web server where public keys of thousands of people are stored. You may enter name of the person or his/her e-mail address or other information and run searching process of the public key of the person. As soon as the program finds the key, you can save it in the **Local Public Key Database** for future use of the public key.

See also:

Local Public Key Database and Key Management

Send archive file as an e-mail attachment

You can send some archive file as an e-mail attachment directly from the BCArchive main window. To send the archive file, open the archive and run the ***Send as attachment*** command from the ***Archive*** menu in the BCArchive main window.
Since the archive file can be encrypted by password only or by password and public key of another user, or just by some public key, there are two ways to send the archive file:

1. You send the archive file without adding any passwords or public keys, just as it is at the moment when you have opened the archive. In this case run the ***Send as attachment to -> Ordinary User*** command. BCArchive will run your default e-mail program and prepare e-mail with empty recipient and attached archive file. All that you should do is to enter e-mail address of the recipient and probably add some comments to the e-mail.

2. You have a public key of some user in your **Local Public Key Database** and you want to send him/her the archive file so that the archive were encrypted by the user's public key. In this case run the ***Send as attachment to -> User with Public Key*** command. BCArchive will show you list of the users in the Local Public Key Database and you will select one of them from the list. After that BCArchive runs your default e-mail program and prepares e-mail with attached archive file addressed to the user you have selected. If you wish, you can add some text to the e-mail and send it.It is also possible to run the same command from **My Computer** window. Windows popup menu for files/folder may contain the command ***Encrypt by Public Key and Send***, if you made it available (see BCArchive Extension for Windows Shell).

See also:

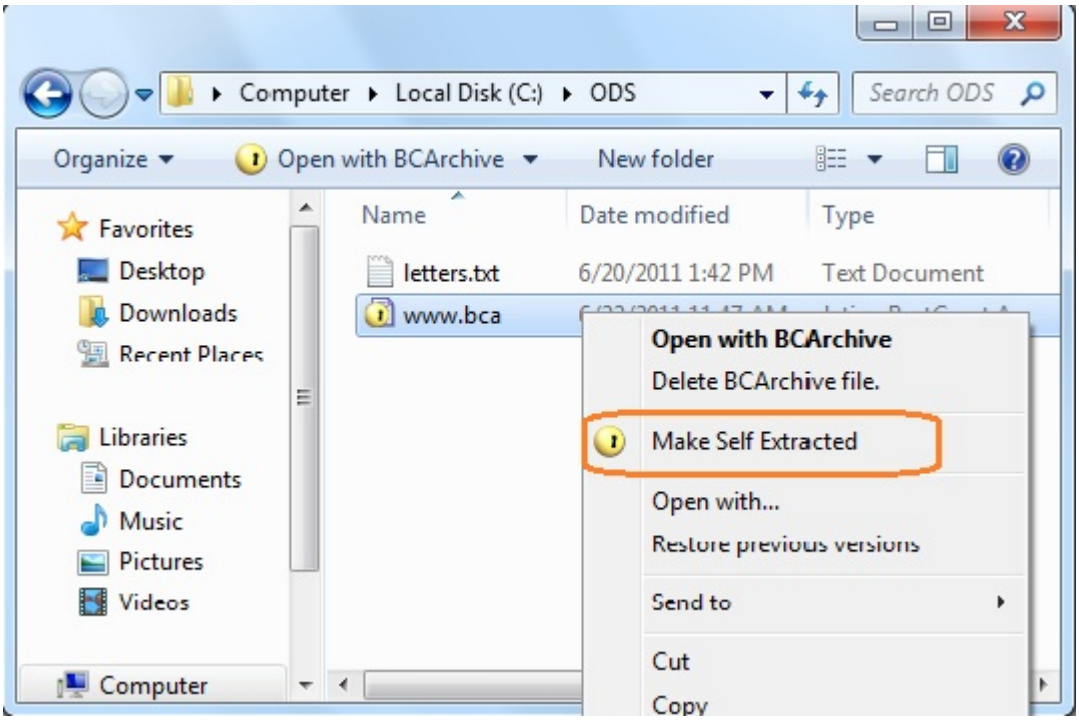
Local Public Key Database and Key Management

Make Self-Extracted archive file

To use all the functions available in the BCArchive software you should have the program installed on your computer. From the other hand, you may wish to create encrypted/compressed archive file for backup purposes or for sending it to another user.

It is possible that the person who will try to extract data from the archive has no the BCArchive program installed. In this case you can transform the archive file to **self-extracted executable file** (i.e. program file that contains executable code and encrypted/compressed data). As soon as some other person runs the self-extracted program you have created, the program will ask him/her to enter an appropriate password and extract the data in decrypted form on the user's computer.

To create self-extracted archive you should open existing archive or create a new one and place all the data you want to the new archive. Then run the ***Make Self-Extracting File*** command from the ***Archive*** menu. The same command exists in right-click menu for **BCA** files in My Computer window:



The BCArchive program will create an executable file, corresponding to your archive file in the same directory where the archive file is stored. For example, if your archive has C:\Archive.bca name, BCArchive will create corresponding C:\Archive.exe self-extracted archive file.

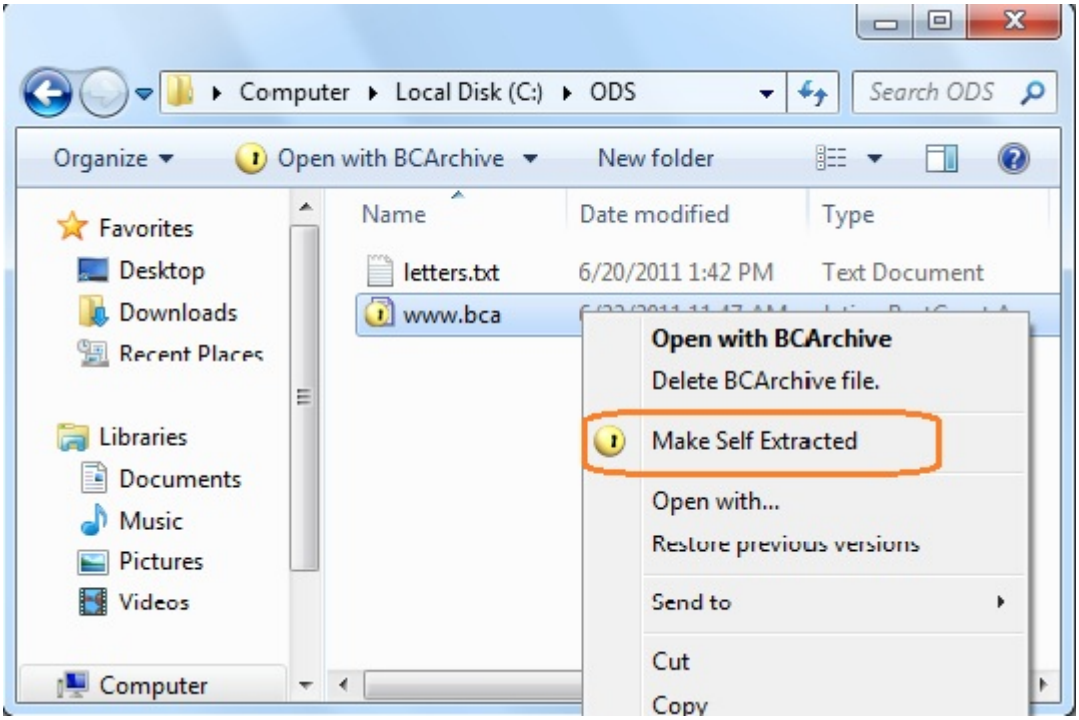
See also:

Create New Archive File
Open existing Archive file

BCArchive Extension for Windows Shell

You may choose the commands you want to be available through right-click menu for your files and folders in Windows Explorer. To make the choice, you should run the command ***Windows Shell Commands*** from BCArchive ***Options*** menu. The following window will appear where you can turn the commands on/off:

For BCA archives, ***Make Self Extracted*** command allows to create self-extracting executable file that can be opened without BCArchive program installed:



To create a new BCArchive archive, you should run the ***New*** command from Explorer using a standard menu or pop-up menu and select ***New -> Jetico BestCrypt Archive***

See also:

[Create New Archive File]
[Send archive file as an e-mail attachment]
[Make Self-Extracted archive file]

How to run BCArchive from the command-line prompt

The following BCArchive commands can be run from command-line prompt:

1. Open existing archive.

   >BCArchive Open archive_name

2. Add file/folder to archive

   >BCArchive Add archive_name file/folder/@*listFile name [-PKID publicKeyID]

   where: @* - prefix before list file name,
   * - means delete the list file when process finishes
   -PKID publicKeyID - see 'Options' below.

3. Extract specified files

   >BCArchive Extract archive_name destination_folder_name [file pattern list ...]

   Sample: >BCArchive Extract "C:\My archive.bca" C:\tmp *.txt

   (All txt files will be extracted from "C:\My archive.bca" to C:\tmp folder.)
   Note, file name pattern may contain * or % symbols
   % means any single symbol is allowed
   * means any symbols is allowed
   Sample: "qwerty" is compatible with "q%e*y*".

4. Encrypt by public key and send.

   Chooses public key user (or uses the one defined by PKID), encrypts files by public key and sends them to the owner of the key.
   >BCArchive Send file/folder_name [-PKID publicKeyID]

   -PKID publicKeyID - see 'Options' below.

5. Make self extracted file .EXE from existing .BCA archive

   >BCArchive MakeSelfExtracted archive_name [new exe path] [-LF logFileName]

6. Show local database public key identifiers.

   >BCArchive ShowPKID

Options:

-PKID publicKeyID

If public key identifier (-PKID) is specified, it creates new archive encrypted by the public key. Note, the public key must exist in local database. Use ***Manage Key Database*** command in Archive menu in BCArchive.
Use ***ShowPKID*** command to view public keys existing in your database.
You may define several -PKID parameters to encrypt by several public keys and send the archive to all users.

-LF logFileName
Writes messages to the log file in batch mode.

Dynamic/manual compacting of archive file

The BCArchive file consists of three parts: **header**, **key block** and **data block**.

Header of the archive file contains information about size and location of key and data blocks.

Key block is a sequence of encrypted keys stored according to the ASN.1 format. The keys are encrypted by another key, generated from password or by public key according to the settings the user has chosen when he/she created the archive file.
You can add new or remove existing password to the archive. As well you can add public key of some user to the archive file – it means that the user will be able to decrypt the archive with his/her secret key. There is no limitations for number of keys in key block, hence you can add as many passwords/public keys to the archive as you wish.

Data block consists of number of clusters and looks like well known FAT file system. Every cluster is compressed and encrypted with the main archive key and symmetric algorithm you have chosen when created the archive. If you update data inside the archive and some cluster becomes modified, BCArchive does not write it immediately to the archive file. Instead, BCArchive creates hidden subfolder in the same directory where archive file is stored and saves modified cluster as a file in the subfolder. Special compacting function removes old cluster from the archive file and then appends the modified cluster to the archive.

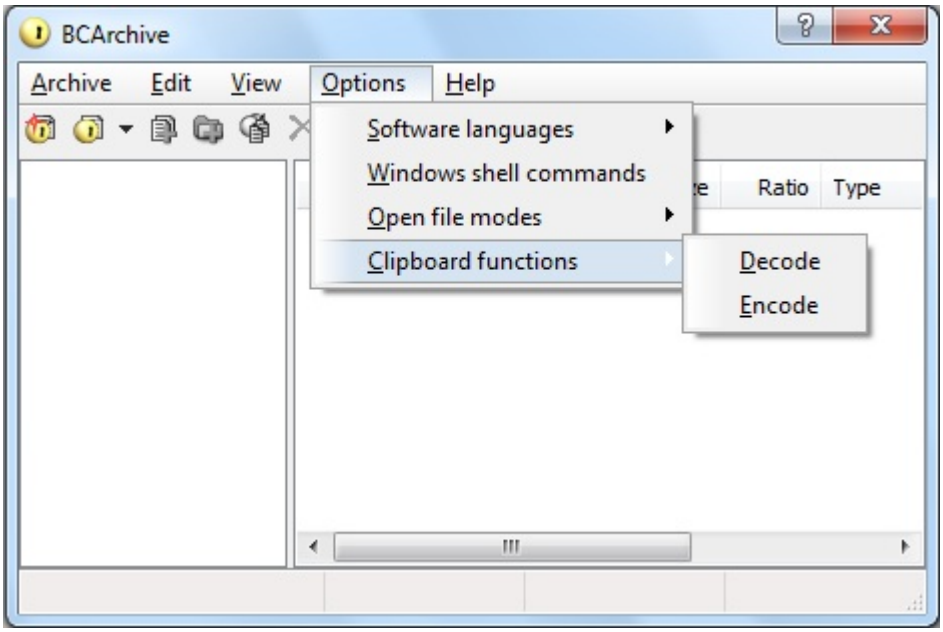
Users can run the compacting function manually by running the ***Compact -> Compact Now*** command from the ***Edit*** menu in the main BCArchive window. Besides, user can set the ***Dynamic Compact*** option. In that case, BCArchive will run compacting automatically. If you are going to make significant modifications in a large archive file, you may turn off the ***Dynamic Compact*** option and save computer resources required for dynamic compact of large compressed data.

Text Encoder

BCArchive includes **BCTextEncoder** utility intended for fast encoding and decoding text data. Plain text data are compressed, encrypted and converted to text format. The result of such conversion may be copied to the clipboard or saved as a text file.

BCTextEncoder uses **public key encryption** methods as well as password based encryption. It uses strong and approved symmetric and public key algorithms for data encryption.

To open **BCTextEncoder** window - run the command ***Encode*** or ***Decode*** from ***Options-->Clipboard functions*** menu:



See also:

BCTextEncoder and its Assistant
Quick Start
How to use BCTextEncoder
BCTextEncoder Commands and Options
Encoded Data Format
BCTextEncoder Assistant Options

Local Public Key Database and Key Management

- Local Public Key Database and Key Management
- Create or import Secret/Public Key Pair
- Send your public key to another person
- Add Public Key to Local Public Key Database
- Backup/Restore Local Key Database

Local Public Key Database and Key Management

A lot of people around the world have their **secret(private)** and **public** keys. They make their public keys opened for everyone and keep corresponding private keys in a secure place. Public key can be used by anyone to encrypt data, but only an owner of corresponding private key can decrypt the data.

For example, you decide to send an encrypted container to your friend John. John may have his public key created earlier and stored on a **Public Key Server** in Internet. John may also send you his public key attached to e-mail. As soon as you get John's public key, you can encrypt the container with this key and send it to John. After receiving the container John will be able to access data with his **secret** key.

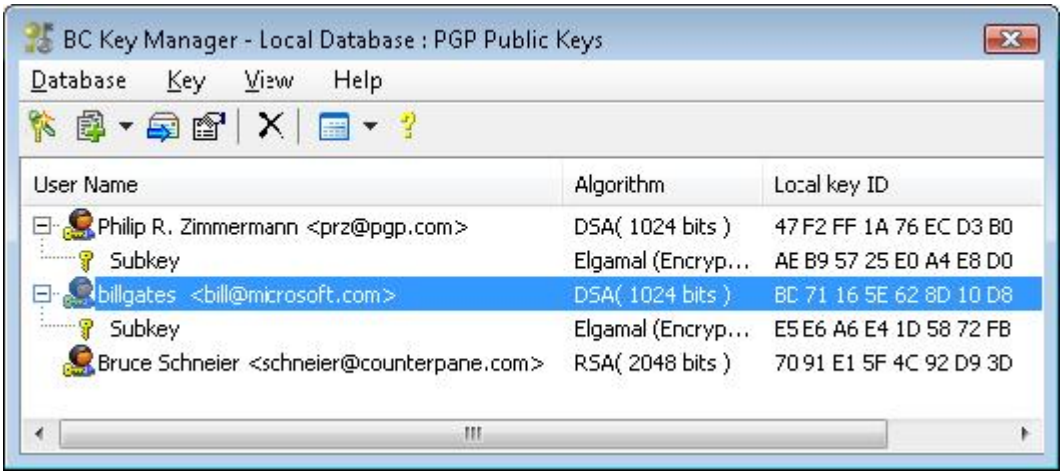
BCArchive includes the **BC Key Manager** utility to manage your own public/secret key pair as well as public keys you have received from other people. You can run **Key Manager** utility from BCArchive Program Folder or using ***Manage Key Database*** command in ***Archive*** menu of BCArchive main window.

See also:

Create or import Secret/Public Key Pair
Add Public Key to Local Public Key Database
Backup/Restore Local Key Database

Create or import Secret/Public Key Pair

BC Key Manager utility allows you to create your own public/secret key pair. It may be useful if you decide to send your public key to other people so that they will be able to encrypt some information for you using your public key. As soon as you receive the information encrypted by your public key, you can decrypt it using your private key. Any other person, who has not your secret key and does not know the password for it, will not be able to decrypt the information.
It is also possible that you have already had the public/secret key pair generated earlier, for example, with a help of the **Pretty Good Privacy (PGP)** software. Since **BC Key Manager** understands a number of formats, you can import the key pair from the file created by other software.
The main window of BC Key Manager looks like:

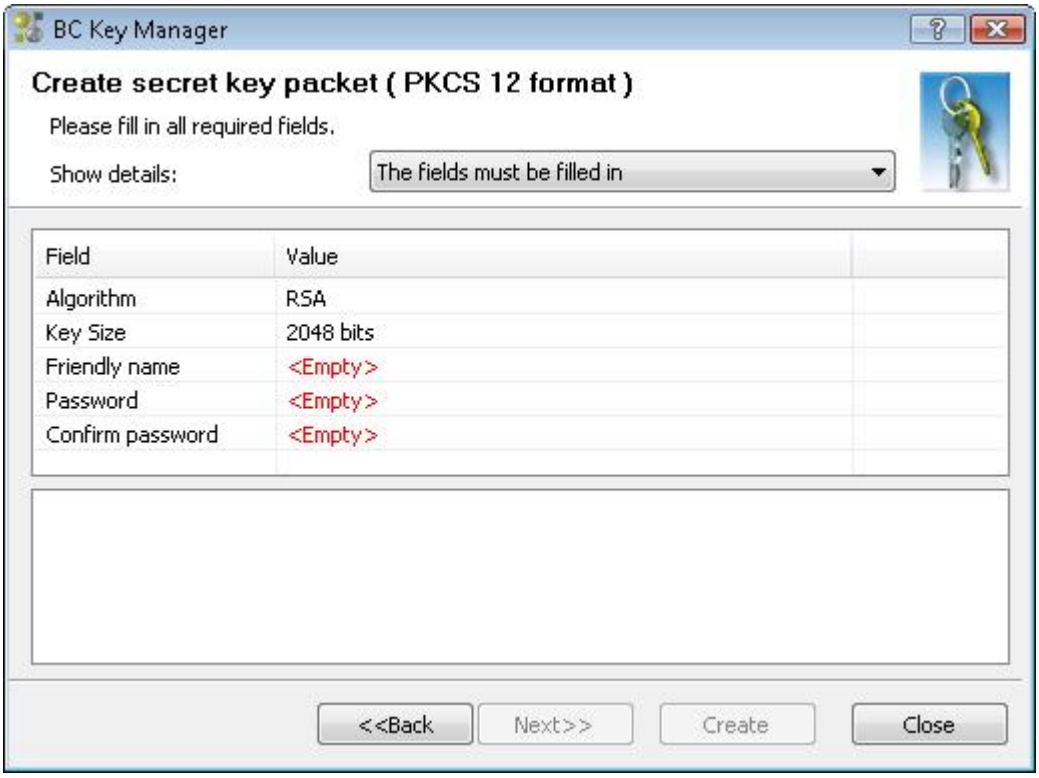


To create or import your public/secret key pair, run the ***Generate New Public/Secret Pair*** command from the ***Key*** menu in the BC Key Manager utility. The following window will appear:

In the BC Key Manager window select ***Generate new private key*** if you wish to create new key pair or the ***Import existing private key*** option if you want to use existing key pair you have created earlier using BC Key Manager or some other program.
When BC Key Manager finishes the key pair generating process, it can do all or some of the selected actions depending on the options you choose in the first BC Key Manager window:

- ***Create file with your secret key in PKCS#12 format*** -The key will be protected by password and you can save the file in any place you wish for later use or backup purposes.
- ***Create file with your public key in X.509 format*** - As soon as you create such file, you can send it to other people so that they will be able to send encrypted information to you (as it was mentioned in the beginning of the chapter).
- If you create new key pair or import existing one, Key Manager will save it in its **Local Key Database** if the ***Local Public/Secret Key Database*** option is set.
- The key pair can also be saved in a separate file in internal BC Key Manager format for backup purposes if you select the ***Files chosen later*** option.

After selecting all the option you want, click **[Next>>]** in the BC Key Manager window. The following window will appear:



In the **Create Secret packet** window you can choose the settings for creation a secret key for you. The program shows the field you must fill in drawn by red color and it means that the user should enter some strings into the fields:

- ***Friendly name*** - It is the information that will be used to identify your public key among the keys of other people. For example, if you enter the **'John Smith - JohnSmith@my_email.com'** string, your friend can easily find your public key in the list of public keys of other people he/she has on his/her computer.
- ***Password and Confirm password*** - Enter a password for your secret key into the ***Password*** field and enter the same password again into the ***Confirm password*** field again to verify that you have typed a correct password. The

password will protect your secret key so that if even someone steals a file with your secret key, the intruder will not be able to use the file to decrypt information, encrypted by your public key.

It is also recommended to pay attention to the ***Key Size*** field in the Create secret packet dialog window. Public/secret key algorithm can be used with different key sizes and it is recommended to use the algorithm with key size equal to at least 2048 bits.

If you click [**Next>>**] , the **Create Certificate** window will appear. **Certificate**(as it is understood in the context of the public/secret key encryption technology) is the file with text information about your public key. Since you are going to send the public key to other people for using it on other computers with probably other software, information about your public key should be sent together with other technical information, like name of the encryption and secure hash algorithms, key size, format of the file where the key is stored and other.
The **Create Certificate** window shows you the information, which will be stored in the certificate file created for your public key. Please note that you should enter the information required in the ***Subject*** field. When you double-click on the field and start to edit it, the **Get certificate subject** dialog window will appear.

The dialog window contains a number of fields you may fill in to identify your public key among thousands of public keys created by other people. Please note that entering such information is specific for the BC Key Manager software only. It is a common practice for software that uses public/secret key technology and conforming the X.509 standard. You can fill in not all the fields in the Get certificate subject dialog window, but BC Key Manager requires the information be entered to at least one field of the window.

After entering the information click [**OK**] in the Get certificate subject window, and then [**Create**] in the Create Certificate window. After that BC Key Manager will generate a public/secret key pair for you and save it to your **Local Public/Secret Key Database**.

Send your public key to another person

If you want to receive encrypted data from another person, the data have to be encrypted by your public key, so you should send the public key to the person. You can use BCArchive to simplify the process of sending the public key by e-mail in the following way.

Run **BC Key Manager** utility from BCArchive program group. Run the ***Send Public Key to E-Mail Recipient*** command from the ***Key*** menu. BC Key Manager will show you a list of all public keys in your Local Public Key Database. Select your public key from the list and click [**OK**].

BCArchive will run your default e-mail program and prepare e-mail with empty recipient and encoded public key, written in the PKCS#12 format (so called x.509 certificate). All that you have to do is to enter e-mail address of the recipient and send the e-mail.
What your recipient should to do when he/she receives public key attached to e-mail?

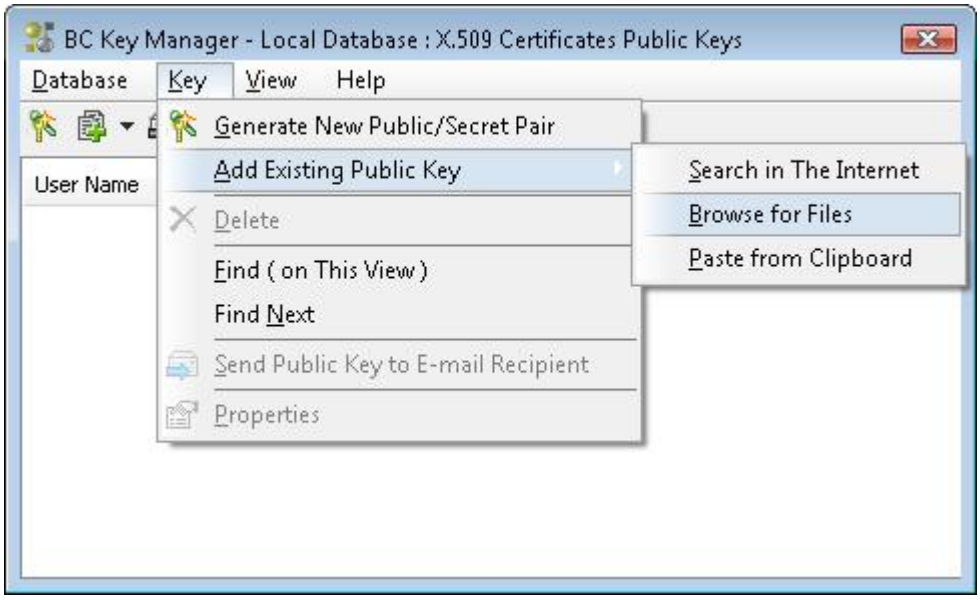
The recipient should add the attached public key to the BCArchive **Local Key Database** on his/her computer in the following way:

- Save the attached public key certificate file to some folder.
- Run the **BC Key Manager** utility from BCArchive program group.
- Run the ***Key -> Add Existing Public Key -> Browse For Files*** command.
- BCArchive **Choose Public Key** dialog window will appear.
- Browse the folder where you have saved the certificate file with the public key, select the file and click [**Get It**].

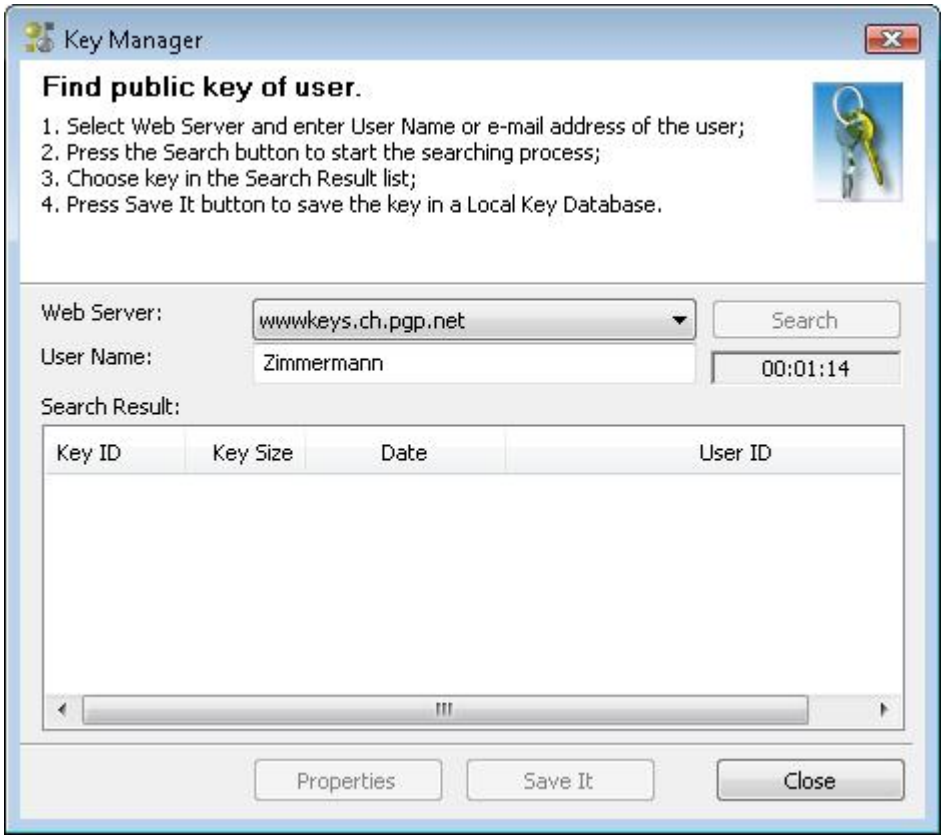
Add Public Key to Local Public Key Database

You can send compressed archive to another person encrypted by public key of the person. If you are going to send encrypted information to a person continuously, you should save the public key in your **Local Public Key Database**. To add a public key to the database use one of the following ways:

1. Load the key from file, where the public key is stored. The person, you are going to correspond with, can send you the file with his/her public key. Key Manager supports **PKCS 12** format as well as **Key Ring** format of the **Pretty Good Privacy (PGP)** software. To save public key from the file in one of the formats, run the Key Manager utility from BestCrypt Program Folder or from ***Utilities*** menu of BestCrypt Control Panel. Then run ***Add Existing Public Key -> Browse for File*** command from the ***Key*** menu and browse the file where the public key is stored. The following picture illustrates the method:



2. Your correspondent may have his/her public key stored on some **Public Key Server(s)** in Internet. In this case you can run the process of searching the public key in Internet. If you run the ***Add Existing Public Key -> Search in the Internet*** command from the ***Key*** menu, the following window will appear:



Select one of the Web servers where the public key may be stored in the **Web server** edit box, enter name of the person or his/her e-mail address in the **User Name** edit box and click [**Search**]. BC Key Manager will start to look for the user's public key and if there are a number of people whose names are the same as the name of your friend, BC Key Manager will display all of them in the Search result list. Select the string from the list, corresponding to the person you are looking for and click [**Save It**] to save the public key in your **Local Key Database**.

See also:

Add Public Key/Password to existing archive file

Backup/Restore Local Key Database

Local Public Key Database saves your time, because when you add public key of other user to encrypted container, you do not need in accessing Internet to download the public key again. It is recommended to backup (or export) the database file regularly and save the file on a reliable storage medium. If in future you decide to change your computer or reinstall the software, you can restore (or import) the database from the backup copy.

To **save (export)** Local Key Database run **Public Key Manager** utility from BestCrypt Program Folder or from ***Utilities*** menu of BestCrypt Control Panel. In the main window of the program run the ***Export to File*** command from the ***Database*** menu. BC Key Manager will ask you to enter path and name for the file where you want to save your **Local Public Key Database**.

To **restore** Local Key Database from earlier saved (exported) database, run the ***Import from File*** command from the ***Database*** menu. The program will ask you to enter path and name for the database file. After that BC Key Manager will copy the database to the folder where the software is installed and start to use the database you have imported.

You can also **change the location** of the Local Key Database by running the ***Choose Store Folder*** command from the ***Database*** menu of the BC Key Manager utility.

BCTextEncoder

- BCTextEncoder and its Assistant
- Quick Start
- How to use BCTextEncoder
- BCTextEncoder Commands and Options
- Encoded Data Format
- BCTextEncoder Assistant Options

BCTextEncoder and its Assistant

BCTextEncoder is intended for fast encoding and decoding text data. So, there must be an easy way to access the program window as soon as the need arises. From the other hand, it is not very good to keep the BCTextEncoder window always opened. To resolve the issue, a special process is used - **BCTextEncoder Assistant**. The process is always running in the background and monitoring keystrokes on your keyboard to detect pressing the **Hot Key** combination to open BCTextEncoder window.

Additionally, **BCTextEncoder Assistant** is able to show or hide the systray icon and clear all BCTextEncoder settings and modified registry entries. See more information in BCTextEncoder Assistant Options chapter.

See also:

How to use BCTextEncoder
BCTextEncoder Commands and Options
Encoded Data Format
BCTextEncoder Assistant Options
Local Public Key Database and Key Management

Quick Start

Let's assume you were writing an e-mail using your accustomed e-mail application and you decided to encrypt a part of the message. Provided that all needed options have been already set, you will have to make only four simple steps:

1. Select the secret data and put it to clipboard with Ctrl-X;
2. Press a predefined hot key to open BCTextEncoder and click [**Encode**];
3. Enter password;
4. Return back to your e-mail and paste the encrypted data from clipboard with Ctrl-V;

The main advantage of BCTextEncoder is support of **public key encryption**. If you have a public key of your recipient, then you may make slightly different steps:

1. Select the secret data and put it to clipboard with Ctrl-X;
2. Press a predefined hot key to open BCTextEncoder
3. Use ***Encode by*** list box, choose the public key of your recipient and click [**Encode**];
4. Return back to your e-mail and paste the encrypted data from clipboard with Ctrl-V;

Alternatively, you can write the message directly in BCTextEncoder window or load the text from an existing text file. After encoding the text, it may be sent to the recipient immediately, if corresponding option is enabled. If you encrypted the text with a public key, the e-mail will be read automatically from the key, so you do not have to do anything except confirmation of sending.

See also:

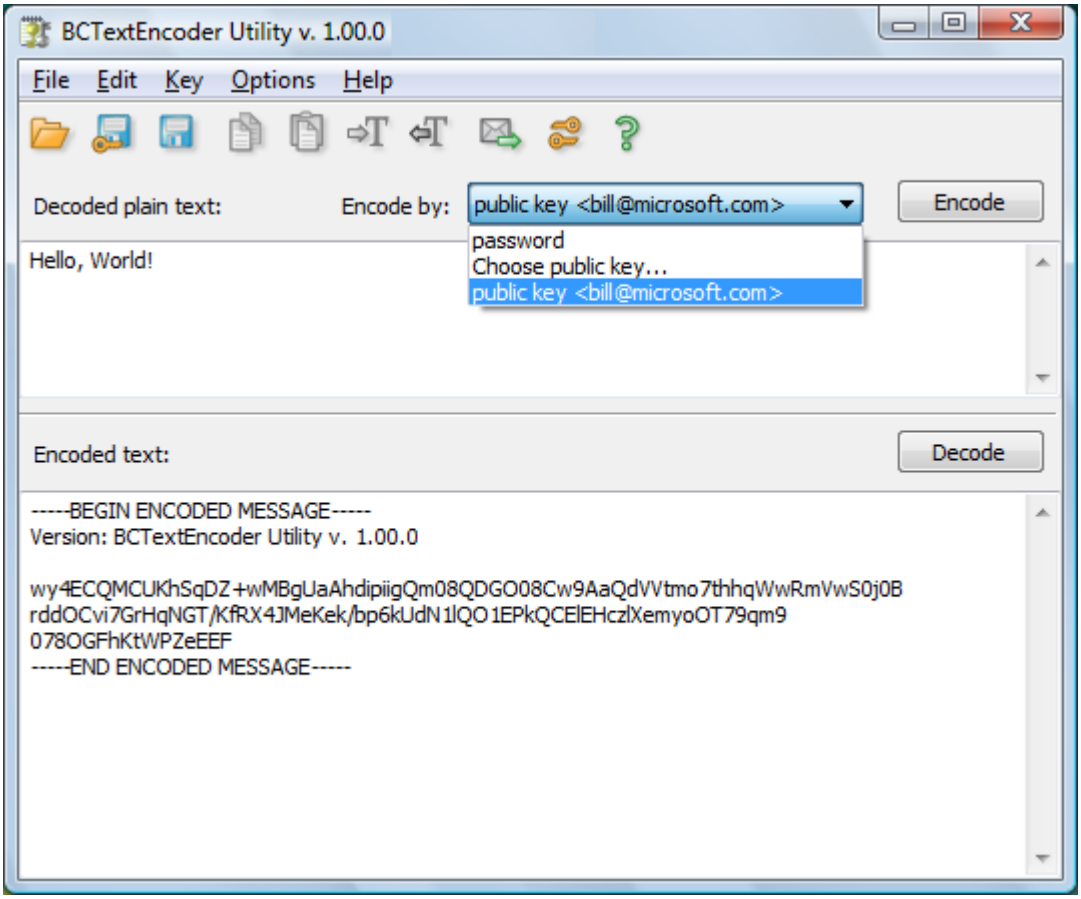
How to use BCTextEncoder
BCTextEncoder Commands and Options
BCTextEncoder Assistant Options
Local Public Key Database and Key Management

How to use BCTextEncoder

Opening BCTextEncoder window

To run BCTextEncoder for the first time, you should run BCTextEncoder.exe file which was downloaded from our site. **BCTextEncoder** main window will appear and **BCTextEncoder Assistant** process will start. When BCTextEncoder Assistant is running, you can open the main BCTextEncoder window using the systray icon or predefined Hot Key. See more information in BCTextEncoder Assistant Options chapter.

BCTextEncoder window consists of two panes - **Plain Text** pane and **Encoded Text** pane.



Text Encoding

1. Put the text into **Plain Text** pane. This can be done by three ways:

    - Select the text in your e-mail or text editor, copy it to clipboard and open BCTextEncoder. If the option ***Automatically decode encoded text*** is enabled, the contents of the clipboard will be placed to the window automatically. Otherwise, you have to paste it manually.
    - Type the text directly in **Plain Text** pane of BCTextEncoder window.
    - Open the text file with ***Open*** command from ***File*** menu.

2. Using ***Encode by*** box, choose a type of encryption. You may encode by password or by public key of other person. If you have no a public key, please see BestCrypt Key Manager section to know how to generate your own key pair and import an existing

public key. The Key Manager functions are available in ***Key*** menu of BCTextEncoder window.

3. Use **[Encode]** button to encode the text displayed in **Plain Text** pane. The encoded text in Encoded Text Format will be placed in **Encoded Text** pane.

Text Decoding

To decode a text, copy encoded text to clipboard and open BCTextEncoder. If the option ***Automatically decode encoded text*** is enabled, you will be asked for the password and the decoded text will be placed into **Plain Text** pane. Otherwise, you have to paste encoded text manually and use **[Decode]** button to decode the text.

See also:

BCTextEncoder Assistant Options

BCTextEncoder Commands and Options

Menu Items

- ***File***
  - ***Open*** - open a text file to encode or encoded text to decode
  - ***Save*** - save the current window contents to a file

- ***Edit***
  - Increase Indent - indent the original text with the symbol ">"
  - Decrease Indent - remove symbols ">"
  - Copy to Clipboard
  - Paste from Clipboard

- ***Key***
  - Generate New Public/Secret Key Pair
  - Choose public key for encoding
  - Manage Key Database - opens **Public Key Manager**

- ***Options***
  - Copy decoded text to clipboard after decoding
  - Copy encoded text to clipboard after encoding
  - Send encoded text by e-mail now
    Send encoded text automatically after encoding
  - Clear clipboard
  - Clear clipboard automatically on Exit
  - BCTextEncoder Assistant and Hot Key options

- ***Help***
  - About
  - Help index

Toolbar Buttons

 - read text from existing file;

 - save encoded text to file;

 - save decoded text to file;

 - paste clipboard text to currently focused window;

 - copy currently focused window text to clipboard;

 - increase indent;

 - decrease indent;

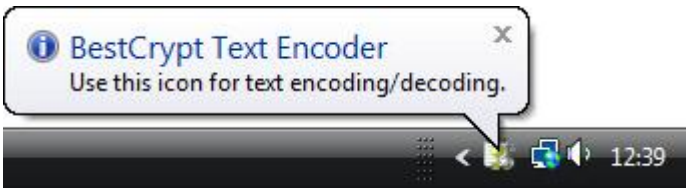
 - send encoded text by e-mail;

 - choose public key for encoding;

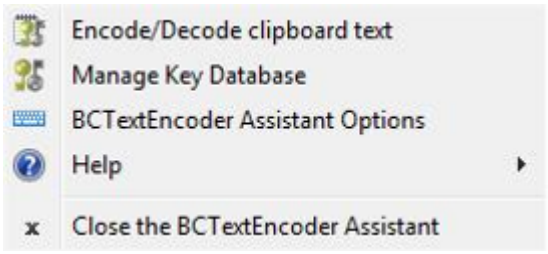
 - about BCTextEncoder;

Sytray Icon Menu.

When you start BCTextEncoder for the first time, **BCTextEncoder Assistant** starts and creates the icon in System Tray area and shows the balloon.



The Systray Icon has the following pop-up menu:

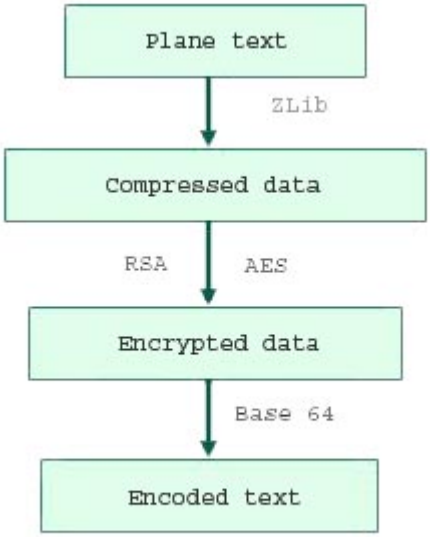


See also:

How to use BCTextEncoder
BCTextEncoder Assistant Options
Local Public Key Database and Key Management

Encoded Data Format

BCTextEncoder not only encrypts, but also compresses the data.
First, plain text data is compressed by **ZLIB** compression algorithm. The compressed data is encrypted by choosen public key or by password. At this step, BCTextEncoder utilizes **RSA** asymmetric algorithm for public key encryption and **AES(Rijndael)** algorithm with 256-bit key for password-based encryption. Finally, encrypted data is encoded by **BASE64** encoding algorithm to text format.
The picture illustrates the process of data transformation:



Example:
The 'Hello, Word!' plain text encoded by password 'password' looks like:

-----BEGIN ENCODED MESSAGE-----
Version: BC Text Encoder Utility v. 1.00.0 (beta)

wy4ECQMC6J8Np1DDfutgzFNqgHsDsam9CbC/QJ3pg8oV7nFbbtQrfygrLRoh/y/10j0B
2hHwpqOX5ACgP4tgt/D9RQmOQSON92mSSvoMVEnm9yq/hIO/XJ0lIi+VsWNpaBBs
mVvhD6VocCAzWJiz
-----END ENCODED MESSAGE-----

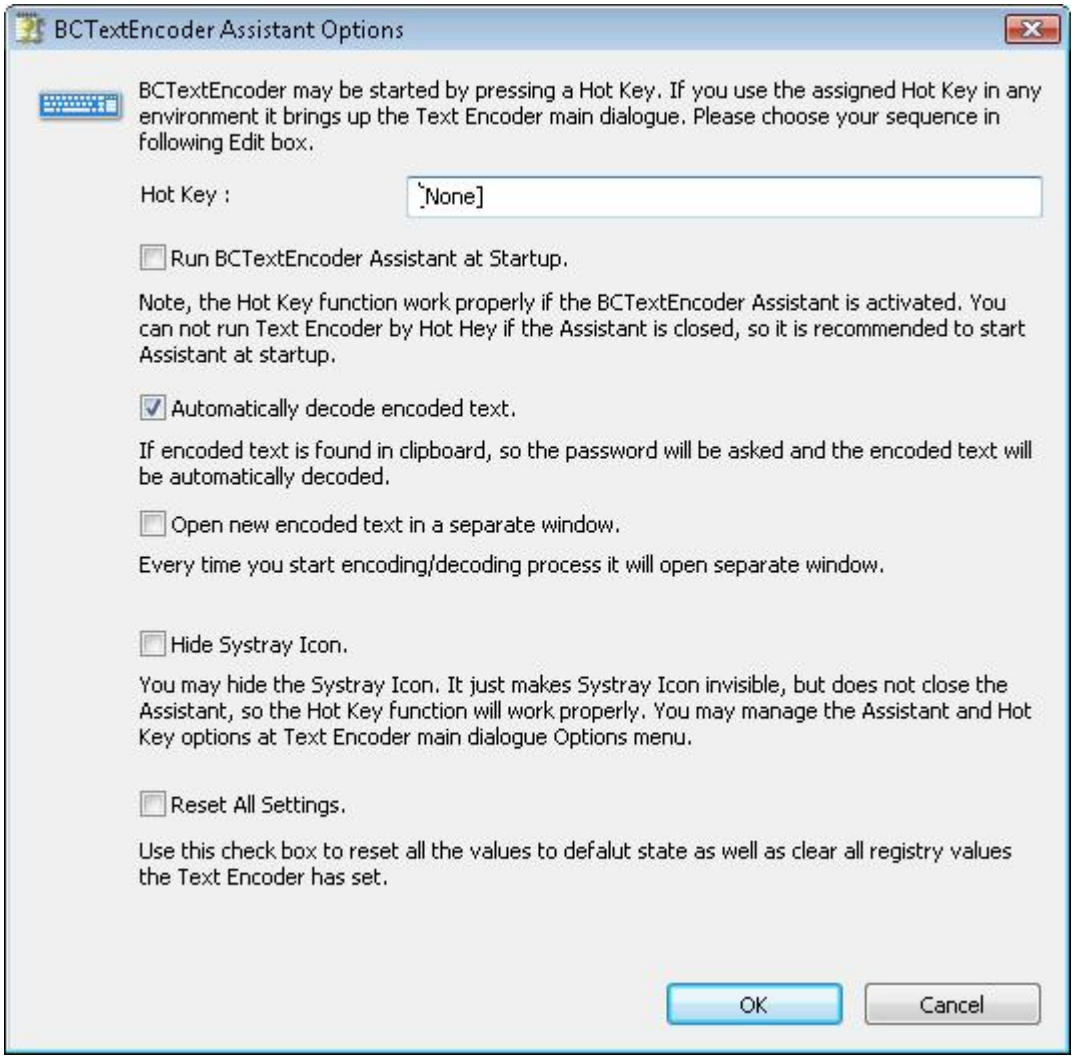
So resulting text contains only valid characters and may be sent by e-Mail or saved as a text file.

NOTE: BCTextEncoder is compatible with well known **PGP utility**. It means that it can decode messages encoded by PGP .

BCTextEncoder Assistant Options

BCTextEncoder Assistant Options dialog is used to change settings concerning BCTextEncoder Assistant activity. To open the dialog you should run the command ***BCTextEncoder Assistant Option***s in ***Options*** menu of BCTextEncoder window OR in **Systray Icon** pop-up menu.

The following window will appear:



Hot Key
You can start BTextEncoder main window by pressing the specified Hot Key from any environment.
Type your key sequence in ***Hot Key***: edit box, f.e. 'Ctrl+Alt+H'.
If you want to disable the Hot Key functionality, just delete all symbols with ***Backspace*** button.

Run BCTextEncoder Assistant at Startup
Note, you cannot run BCTextEncoder by Hot Hey if the Assistant is not running, so it is recommended to start the BCTextEncoder Assistant at system startup.

Automatically decode encoded text
The option allows to decrease amount of steps for decoding to minimum. If you see encoded text, you just copy it to clipboard and press the hot key you've previously assigned. BCTextEncoder automatically detects encoded text in clipboard, asks for the password and shows decoded text.

If encoding signatures are not found, the text is considered as plain text you are going to encode. So the text is placed to **Decoded plain text** window, you need to choose encoding options and click [**Encode**] to encode the text.
If you set options ***Copy encoded text to clipboard after encoding*** and ***Copy decoded text to clipboard after decoding*** you will minimize the operation steps even more.

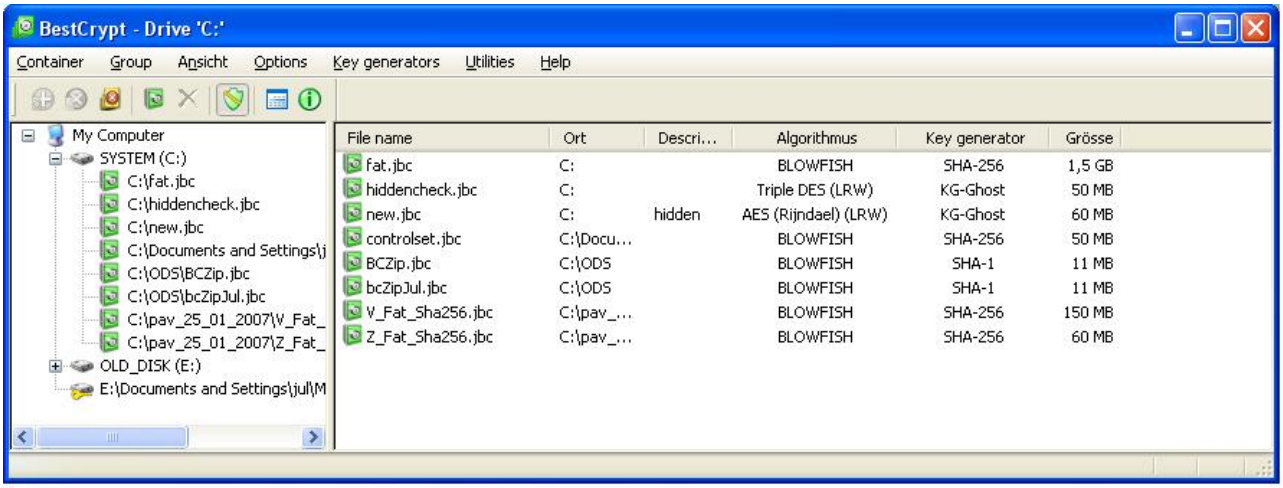
Open new encoded text in a separate window
Every time you start encoding/decoding process, it will open a separate window, so you can work with several documents simultaneously.

Hide Systray Icon
Since you assign the Hot Key, you may hide the **BCTextEncoder Systray Icon**. It just makes the icon invisible, but it does not kill the BCTextEncoder Assistant process, so the Hot Key function will work properly. In that case, you will get access to this dialog through ***Options*** menu of BCTextEncoder main window.

Reset All Settings
Please use this check box to return all settings to default state. The command also clears all the registry entries BCTextEncoder has set.

What is the BestCrypt Data Encryption System?

The BCArchive utility is a part of the BestCrypt software system.
BestCrypt Data Encryption System allows users to keep any form of data (files, letters, pictures, databases) in encrypted form on the hard disk, networks disks, removable disks, CD-ROM's and floppies. BestCrypt then lets user access it from any application.
Using BestCrypt you can create a container file (for example, you may create a 5Mbyte container file called LETTERS.jbc). Then you can mount this container as an additional logical drive: it will show up in Windows as an additional 5Mbyte virtual disk. When mounted, this logical drive looks and operates just like an ordinary disk drive: you can store your files on it. All files stored on the disk are automatically encrypted. Every read operation, which addresses the drive, causes decryption of the data, and every write operation causes encryption of data to be written. This approach is called **transparent encryption**. Using this system, your data is always stored in a safe encrypted form and appear decrypted only in the application you use to process them, and only while they are processed.
The following picture shows the **BestCrypt Control Panel**, used to perform all control operations (creating and mounting containers, setting BestCrypt options and so on):



BestCrypt uses encrypted logical disks technology to provide transparent encryption of your data. You only need to choose a drive letter and a password for your new BestCrypt logical drive. After password verification, access and use of encrypted data become transparent for any application.
To achieve maximum security BestCrypt utilizes the well-known encryption algorithms:

- Blowfish, in Cipher Block Chaining mode (256-bit key size);
- Twofish, in Cipher Block Chaining mode (256-bit key size);
- GOST 28147-89, in Cipher Feedback mode (256-bit key size);
- Rijndael, in Cipher Block Chaining mode (256-bit key size).

BestCrypt system has a number of additional utilities:

- Wiping utility (BCWipe)
- Container's Guard Utility
- Swap File Encryption
- BestCrypt Volume Encryption

We offer you a fully functional trial version of BestCrypt: http://www.jetico.com/download.htm

If You Want to Comment on the Software

If you have a product suggestion or comments on how to make BCArchive documentation better, send us E-mail at this e-mail address:
support@jetico.com

Be sure to include your name, e-mail, version number of BCArchive. Please visit Jetico WWW-site to get information about our other products, Frequently Asked Questions lists, Download Evaluation Software page and other:

http://www.jetico.com

We are always trying to improve BCArchive. User feedback is important and extremely valuable to the development team.

Thank you for your time!
The Jetico Team