Topic not found

Please contact the author to get the latest version of this help file.

Introduction

Welcome to Blowfish Advanced 97 - the program for protecting your private files on your Windows PC.

Blowfish Advanced 97 is a file encryption program, protecting your files with a password so that no-one but you has access to the file contents. As a second feature Blowfish Advanced 97 is able to wipe sensitive files that are no longer needed, nobody will be able to restore them.

Today we are in the information age and encrypting data becomes more and more necessary for most people. There are many reasons why data has to be protected from unauthorized access, e.g. sensitve medical data, private or business documents or just some hot stuff from the Internet.
There are many ways to secure data. Beneath cheap solutions like hiding files with steganography, the only way to make files really unaccessible is to use strong cryptography, which means high-end encryption algorithms and a password long enough to resist any attacks.
This is what Blowfish Advanced 97 does, combined with a richly featured user interface for daily work, an open cryptengine based on the UCDI which allows you to add new encryption drivers with different algorithms to the application, and the possibility to check the driver source code just for a maximum of reliability.

Equipped with the most proven and modern security technologies, capable to encrypt and wipe even the largest number of files with just a few mouseclicks, incredibly fast and easy to use - Blowfish Advanced 97 is one of the top-level software encryption products available today.

Markus Hahn, November 1997

Where to get Blowfish Advanced 97

Blowfish Advaced 97 can be downloaded from many servers around the world.

The latest versions will first appear on Markus Hahn's web page located at

http://members.tripod.com/mc_hahn/software.html

There you will also find additional tools plus the newest UCDI drivers and their source codes. The address of the page may change, if necessary contact the author for the new location.

The standard distribution server for Blowfish Advanced 97 is at

http://www.tucows.com

in the "Windows 95 / Security Applications" section. Tucows is present all around the world through its mirror servers and contains the latest and best shareware and freeware available. Due to their reviews the software will need some days until it is available on the server.

Registration

How to register

This is an evaluation version of Blowfish Advanced 97.
After the trial time of 30 days max. you should register your own copy of Blowfish Advanced 97 for the small fee of just

US $20.00

Otherwise you must stop using this software.

Registered users of Blowfish Advanced 95 can update for US$ 10.00.

If you want to register the program now please use the **Registration Form**.

If you register, you will get your personal key to unlock the software's 5 characters password restriction. This key will also unlock all updates of the program.

Soon there will be the possibility to register to software online. For the latest information please check out the author's webpage.

Copyright, Warranty and Contacting the Author

Distribution

You are encouraged to spread this shareware around the world by publishing it on CD-ROM, BBS, offering it via FTP or the World Wide Web. The only limitation is that the complete package must not be changed in any way.

Export Restrictions

The maximum key length of this evaluation version has been restricted to 5 characters (equals 40 bits), so that **the sharware version of Blowfish Advanced 97 is re-exportable from the United States**.

Some countries (such as France, Iran, Iraq, Russia and China) have laws that prohibit or regulate the use of cryptography. Please check out your country's national laws and governmental policies on cryptography before attempting to use the registered version of Blowfish Advanced 97. In some countries, the use of cryptography is restricted by law. For example, in the UK and Germany it is illegal to transmit encrypted data by radio communication. In some countries, it is outright illegal to encrypt data at all (France). In other countries, they are working on it.

Warranty

Please read this help file carefully before using the program with your data. This program contains powerful encryption technology and data wiping features that can destroy data if used improperly. This software is provided as is and without warranty. The author assumes no liability for damages, either direct or consequential, which may result from the use of this product.

Copyright

Contact

If you have problems, questions or suggestions you can contact the author at the following address by snail-mail:

Markus Hahn
Schellingstrasse 13
72622 Nuertingen
Germany

Or send an e-mail to:

hahn@flix.de

Latest versions of Blowfish Advanced 97 can be downloaded at many software servers in the Internet or directly at the author's webpage.

To: Markus Hahn
Schellingstr. 13
72622 Nuertingen
GERMANY

Registration form for Blowfish Advanced 97

Print this file and send it to the address above.
Please write clear and legible!

Name_____ Date_____

Company_____ Phone_____

Street_____ FAX_____

City _____ State _____ Country _____ Zip _____

E-Mail _____

Please tick:

[ ] x Single User License: US $20.00

[ ] x 10 User License: US $150.00

[ ] x 25 User License: US $300.00

[ ] x 100 User License: US$ 800.00

[ ] x Company Network License: US$ 2000.00

[ ] x Update Blowfish Advanced 95: US $10.00

Payment: [ ] cash in $US (enclosed)
[ ] Eurocheque (enclosed), add DM 5.00
[ ] other cheque (enclosed), add US $5.00

Please send me the registration key via:

[ ] Postal Mail (key and software on 3.5" floppy), add US $5.00
[ ] E-Mail (don't forget your address above!)

Total US$ _____

I assure that the import of strong cryptography in my
country is legal and that I'm allowed to use Blowfish Advanced 97.

Signature_____

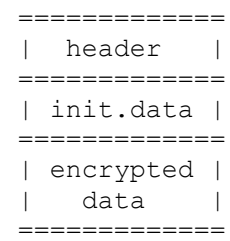
File Format and Technical Reference

File Format

In Blowfish Advanced 97 data has to pass two layers before being read from or written to an encrypted file.

The first layer ist called CryptFile. This layer enables the program to work with any UCDI UCDI driver, independant of its block size, key size or block link mode. The CryptFile layer was designed to implement a basic UCDI server which other programs can extend for their own needs.

A file created with the CryptFile layer only will just look like that:



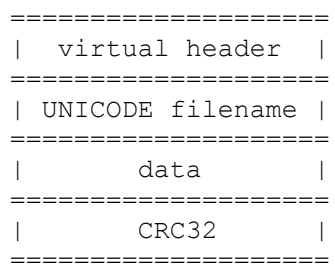
The `header` is described in Object Pascal in the following way:

```
TCryptFileHeader = record
  Magic : Integer;
  SizeOfHeader : Word;
  Version : Word;
  LengthLo : Integer;
  LengthHi : Integer;
  CipherInitDataSize : Word;
  CipherBlockSize : Word;
  keySalt : array[1..11] of Byte;
  KeyHash : Integer;
end;
```

`Magic` is a 32bit constant for identifying an encrypted file.
`SizeOfHeader` stores the number of bytes the header needs.
`Version` is a 16bit representation of the version of the CryptFile module, where the higher byte stores the major and the lower byte the minor number.
`LengthLo` and `LengthHi` present together a 64bit number describing the number of encrypted bytes stored in the file. Choosing a 64bit representation allows future versions to handle even very large files. Blowfish Advanced 97 supports only file length up to 4 GB.
`CipherInitDataSize` holds the number of bytes the algorithm needs to decrypt the data, beneath the password, of course.
`CipherBlockSize` stores the block size of the algorithm used to encrypt the data. Many encryption algorithms encrypt data not bytewise but blockwise, which means e.g. putting 8 bytes together and encrypt them in one pass. This enables the program to detect incompatible algorithms on the fly.
`keySalt` stores 11 bytes of salt. Salt is just random data and is appended to your password before it is going to be used for encryption. The result is that even with the same password each encryption will look different, even if the algorithm does not chain blocks (ECB mode).
`KeyHash` holds a 32 bit checksum of your key and the salt. This checksum is neither reversible nor can anyone get any information out of it, which would lead to the original password. The checksum is built from the 128 bit MD5 hash of the password+salt, XORing all 32bit quaters together. This allows a quick recognition of wrong passwords. As you can easily see the chance, that a wrong password passes such a

check is 1 : 2 32 (or 1 : 4,294,967,296) which is enough for the normal usage, encrypting at least thousands of files, but too less for brute-force attacks where someone must try out much more possibilites.
Right after the header the initialisation data of the used encryption algorithm is appended. The number of bytes equals the value stored in the header.
After the init. data the encrypted data follows. The length of the encrypted data must be adjusted to the block size of the algorithm which was used for encryption. E.g. data encrypted with Blowfish will be aligned to the next 8 byte border. If you have e.g. 29 bytes to encrypt all the encrypted data will be 32 bytes large.

The second layer used by Blowfish Advanced 97 is stored completely in the data section of an encrypted file and looks like that:



The great advantage of the double layer technology is that the second layer doesn't have to care about anything that has to do with encryption. It just passes its data to the first layer which handles the encryption and the file input and output.
The virtual header can be described in Object Pascal like that:

```
TVirtualHeader = record
  Magic : Integer;
  HeaderSize : Byte;
  LowestVer : Byte;
  filetime : TFileTime;
  Attributes : Integer;
  FileLen_Lo : Integer;
  FileLen_Hi : Integer;
  FileNameLen : Word;
  Compress : Byte;
end;
```

`Magic` is a 32bit constant for identifying a file especially encrypted with Blowfish Advanced 97. Remember that other programs might use the CryptFile layer described above for other purposes, too.
`HeaderSize` stores the number of bytes a header needs.
`LowestVer` holds the lowest version number of the engine needed to decrypt the file correctly.
`filetime` is a 64 bit date+time stamp from the original file, compatible to the Win32 API functions and more flexible than the old 32 bit date+time stamp used by DOS and Win16 programs.
`Attributes` stores the original file attributes, e.g. the readonly or hidden attribute.
`FileLen_Lo` and `FileLen_Hi` hold the original 64bit file length.
`FileNameLen` stores the number of bytes used for the original file name, which is appended after the header.
`Compress` is a flag which tells the compression state of the file. If this byte is zero the data wasn't compressed. Actually the standard compressor is LZ77.
The filename is appended to the virtual header in the UNICODE format, using a 16bit value for each character. This features makes the file format very well prepared for future operating systems which uses UNICODE filenames for standard, e.g. Windows NT.

After the filename the original file data follows.
A 32bit CRC32 checksum closes the whole data stream which is passed to the first layer to be encrypted in its data section there.

All integer values are stored in Intel byte order. The structures are not aligned in any way, so the smallest size is used, e.g. the size of `TVirtualHeader` is 4 + 1 + 1 + 8 + 4 + 4 + 4 + 2 + 1 = 29 bytes.

Key Setup

Different encryption algorithms require different key lengths. The Blowfish encryption algorithm needs e.g. a key of 56 bytes. It is very uncomfortable to find passwords that have exactly the right length every time, so the program converts the password into a key for the individual algorithm.

Blowfish Advanced 97 uses a key setup in which your password is hashed with SHA-1, the "Secure Hash Algorithm". The advantage is that the key result is in binary form and looks like random data. Additionally the length of the password is not restricted to the maximum key length of the selected algorithm, so it can be hashed up or down to the right size.

To understand the key setup of Blowfish Advanced 97 let us make two examples.

Let our password be "helloworld". We want to create a key of 16 bytes. The SHA-1 allows us to input as much data bytes as we want to, and puts out a hash of 20 bytes. A hash is the same like a CRC32 checksum, but secure for encryption purposes.
To resize the 20 bytes of the hash to the required 16 bytes for the key we take the first 16 bytes of the hash and XOR the rest of 4 bytes over the beginning of these 16 bytes. So we didn't ignored any part of the hash:

```
password:                      "helloworld"
                                     |
                                  SHA-1
                                     |
                a3d4ff09e22710946702eab2cc382596a8e3197322
                a3d4ff09e22710946702eab2cc382596a8
                ||||||||
          XOR   e3197322
                ||||||||
key:            40cd8c2be22710946702eab2cc382596a8
```

In the second example we assume that our password is still "helloworld" but we need a key for Blowfish which has the required length of 56 bytes.
As already mentioned SHA-1 only returns 20 bytes. So we have to create 36 bytes more from the password in the following way: we hash the password with SHA-1 and get 20 bytes. Then we add those 20 bytes to the original password and hash the modified password again. The result is a new hash which means 20 new bytes for our key. Due to the modified password this new hash is completely different from the first one. Now we append this second hash to the modified password again and rehash it to get the last 20 bytes. Of course now we have 4 bytes too much, so we XOR them over the first hash as we did in the first example. Now we have the needed 56 bytes for the Blowfish encryption algorithm.
Please remember that your password is always combined with 11 bytes of salt.
SHA-1 is besides MD5 the most secure hash algorithm available today.

Key Disk Creation

Beneath conventional passwords Blowfish Advanced 97 is able to use keydisks instead. A key disk contains always one file with the max. size of 64 kB which binary content is used as the password.

Of course every file can be used a key file, but Blowfish Advanced 97 offers a smart method to create key files very fast and secure.
For creation the program request 128 key kits from the user. As you might have recognized the program accepts keys with no ASCII code like F1, too. So every key hit delivers the program a 2 byte key code, 256 bytes total.
To improve the randomness of this key data the key is extended to 2 kB (2048 bytes) in the following manner: the 256 bytes are splitted into two halves, each 128 byte halve is hashed by MD5 and the hash splitted into four 32bit values which were used as the seed for a LFSR (linear feedback shift registers) random generator which outputs 256 bytes every time. By this way we get 2 * 4 * 256 = 2048 bytes of random data, which were stored in the keyfile.
In the shareware version the size of a keyfile is just 5 bytes, created by hashing the 256 bytes and then XOR-warp-around the 16 bytes of the MD5 hash into the file.

Algorithms

Blowfish Advanced 97 comes up with now 7 standard UCDI drivers, implementing Blowfish, PC1 (an RC4, clone), triple-DES, GOST, Cobra128, TwoFish and CAST. IDEA can be downloaded as a freeware driver.

Blowfish

The algorithm was designed by Bruce Schneier (e-mail: schneier@counterpane.com). Blowfish is a very fast algorithm, performing excellent on modern 32bit processors. Another advantage is its variable key size up to 448 bits (56 bytes). It was first published in Doctor Dobb's Journal, issue 4/94. After a year of intensive cryptanalysis it was still unbroken (as reported in DDJ 10/95).

CAST

CAST comes from Canada (designed by C. Adams and S. Tavares). This algorithm has a good performance and runs well on 32bit processors. It is resitant to any know cryptoanalysis and thus can be considered secure. It's also an AES candidate to replace DES in the near future. The UCDI driver for Blowfish Advanced 97 implements CAST with a 128bit key (also known as CAST5) and was written by Walter Dvorak (e-mail: e9226745@student.tuwien.ac.at).

Cobra128

This is a new algorithm, designed by Christian Schneider (e-mail: schneider@interdevelopment.de). It was published in the newsgroup sci.crypt.research in the middle of April 1996. You can describe Cobra128 as a mutation of Blowfish using some interesting and already proved extending techniques. Cobra was originally designed to be a 128bit block cipher with 24 encryption rounds and a key size of 72 bytes. Due to it's open architecture it can be reduced or extended for larger or smaller block handling. Blowfish Advanced 97 uses Cobra128 with 24 rounds. The UCDI driver is also the first reference implementation written in C. For more information about Cobra please contact Chris via e-mail.

GOST

An algorithm descending from the former Sovjet Union. It's like the russian counterpart to the western DES algorithm. Although it has been used for a long time there are no known weaknesses. The only strange part is a short table of fixed data (the so called substitution boxes, s-boxes). These data is exchangable and might have influence on the algorithms security. Now you can speculate that some USSR institutions might have had "better" s-boxes and some "minor" institutions s-boxes easier to crack, but nobody knows. The s-boxes used in BFA97 are choosen randomly so there's no built-in weakness. GOST isn't as fast as Blowfish, but it encrypts data in 32 rounds as standard (however the encryption function is simpler than the one of Blowfish). The key length of GOST key length is 32 bytes (equals 256 bits).

PC1

This algorithm is 100% compatible to the RC4 stream cipher. RC4 was developed in 1987 by Ron Rivest. 1994 someone posted the source code in a mailing list and since then was spread around the world. RC4 is a stream cipher, handling single bytes. The UCDI driver used by Blowfish Advanced 97 implements PC1 or RC4 respectively in ECB mode with a key size of 160 bits, instead of the most RC4 engines which use just 40 bits due to US export restrictions.

Triple-DES

DES is the standard encryption algorithm, designed by IBM in the middle 70es. Although it has been cryptanalyzed for more than 20 years no weakness was found. The only problem of DES is its short key length of 7bytes (equal 56 bits). If one has access to very fast computers one can try out all possible keys within a few hours. There are some DES variants, extending the original algorithm to a new one with a larger key. The most common one is triple-DES, where a 64bit data block will be encrypted three times with DES, using three different keys (or a single key splitted in three parts). Due to this the key length is 21 bytes (168 bits), which improves the security very much but also slows down the algorithm. The triple-DES UCDI driver included in Blowfish Advanced 97 is 100% compatible to the DES standard.

TwoFish

TwoFish is the AES candiate from Counterpane - a new, fast and very flexible encryption algorithm. After extensive cryptanalysis there are still no weaknesses known. For more information about TwoFish visit http://www.counterpane.com. The version of TwoFish used in Blowfish Advanced 97 has a key size of 256 bits.

Which algorithm is the best one?

This ist the most frequently asked question about Blowfish Advanced 97. And the answer is always the same: all algorithms offered by me for BFA97 are unbroken, which means that there's no better possible attack than brute force (trying out all possible keys). Some algorithms are rather new (Blowfish, TwoFish, Cobra128), some are proven over a long time (IDEA, GOST and especially triple-DES). Personally I would prefer Blowfish for my daily work because it's so fast. For critical encryption jobs I recommend IDEA or triple-DES. But remember: the algorithm is worth nothing if the password is too short or to simple. For speed comparison: Blowfish is the fastest algorithm. TwoFish and Cobra128 follow closely, with around 90% of the speed of Blowfish. IDEA and GOST ares just 50% slower than Blowfish. The stream cipher PC1 reaches about 40%, and the slowest one is triple-DES performing with just 20%.

Random Number Generation

The random generator used by Blowfish Advanced 97 uses a SHA-1 rescrambling method. To initialise the generator a string with various data (system date and time, drive informations, etc.) is built and hashed by SHA-1. By this we get a 20 bytes buffer of random data, from which just 16 bytes were used to avoid predictable random sequences. If another 16 bytes are requested the hash value is hashed with itself to a new digest. This method provides a much better randomness than conventional 32bit random number generators.

Data Compression

Blowfish Advanced 97 uses the LZ77 algorithm to compress data. LZ77 offers fast data compression, which is necessary for bulk data encryption purposes, but also provides a good compression ratio, comparable with ZIP compressors working in Super Fast Mode.

Wiping

Blowfish Advanced 97 offers three file wiping methods, all based on the same writing process, just differing in the number of overwrite cycles.
Overwriting files under operating systems like Windows 95 or NT is not that easy as it looks. Data is buffered and cached before it is going to be written physically to disk. Especially if a file is deleted soon after the data was written nothing or just a small part will be really erased.
The most thorough way for wiping is to overwrite the disk sectors directly, but this leads to major problems, e.g. different file systems (FAT12, FAT16, FAT32, NTFS, etc.) and the high risk of manipulating disks on such a low level.
For that Blowfish Advanced 97 uses another method instead, which works on every Win32 system properly. By a special Windows API call data will written through all caches and buffers directly to disk. You can even listen to this process, because your hard disk has to work much more than usual. After the wiping process files are just closed and then deleted. Setting a file length to zero or renaming it before deleting cause problems, especially under Windows 95.
With low level disk examination utilities you might recognize that files wiped with Blowfish Advanced 97 might be restorable. That is true, but where the original data had its place there are only random numbers now - the data has really gone forever.

Programming

Blowfish Advanced 97 was programmed in Borland Delphi 4.0 and Microsoft Visual C++ 5.0 running under Windows 95 on an P200+ processor with 64 MB RAM.

Security Aspects

If you choose a password there is one important thing you must think about first: every password barrier can be broken using the brute force attack, which means that one will try out every possible password.

If your password is three letters long and one estimates that you have only used characters from "a" to "z2 then there are 26 * 26 * 26 = 17,576 possibilities for choosing a password
In the worst case one will have to try out 17,575 combinations to find your password. With a well optimized key search program running on fast computer your password will be broken within just a millisecond.
But if you are using only 2 letters more, one must try out about 11 million combinations, which takes much more time, even with a fast computer.
The rule is simple: the more letters you use, the more combinations exist, the harder is an encrypted file to break. With a key of 6 bytes (equals 48 bits) created from a long enough password by a hash function there are 2566 = 281,474,976,710,700,000,000 combinations. Let us assume a fast computer can try out one million keys per second, then it will take about 9 years to test all possible combinations. Using just one byte more and a brute force attack really gets difficult.

How should you choose your password?

Do not use common words, such as the name of your husband, wife, daughter, son, dog, cat, lover, your insurance, house or telephone number, the numbers of your birthday or year, not even in reversed order. You can be sure these passwords will be tried out first.
If you don't want to use passwords with extra characters like "&%$*" then use passwords at least 10 letters long. Use numbers or binary values. E.g. a good password is a sentence with no sense, but which can be remembered, e.g. "The dog is too green?".

Please remember always the golden rule:

DON'T FORGET YOUR PASSWORD!
REMEMBER IT!
IF YOU CANNOT REMEMBER IT: USE KEY DISKS!

There are no possibilities to restore files encrypted by Blowfish Advanced 97 with an unknown password or keydisk file of a sufficient length. If the password or the key disk is lost, you will never be able to decrypt the files. The program doesn't store the password, neither in an encrypted file nor anywhere else. The password is even deleted in memory after its usage.

Credits

Thank you to all people who have supported me in the development of this software.

Special thanks go to:

Markus Dietrich, for the manual corrections

Peter C. Gutmann, for the SFS wiping method

Christian Schneider, for the Cobra encryption algorithm

Bruce Schneier, for the Blowfish encryption algorithm and his famous book

Applied Cryptography
2nd Edition, John Wiley & Sons Inc., ISBN 0-471-11709-9

Walter Dvorak, for the CAST implementation

All Betatesters

Inprise (http://www.inprise.com), for the Borland Delphi 4 programming language.

Drag and Drop Capabilities

Blowfish Advanced 97 provides multiple drag and drop support.

Beneath the dragging and dropping of items inside the file browser the program allows drops on its icon and on the program at runtime.

If a drag and drop action was detected Blowfish Advanced 97 will ask you what to do with the dropped files and folders. You can select to encrypt, decrypt, wipe or view the provided objects and force the program to terminate after the job is done (if the application was started by drag and drop). This is especially useful if files and folders were dropped onto the program icon and you do not want to do anything else.

Button Bar and Browser Tools

The button bar enables you to to start the file actions intuitively, to call the options dialog, to exit the program, to get the program version info and to call the help file.

The browser tools assist you to navigate fast and very comfortably through your file system, to create a new folder or set the browser viewing style.

Please select the following descriptions to get further information:

Button Bar:

Encrypt
Decrypt
Wipe
Options
Exit
About
Help

Browser Tools

Up One Level
Create New Folder
Browser Styles
Favorites

Encrypt

By pressing this button Blowfish Advanced 97 starts to encrypt all the files and folders you have selected. If you select a folder every file and subfolder in it which match the file mask and the attributes are going to be encrypted.

Please remember that encrypted documents cannot be loaded into their relayed applications anymore. For that the program offers different schemes for renaming the files.

Do not encrypt anything you might need for running your system properly, e.g. initialisation files or system files.

Before the starting the encryption process a password or keydisk must be entered in the password dialog.

Decrypt

By pressing this button Blowfish Advanced 97 starts to decrypt all the files and folders you have selected. If you select a folder every file and subfolder in it which match into the file mask an the attributes are going to be decrypted.

Before the decryption process starts a password or keydisk must be entered in the password dialog.

Wipe

By pressing this button Blowfish Advanced 97 starts to wipe all the files and folders you have selected. If you select a folder every file and subfolder in it which match into the file mask and the attributes are going to be wiped.

You can select different wiping methods for faster execution or more overwrite cycles.

Be careful!
If a file has been wiped its original data will be lost forever and cannot be restored even with low-level disk editing utilities!

Options

Here you can call the option setup dialog to configure Blowfish Advanced 97.

Options are useful to switch the functionality of Blowfish Advanced 97 in a kind of way to fulfill your personal wishes as best as possible.
With options you can use special features of the program, manage your UCDI drivers, and so on.

Although Blowfish Advanced 97 works fine in its default configuration, please take some time and try to learn more about the options, so you would be able to use all the powerful features offered by Blowfish Advanced 97.

For more informations please have a look at the overview.

Password Dialogs

The password dialogs offer you the possibility to enter a password or to set a key disk and to change to most common options.

Exit

Click on this button to exit Blowfish Advanced 97.
All actual settings will be stored in the configuration file BFA97.INI. You can force the program not to store sensitive settings in the options dialog.

About

Here you can read some informations about the program status (shareware or not) and the version numbers of all modules used by Blowfish Advanced 97.

Help

Calls this help file.

Up One Level

The common button for reaching the next directory level above, e.g. to change from "C:\WINNT\SYSTEM" to "C:\WINNT". This button will be ignored if the root directory has been reached already.

Create New Folder

Click this button to create a new folder in the actual path.
The default name is "New Folder", but the item entry of the new folder will be set immediately into edit mode, so you can rename it.

Browser Styles

These are the common four buttons for changing the browser style.

Those styles are:

- Large Icons, useful for comfortable browsing
- Small Icons, to get more icons on the screen, used for huge file collections
- List, the same, but you have a horizontal instead of a vertical scrollbar
- Details. to get all file informations, date and time, attributes, etc.

Favorites

Here you can enter a new path and selection to which the browser should change.
E.g. "C:\WINDOWS*.EXE" will show you all EXE files. Just typing "C:\WINDOWS" will bring up all files in the directory. This feature enables you to filter out just those files you are looking for, e.g. extracting *.TXT files in a folder which is full of other files with different extensions.

If you want to store the actual path in the favorite list, just click with the right mouse button on the control and then selecting **Add to Favorites...**. Then the path will be permanently available in the drop-down list. So you can change completely different paths very quickly and don't have to browse down and up with far too much trouble.

You can edit your favorites by clicking with the right mouse button on the control and then selecting **Organize Favorites** from the popup menu. In the dialog that appears you can remove old entries from the list.

Apply

A click on this button will fix the actual settings, so they will be stored in the configuration file when the program is terminated and will be reloaded the next time. Otherwise all modified options will vanish.

This function is useful if you want to change some settings not only for the actual session.

Options

Password Dialog Options:

- Password Input
- Show Password
- Auto Confirmation
- Use Keydisk
- Store/Restore Pathnames
- Wipe Original Data
- Compress Data
- Remove Source Files
- Stealth Filenames
- Target Path for Encrypted/Decrypted Files
- Keep This Key

UCDI Drivers:

- UCDI Driver Selection
- Add Driver
- Remove Driver
- Test Driver
- Test All Drivers
- Test Drivers During Startup

File Handling:

- Append ".bfa" Extensions to Encrypted Files
- Write-protect Files after Encryption
- Ignore CRC32 Errors
- Warn before Overwriting Existing Files
- Skip already Encrypted Files
- Confirmations
- Show Job Report
- Log All Messages To Job Report
- Temporary Path for Viewing

Wiping:

- Let Wiping Operations be confirmed
- Wiping Method
- Rename Method

Browser:

- Exclude Files
- Style

Miscellaneous:

- Save Sensitive Settings
- Show Hints
- Flashing Progress Window
- Flat Buttons
- Fix problem with binary passwords
- Key Disk File

Create Key Disk
Clear Password List
Install/Remove File Types

Password Input

Here you put in your password to encrypt or decrypt the files you selected.

The password can be everything, a single word or a complete sentence. The maximum length of a password is 32,000 characters.

You can also type in the complete ASCII character set by using **binary sequences**.
E.g. if you want to have the character with the ASCII code # 8 (that's the code for the backspace key) at the end of your password named "special" you just type "special\08". The "\08" sequence tells Blowfish Advanced 97 to ignore the "\" prefix and to take the next two characters as the hexadecimal representation of the ASCII code. If you want to use the "\" alone you will have to type "\\".
By this method the key spectrum can be extended easily and brute-force attacks will be much harder.
You may even use zero bytes in your password, e.g. "sec\00ret".

If the password cannot be detected with Auto Confirmation turned on or if Auto Confirmation is turned off you will have to enter the password (for encryption) a second time to avoid typing mistakes.

Please read the Security Aspect section to avoid flaws when choosing your password.

Show Password

If your are sure that nobody (not even a hidden camera) has the possibility to look over your shoulder you can turn on this switch to see your password instead of the blank characters.
Blank characters are much better to hide a password instead of the usual asterisks, because it is harder for someone to guess the length of the password when (s)he has the chance to peep.

Auto Confirmation

Confirming a password again and again gets some kind of bore after a while.
With **Auto Confirmation** turned on Blowfish Advanced 97 will remember the password you entered and confirmed. If you enter the password later again the program detects it and continues without further requests. As soon as your password is recognized the password input dialog will be closed, so you need not even press the enter key. This saves a lot of time and avoids typing errors which occur even with confirmations. After retyping a password again and again it gets so common that you may even duplicate a wrong key-stroke.

How will a password for Auto Confirmation be stored by Blowfish Advanced 97?

Of course your password will not be saved in its original form. It is sufficient to save a secure checksum of it and to compare this checksum with the one from the password just entered.
If you have look into the configuration file BFA97.INI the following list can be found:

```
[PasswChecks]
NumOfPasswords=3
Password#1=C9512E680AC7308E
Password#2=B10B567EC7682FBF
...
```

Here you see the checksums of the passwords. They are neither reversible to the original passwords nor can anyone get any information out of them.
For the experts: the first 32 bits of an entry store a password salt, the second half stores the checksum. The checksum is created by hashing the password plus the salt with MD5 and then XOR all four 32 bit quaters together.

Use Key Disk

Activate this switch if you want to use a keydisk instead of a password.

A keydisk is a removable disk, usually a conventional 3.5" floppy disk, with a defined key file (maximum length: 32 kB) on it. By default this file is located at "A:\BFA97.DSK", but you can change this setting in the options dialog. There you can also create a secure key disk file with real random data.

Insert the key disk only at the time when Blowfish Advanced 97 wants to read it. That will be shortly after you leave the password dialog with the "OK" button.

Using key files is the right choice for those people who have problems to remember or don't want to remember passwords. It is also a fine feature for sharing keys in a small workgroup.

Please backup your key disks!
If a key disk is lost your data encrypted with the key file will be gone.

Keep key disks at a safe place!
If someone has access to them (s)he will be able to copy it and then decrypt all your encrypted data.

Store/Restore Pathnames

Depending on the action you started you will either be able to store pathnames (for encryption) or restore pathnames (for decryption).

Storing pathnames is useful if you want to keep complete directory structures, but don't want to leave the files at their original locations. The original pathname of a file will be stored in the header of the cryptfile. Also encrypted, of course. If you decrypt the file its original path will be restored, even if it does not exist.

E.g. if you encrypt the file "D:\Documents\Private\mydiary.doc" with pathname storing turned on and you decrypt the encrypted file (with activated pathname restoring) to the target path "E:\Backups" the original file will be recreated as "E:\Backups\Documents\Private\mydiary.doc".

Wipe Original Data

This option tells the program not only to delete an original file but also overwrites its data before. The reason is that when you delete a file only its name entry will be declared as invalid, but the original data stays on your hard disk until it will be overwritten by a new file's data. It is hard to say until when this situation will occur. But it is easy to undelete a file or to recover the data with a disk editing utility.

In the options dialog you can choose how and how often a file should be overwritten.

Compress Data

Blowfish Advanced 97 can compress file data with the LZ77 algorithm before encrypting it. By setting this switch you can turn on data compression.

If a file cannot be compressed, e.g. multimedia files like JPEG or archives like ZIP, it is stored in an non-compressed form to avoid waisting disk space. Please check this out, because the program tries to compress every file and starts a second time, with compression temporarily turned off.

Compression is time-consuming, due to that the whole process slows down. But on high compressable documents, like text files or spreadsheet tables it does not only save disk space, but increases the security, too. This is because cryptanalysis on compressed data is much harder then on noncompressed data.

Remove Source Files

If you encrypt or decrypt files to another target path the process is like copying with additional en- or decryption. By activating this switch the source files will be removed, so the process is much like moving files.

It is strongly recommended to wipe the file data when removing original source files. For decrypting files this is not necessary at all

Stealth File Names

Activating this option will cause a complete renaming of encrypted files. This is useful because then nobody can predict anything abour what was in your file when the original file name and extension have vanished.

Depending on the settings in the options dialog files will be renamed either with random characters or with an enumerating name mask. There you can also choose if the program should add the ".bfa" extensions to encrypted files.

Target Path for Encrypted/Decrypted Files

By default encrypted files will be written at the same place where the original files were located. If you want to have the encrypted or decrypted files at another location, e.g. for backup purposes, you can type in a destination if the switch is turned on.

Use the "Browse..." button to select a previously entered pathname from the list or to browse for a new folder.

In combination with the Store/Restore Pathnames switches you can recreate complete directory structures where you want them to be.

Keep This Key

The password or keyfile content respectively will be hold for the next decryption request, so a second password input action won't be necessary. This switch is only used in the small password dialog for viewing encrypted files and for executing jobfiles.

UCDI Driver Selection

Here you select the encryption driver you want to use from the list. Every UCDI driver returns its title which can be found in the list box.

Please read the UCDI page for more informations about the Universal Crypt Driver Interface.

Add Driver

Here you can add another UCDI driver to the list. UCDI drivers should have the ".UCD" file extensions, but you can also add files with any other naming convention.

Please test the UCDI driver before using it to encrypt your data. Buggy drivers may destroy your data rather than encrypting and protecting it.

Do not trust and use any UCDI driver which is offered without the original source code!

You cannot really know what an unknown driver does. Perhaps it is just encrypting your files, perhaps it uses are very weak encryption behind your back or sends your password through a secret IP port anywhere else.

If you are not sure if a third part driver is valid please contact the author.

Remove Driver

Click on this button to remove the selected UCDI driver from the list.

Please use this option with care! If you loose the UCDI driver you used for encrypting your files it might become very difficult for you to decrypt your data, especially when it was not one of the original drivers of Blowfish Advanced 97. Then you will have to get it from the original source or look for a 100% compatible counterpart.

Test Driver

Every UCDI driver provides a function to test itself for its integrity.
This selftest can be called if you click on this button. If the selftest was successfull Blowfish Advanced 97 will show you the driver's characteristics.

If a driver fails its selftest you should immediately stop using it!
Be aware that damaged or manipulated drivers might destroy instead of encrypt or decrypt your files.

Test All Drivers

To let every UCDI driver in your list execute its selftest function click on this button.

Test Drivers During Startup

This option will execute the selftest functions of every registered driver during the startup process. It is recommended to activate this swich.

Append ".bfa" Extensions to Encrypted Files

You can associate encrypted files with Blowfish Advanced 97. For that the files must have the extension ".bfa". By activating this switch the extension will be added to the name of every encrypted file, independant if it was renamed or not.
E.g. a file with the original name "MyDiary.txt" will be renamed to "MyDiary.txt.bfa".

Write-Protect Files after Encryption

With this option set all files will be write-protected after encryption. This will avoid mistakes, e.g. if encrypted files are opened in other applications they can only be viewed, not altered (which may destroy them).

Ignore CRC32 Errors

To check the integrity of decrypted data Blowfish Advanced 97 calculates a CRC32 checksum and compares it with the one that was stored in the cryptfile.
If these checksums are not equal the original file will not be restored to give the user the chance to solve the problem, e.g. to fix a disk error or to retransmit the file.
You can set this switch to pass around the integrity check you can turn off the integrity check. Then a file will always be decrypted, even when the data seems to be damaged.

Please use this option with care!
Backup the encrypted file to have another chance for decryption.

Warn before Overwriting Existing Files

If you turn off this switch the program will not warn you before it is going to overwrite an existing file.

It is recommended to set this switch because you will always have the chance at runtime to skip over overwrite warnings by clicking the common "All" button.

Skip already Encrypted Files

If this switch is turned on Blowfish Advanced 97 will check if a file is already encrypted. If so it will not be encrypted a second time. By turning the switch off you can deactivate this check. Then the program will encrypt every file, whether encrypted or not.

Please remember that if you encrypt files multiple times you have to decrypt them multiple times in reversed order, too.

Confirmations

If you turn on this switch Blowfish Advanced 97 will ask explicitly before starting an encryption, decryption or wiping process.

This option is was implemented for power users to avoid unnecessary requests.

This switch does not have an effect for wiping processes. If you want to wipe directories or files without requests you must also turn off the Let Wiping Operations be confirmed switch.

Show Job Report

If this switch is enabled Blowfish Advanced 97 logs error messages and will show them in the Job Report window after the encryption, decryption, wiping or file delacking has been done.

Log All Messages To Job Report

By default Blowfish Advanced 97 only saves error messages. If you want to log all messages, e.g. to have a look over the compression ratios of all files, activate this switch.

Temporary Path for Viewing

For viewing encrypted files Blowfish Advanced 97 decrypts them to a temporary directory and wipes them when the program is going to terminate.

Here you can set a different temporary path. E.g. if you just want to have a look at your encrypted files at an other person's computer you may set the temporary path to the floppy drive to let the files be written temporarily to your own disk.

Let Wiping Operations be confirmed

If this switch is turned on every wiping process must be confirmed before it starts, to avoid destroyed data by mistake
If you want to wipe files without any warnings turn this switch off.
Please remember the the Confirmations switch must also be turned off.

Wiping Method

Here you can select from four methods you prefer for wiping your data.

Delete only

Files will only be deleted, not overwritten.
This method is not recommended.

Simple

Files will be overwritten once with random data.
Useful for wiping huge amounts of data very quickly.

DoD Method

The original data will be deleted by overwriting it three times according to the NTSC-TG-025 regulations (Version 2, Sep 1991). This is the recommended wiping method.

SFS Method

This method overwrites the file 35 times by a special algorithm which will kill every information on a magnetic storage. SFS wiping was developed by Peter C. Gutmann and had its debut in his Secure File System (SFS) for DOS. Please remember that wiping files with this method will take much time, even on fast SCSI drives. Recommended for top level security and for people with paranoia.

Rename Method

Here you can select which method should be used when encrypted files are going to be renamed to hide their original names. The renaming methods does not have an effect on ".bfa" extensions being added to an encrypted file.

With random characters

Files will be renamed to 8 random characters. The filenames look really scrambled.

With user defined rename mask

You can choose a header and the extension for renamed files. Blowfish Advanced 97 will insert a running decimal number between these two parts to avoid file collisions.

Exclude Files

With these switches you can set the file attribute mask to let Blowfish Advanced only show and handle files and folders which fits into through their attributes.

You can exclude

- files and folders with the "archive" attribute set
- hidden files and folders
- write-protected files and folders
- system files and folders

If you want to see everything in the browser just turn off all switches.

Please remember that encrypted files will store the original attributes and will be set to the standard archive attribute.

Style

Here you configure the file browser of Blowfish Advanced 97 as you like it best.

Place Drives First

Drives will appear at the top of the list.

Auto Arrange Icons

Icons will be automatically arranged when the window is resized.

Hot Tracking

The mouse pointer selects the item under itself after a short delay.

Grid

Gridlines will be shown if the browser is set to Details style.

Font...

Selects the font for the file browser.

Show Button Bar

Turns the button bar on/off. This sworks because you can start all actions by using the browser's popup menu.

Show Tool Bar

Turns the browser tools on/off.

Save Sensitive Settings

Blowfish Advanced 97 keeps its settings in the configuration file BFA97.INI, located in the same directory where BFA97.EXE is placed.

If you turn off this switch the following informations will not be stored in BFA97.INI:

- usage of a keydisk
- data compression switch
- switch for using a target path
- last used target path
- last used encryption driver
- keyfile name
- last browser path
- all destinations paths
- all favorite paths

The default values will be used instead or the lists will be cleared respectively.

Show Hints

Here you decide if the main buttons should be equipped with hints.
If you are familiar enough with Blowfish Advanced 97 you might turn them off.

Flashing Progress Window

The progress window flashes red, green or yellow depending on the executed process.
Here you can turn this cool feature on or off.

Flat Buttons

To switch to traditional button style or to use the new flat variants.

Fix problem with binary passwords

Until version 1.06 of Blowfish Advanced 97 there was a bug in the interpreter for binary passwords, which are declared through the "\xx" syntax. This problem had neither an effect on the security nor on the capability to decrypt the data correctly.

Since 1.07 this bug is fixed, binary passwords will now be translated correctly, but aren't compatible to fomerly ones. This options was implemented to give the users the possibility to decrypt their old data (which was encrypted with such binary passwords) with the always latest version of the software.

Please activate this switch always if you don't need it for decryption of old files.

Create Key Disk

To create a secure key disk click on this button.

The program will request 128 key hits from you before the 2kB (5 bytes in the shareware version) key disk file is going to be written. For details about the creation process please read the Technical Reference.

This should be your favorite method to create new, secure keydisks.

Key Disk File

Here you can define the path and name of the keydisk used by Blowfish Advanced 97.
Only such files placed on removable disks should be defined.

Clear Password List

Here you can delete all password entries created by Auto Confirmation.

Install/Remove File Types

Blowfish Advanced 97 can be associated with files owning the extensions ".bfa" (encrypted files) and ".bfj" (job files).

By clicking on **Install File Types** the necessary entries in the Windows registry will be added to show the correct icons and file descriptions. **Remove File Types** deletes those entries from the registry.

Please remember that you have to restart the applications to get the correct icons and file descriptions in the browser.

This features enables you to integrate Blowfish Advanced 97 quickly in any Windows environment and vice versa. So you don't have to execute long-winded installations or uninstalling procedures. E.g. you can temporarily work with your own copy of Blowfish Advanced 97 on other people's computers and then remove all traces.

Browser

The browser of Blowfish Advanced 97 allows you to navigate easily through your whole file system on your computer and on networks you are connected to.

Remember those special features:

- Use your own favorites for easier and faster browsing.
- A double click on an encrypted file with the ".bfa" extension will view this file.
- Dragging and dropping of files and folders causes a move process.
- Dragging and dropping of files with the ALT key pressed causes a copy process.
- Activate the richly featured popup menu with a right mouse button click.
- Use the Details style to sort the items.

Popup Menu

The popup menu can be activated by clicking with the right mouse button onto the browser and provides then all needed functionality for handling Blowfish Advanced 97.

Depending on the item below your mouse pointer some menu entries might not be visible.
Please remember that all menu items are equipped with key shortcuts, too.

The menu entries are:

Encrypt

All selected files and folders will be encrypted. This is the same as a click on the Encrypt button.

Decrypt

All selected files and folders will be decrypted. This is the same as a click on the Decrypt button.

Wipe

All selected files and folders will be encrypted. This is the same as a click on the Wipe button.

View

The selected file will be started with its associated application.
If the file has the ".bfa" extension Blowfish Advanced 97 ask you for a password or a keydisk to decrypt the file before it will start the file itself. If the key was the right one the file will be decrypted in the temporary path for viewing and added to an internal list. Just before Blowfish Advanced 97 terminates the temporary files will be wiped.
The password or keyfile content respectively can be temporarily retained, so if you want to view other files encrypted in the equal manner just double-click on the icons and view without any further password requests. Easy - but secure.

Browse

Opens the standard path browser to change the actual path.

Refresh

Forces the browser to re-read the current path.
Use this method if files or folders have been changed since the last refreshment.

Options

You can call the options dialog with the right sheet for configuring the browser.

Select... All

If you want to select all files and folders in the current path.

Select... All Files

If you want to select all files in the current path.

Select... All Folders

If you want to select all folders in the current path.

Select... Encrypted files

If you want to select all files with the ".bfa" extension.

Select... Decrypted files

If you want to select all files without the ".bfa" extension.

Select... Filenames Containing String...

If you want to select files containing a defined string within their names.

Copy

Copies all selected files to the path you have declared in the destination dialog.

Move

Moves all selected files to the path you have to declare in the destination dialog.

Delete

Removes all selected files and folders.
Please remember that items deleted with Blowfish Advanced 97 will not go to the Recycle Bin

Rename

Switches the selected file or folder item to edit mode so you can rename it.

Clear Empty Disk Space

With this option you can clear the whole empty file space on the actual drive. The program will create a temporary file with random data until the disk is full. This will overwrite every file cluster not in use.
A popup dialog will show you how much random data have already been written to the disk. You can break the process by clicking on the "Cancel" button.
It is recommended to clear your empty disk space as often as you defrag your disk.

Clear File Slack

When data is stored into a file the operating systems allocates as much clusters on the data medium as needed to keep all the file's data. A cluster has a defined size, e.g. 512 bytes or even 32 kB. The unused space in the last cluster of a file is called slack and may contain (sensitive) data from deleted files.
You can clear the slack of the selected files and complete folders by using this menu entry.

Format

Use to format the drive you selected with the standard dialog of Windows.

Destination Dialog

The destination dialog offers you not only the possibility to enter a target path but to select from a history of target paths which you have already typed in.

Use the **Remove** button to delete an entry in the list which is no longer needed.

Use the **Browse...** button to select a new path.

If a path does not exists Blowfish Advanced 97 will create it for you.

Job Report

The **Job Report** window shows you all messages created during the last encryption, decryption, wiping or deslacking process. If the Log All Messages To Job Report switch is on every handled file will cause a message, otherwise only error messages are logged.

The menu provides the following items:

Messages | Save...

Here you can store the message to a text file.

Messages | Copy

The messages will be copied to the clipboard.

Create Job File

The finished job will be saved to a job file. If the job file already exists the program query for appending the job to the file ot to overwrite it. For further informations please read the Job Files section.

Exit

Closes the job report window and returns to the main window of Blowfish Advanced 97.

Help

Starts the help file with this page opened.

Job Files

Encrypting or decrypting files is easy to manage with Blowfish Advanced 97, nevertheless it is always the same work when you encrypt, decrypt or wipe the files and folders every day.

To automatize the daily security operations Blowfish Advanced 97 offers you **job files**. A job is nothing more than a description which tells the program what and how to encrypt, decrypt, wipe or deslack. E.g. when you have encrypted a number of files you can save this action in a job file. For that use the menu entry in the Job Report dialog. If you want to execute the same job again just click on the job file. Then you will not have to select the files again, you will not have to set up the original options and the right encryption driver, you will not have to reselect the target path - you will just have to type in the password or insert the keydisk. You may even add another job to an existing jobfile. When the jobfile is executed every job stored in it will be worked out.

E.g. you may create a job file which contains decryption jobs at your sensitive files for the everyday work day in the office. Every morning you will just have to start the jobfile, enter the password and that's it. By the same way create the counterpart to this jobfile, which encrypts your files. Just add some jobs for clearing up, e.g. to wipe all temporary Internet files, and you have the jobfile to start when leaving the office.

If you have installed file types you can click on the job file directly without having to start Blowfish Advanced 97. How about placing the daily decryption job file onto the Windows desktop or into the Autostart folder?

UCDI - Universal Crypt Driver Interface

UCDI was developed to enable applications to work not with single one encryption algorithm but to offer an open interface to link drivers with any encryption algorithm.

UCDI drivers are Win32 DLLs (Dynamic Link Libraries), with the default extension ".UCD" and a defined function interface. Beneath key setup, encryption and decryption routines an UCDI driver offers informations about its block size, its block chaining method (if any), how many initialisation data has to be stored, which key length is needed and how much memory is needed for its execution. The UCDI was also designed to allow proper multithreading with the same driver by handling everything with contexts.

A big advantage is that **the source code of a driver can and should be published**, so the **trustworthness** of an application using UCDI drivers is much higher than in a program that just claims to use e.g. Blowfish in CBC mode. Everybody can check out the driver source code for flaws.

Another big chance is that **new or modified algorithms can be added to an application**. So if you want to use your own special algorithm which isn't offered by the orginal application, e.g. "triple-DES combined with IDEA in triple-encrypting mode with CFB", you just have to develop the UCDI driver with your favorite programming language like Microsoft Visual C++, Borland C++ Builder or Borland Delphi.

Although it's possible to link UCDI drivers directly to an application it will better to use a layer module fo the driver handling. Blowfish Advanced 97 uses an internal UCDI server called CryptFile to manage encrypted file data streaming with any UCDI driver.

For more information about UCDI please check out the author's webpage.

http://members.tripod.com/mc_hahn/software.html

There you can download UCDI drivers with their source codes and demo applications

The usage of the UCDI and all available drivers provided by Markus Hahn in own customer applications are free without any necessary license fees.

Blowfish Advanced CS

will be the successor of this program. It's under development actually and is going to be released in the first quater pf 1999. Currently I cannot promise to finish the software at a defined date, but I work hard on it. Beta versions will be available to the public as soon as possible.

Here are some new features of the software:

- 100% file compatible to Blowfish Advanced 97
- self-decrypting files
- ZIP-like archive format
- improved compression speed
- usage of Extended UCDI (already available on my webpage)
- integrated dummy cryptfile and -archive generator
- integrated password finder
- more professional GUI to keep all the new features
- better integration into the Windows shell
- possibility to work with encrypted files
- advanced internal achitecture (random number generation, secure memory)
- a command line version

Registered users will be able to upgrade to Blowfish Advanced CS for a small fee.