**Encrypted Backup**

*Percona XtraBackup* has implemented support for encrypted backups. It can be used to encrypt/decrypt local or streaming backup with [xbstream](https://www.percona.com/doc/percona-xtrabackup/2.3/glossary.html#term-xbstream) option (streaming tar backups are not

supported) in order to add another layer of protection to the backups. Encryption is done with the libgcrypt library.

Creating Encrypted Backups

To make an encrypted backup following options need to be specified (options [xtrabackup --encrypt-key](https://www.percona.com/doc/percona-xtrabackup/2.3/xtrabackup_bin/xbk_option_reference.html#cmdoption-xtrabackup-encrypt-key) and [xtrabackup --encrypt-key-file](https://www.percona.com/doc/percona-xtrabackup/2.3/xtrabackup_bin/xbk_option_reference.html#cmdoption-xtrabackup-encrypt-key-file) are mutually exclusive, i.e., just one of them needs to be provided):

* --encrypt=ALGORITHM - currently supported algorithms are: AES128, AES192 and AES256
* --encrypt-key=ENCRYPTION\_KEY - proper length encryption key to use. It is not recommended to use this option where there is uncontrolled access to the machine as the command line and thus the key can be viewed as part of the process info.
* --encrypt-key-file=KEYFILE - the name of a file where the raw key of the appropriate length can be read from. The file must be a simple binary (or text) file that contains exactly the key to be used.

Both [xtrabackup --encrypt-key](https://www.percona.com/doc/percona-xtrabackup/2.3/xtrabackup_bin/xbk_option_reference.html#cmdoption-xtrabackup-encrypt-key) option and [xtrabackup --encrypt-key-file](https://www.percona.com/doc/percona-xtrabackup/2.3/xtrabackup_bin/xbk_option_reference.html#cmdoption-xtrabackup-encrypt-key-file) option can be used to specify the encryption key. Encryption key can be generated with command like:

$ openssl rand -base64 24

Example output of that command should look like this:

RubxnC9bT5nFBt1qNePkjWS/CJ+fip05

This value then can be used as the encryption key

USING THE [--ENCRYPT-KEY](https://www.percona.com/doc/percona-xtrabackup/2.3/xbcrypt/xbcrypt.html#cmdoption-k) OPTION

Example of the xtrabackup command using the [xtrabackup --encrypt-key](https://www.percona.com/doc/percona-xtrabackup/2.3/xtrabackup_bin/xbk_option_reference.html#cmdoption-xtrabackup-encrypt-key) should look like this:

$ xtrabackup --user=bkp --password=bkp123 --backup --target-dir=/root/backups/encpt --encrypt=AES256 --encrypt-key="RubxnC9bT5nFBt1qNePkjWS/CJ+fip05"

USING THE [--ENCRYPT-KEY-FILE](https://www.percona.com/doc/percona-xtrabackup/2.3/xbcrypt/xbcrypt.html#cmdoption-f) OPTION

Example of the xtrabackup command using the [xtrabackup --encrypt-key-file](https://www.percona.com/doc/percona-xtrabackup/2.3/xtrabackup_bin/xbk_option_reference.html#cmdoption-xtrabackup-encrypt-key-file) should look like this:

$ xtrabackup --user=bkp --password=bkp123 --backup --target-dir=/data/backups/ --encrypt=AES256 --encrypt-key-file=/root/backups/encpt1/keyfile

**Note**

Depending on the text editor used for making the KEYFILE, text file in some cases can contain the CRLF and this will cause the key size to grow and thus making it invalid. Suggested way to do this would be to create the file with:

echo -n"GCHFLrDFVx6UAsRb88uLVbAVWbK+Yzfs" > /root/backups/encpt/keyfile

Optimizing the encryption process

Two options have been introduced with the encrypted backups that can be used to speed up the encryption process. These are[xtrabackup --encrypt-threads](https://www.percona.com/doc/percona-xtrabackup/2.3/xtrabackup_bin/xbk_option_reference.html#cmdoption-xtrabackup-encrypt-threads) and [xtrabackup --encrypt-chunk-size](https://www.percona.com/doc/percona-xtrabackup/2.3/xtrabackup_bin/xbk_option_reference.html#cmdoption-xtrabackup-encrypt-chunk-size). By using the [xtrabackup --encrypt-threads](https://www.percona.com/doc/percona-xtrabackup/2.3/xtrabackup_bin/xbk_option_reference.html#cmdoption-xtrabackup-encrypt-threads) option multiple threads can be specified to be used for encryption in parallel. Option [xtrabackup --encrypt-chunk-size](https://www.percona.com/doc/percona-xtrabackup/2.3/xtrabackup_bin/xbk_option_reference.html#cmdoption-xtrabackup-encrypt-chunk-size) can be used to specify the size (in bytes) of the working encryption buffer for each encryption thread (default is 64K).

Decrypting Encrypted Backups

*Percona XtraBackup* [xtrabackup --decrypt](https://www.percona.com/doc/percona-xtrabackup/2.3/xtrabackup_bin/xbk_option_reference.html#cmdoption-xtrabackup-decrypt) option has been implemented that can be used to decrypt the backups:

$ xtrabackup --decrypt=AES256 --encrypt-key="RubxnC9bT5nFBt1qNePkjWS/CJ+fip05" --target-dir=/root/backups/encpt

*Percona XtraBackup* doesn’t automatically remove the encrypted files. In order to clean up the backup directory users should remove the \*.xbcrypt files. In *Percona XtraBackup* 2.3.7 [xtrabackup --remove-original](https://www.percona.com/doc/percona-xtrabackup/2.3/xtrabackup_bin/xbk_option_reference.html#cmdoption-xtrabackup-remove-original) option has been implemented that you can use to remove the encrypted files once they’ve been decrypted. To remove the files once they’re decrypted you should run:

$ xtrabackup --decrypt=AES256 --encrypt-key="GCHFLrDFVx6UAsRb88uLVbAVWbK+Yzfs" -target-dir=/data/backups/ --remove-original

**Note**

[xtrabackup --parallel](https://www.percona.com/doc/percona-xtrabackup/2.3/xtrabackup_bin/xbk_option_reference.html#cmdoption-xtrabackup-parallel) can be used with [xtrabackup --decrypt](https://www.percona.com/doc/percona-xtrabackup/2.3/xtrabackup_bin/xbk_option_reference.html#cmdoption-xtrabackup-decrypt) option to decrypt multiple files simultaneously.

When the files have been decrypted backup can be prepared.

Preparing Encrypted Backups

After the backups have been decrypted, they can be prepared the same way as the standard full backups with the [xtrabackup --prepare](https://www.percona.com/doc/percona-xtrabackup/2.3/xtrabackup_bin/xbk_option_reference.html#cmdoption-xtrabackup-prepare) option:

$ xtrabackup --prepare --target-dir=/root/backups/encpt

Restoring Encrypted Backups

**xtrabackup** has a [xtrabackup --copy-back](https://www.percona.com/doc/percona-xtrabackup/2.3/xtrabackup_bin/xbk_option_reference.html#cmdoption-xtrabackup-copy-back) option, which performs the restoration of a backup to the server’s [datadir](https://www.percona.com/doc/percona-xtrabackup/2.3/glossary.html#term-datadir):

$ xtrabackup --copy-back --target-dir=/root/backups/encpt

It will copy all the data-related files back to the server’s [datadir](https://www.percona.com/doc/percona-xtrabackup/2.3/glossary.html#term-datadir), determined by the server’s my.cnf configuration file. You should check the last line of the output for a success message:

170214 12:37:01 completed OK!

chown -R mysql:mysql /var/lib/mysql

<https://www.percona.com/doc/percona-xtrabackup/2.3/backup_scenarios/encrypted_backup.html>