ISO Tools

Table of Contents

The 3K3y Ripper application can be used to manipulate PS3 game ISOs and recreate PS3 game ISOs from a collection of game files. You do not need a 3K3y Ripper connected to your PC to use any of the features of the 3K3y Ripper described in this document.

ISO Types

There are two types PS3 ISO that can be used with 3K3y, which type you can use depends on how you have configured your 3K3y.

- 1:1 ISOs, an exact copy of the original game disk, you will need a 3Dump for your console to use these ISOs.
- Decrypted ISOs, a copy of the original game disk after all the game data has been decrypted. You do not need a 3Dump for your console to use these ISOs.
- Homebrew ISOs, built without an IRD, the ISO need not be encrypted. You do not need a 3Dump for your console to use these ISOs.

Generally, decrypted ISOs are faster to build, 1:1 ISOs offer a much higher level of stealth especially when you have a 3Dump for your console.

Rebuilding ISOs

You can rebuild a PS3 game ISO from a set of files, typically these files are ripped from a PS3 game disk using Multiman or a similar homebrew application. You may, optionally, use additional data from an ISO Rebuild Data (IRD) file to improve the accuracy of the ISO that is created.
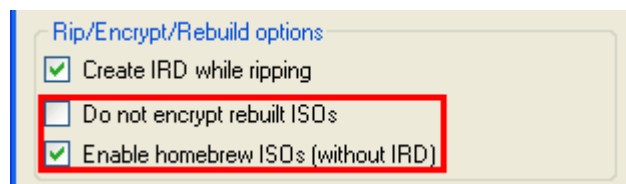
IRDs

To rebuild an ISO that is a perfect 1:1 copy of the original game disk you will need an IRD file. This is important because PS3 game ISOs rebuilt without an IRD are, at best easily detected and at worst, will not run.

During the ISO rebuild a number of MD5 hashes are calculated and checked against known good values from the IRD, if they don't match a warning is issued.

Settings

There are two options that affect the way ISOs are rebuilt. You can


- Do not encrypt rebuilt ISOs **–** Selecting this option will build ISOs without applying encryption.
- Enable homebrew ISOs (without IRD) – Selecting this option allows you to build homebrew ISOs and rebuild PS3 game ISOs without an IRD

Sometimes a PS3 system update will be missing from the game rip, or incorrect. The correct update can be automatically downloaded. If you are rebuilding a lot of game ISOs you may wish to cache the updates instead of downloading them every time.
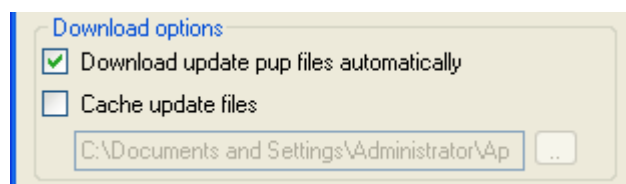
- Download update pup files automatically – Selecting this option will cause any missing PS3 updates to be downloaded automatically
- Cache update files – Downloaded PS3 updates will be cached on your HDD

Rebuilding an ISO
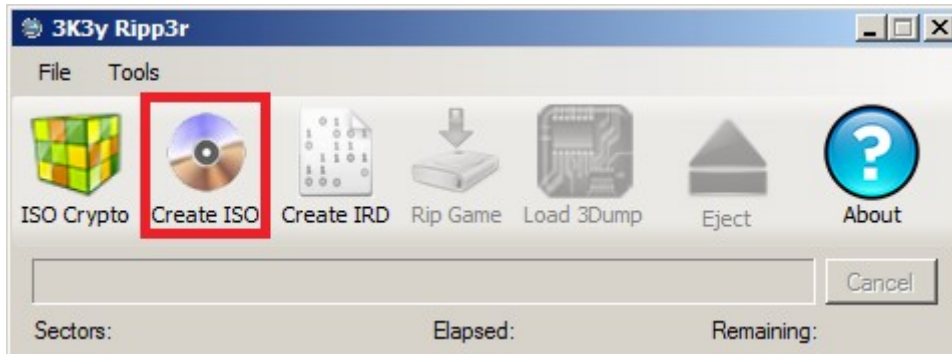
- Click the Create ISO button and use the 'Browse For Folder' dialog to select the directory containing the game files. Click OK
- Next you will be prompted to choose a folder and file name for the ISO file you are creating. Click OK
- Finally choose whether to use an IRD for the game, if required then select the IRD and click OK. If you do not want to use or do not have an IRD for the game click Cancel

The 3K3y Ripper app will now attempt to rebuild the original game ISO. Depending on the size of the game and the encryption applied this may take a few minutes or a lot longer.

Creating an IRD

An IRD file is required to rebuild a 1:1 accurate ISO from a set of files. The 3k3y Ripper App will copy the data from an ISO necessary to rebuild that ISO from just its files. The files in the ISO and the regions of the ISO are check-summed with the MD5 algorithm. All this information is stored in the IRD file for use when rebuilding the ISO.

From ISO

You can only generate an IRD from a 3K3y Ripper PS3 ISO. Generating IRDs from rebuilt PS3 game ISOs is not supported.
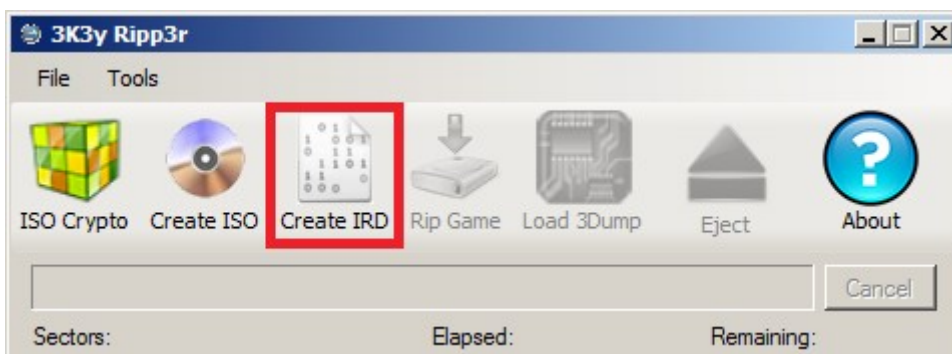
- Click the Create IRD button
- Select the PS3 game ISO you want to create an IRD for, click OK

- Select the directory and filename for the IRD, click OK

The 3K3y Ripper app will now generate the IRD file.

From Disk
You may also create an IRD while you are ripping a game, see the 3K3y Ripper User Manual for details

Uploading IRDs
Once the IRD has been created please consider uploading it to our online database. To upload an IRD select the menu option Tools | Upload Ird Files, you will then be prompted to select the IRD file to upload.

Decrypting/Encrypting PS3 ISOs
To decrypt or encrypt an ISO use the 'ISO Crypto' function of the 3K3y Ripper.
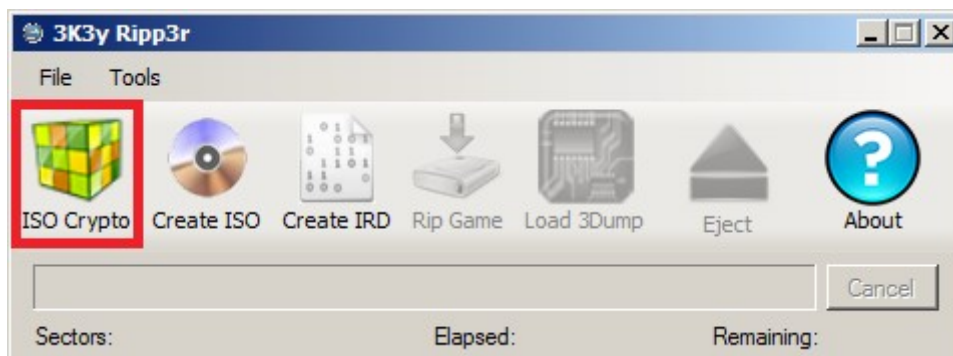

Decrypting
Select the encrypted ISO you want to decrypt (a .iso file), 3K3y Ripper will detect the encrypted ISO and decrypt it for you. The decrypted ISO (a .dec.iso file) will be created in the same directory as the original, encrypted ISO.

Encrypting
Encrypting an ISO is a very similar process, select the decrypted ISO you want to encrypt (a .dec.iso file) and 3K3y Ripper will write the encrypted ISO (a .iso file) to the same directory

Output Options
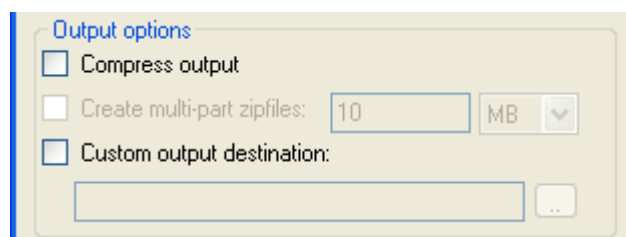
Zip files
You may, optionally, choose to have the decrypted file compressed and saved directly to a 7zip

archive. The archive can be written in pieces of fixed size.

Decrypted ISOs can be directly extracted from an archive and re-encrypted 'on the fly', when prompted to select the source file select the 7zip archive.

Output Directory
Files generated during the ISO encryption/decryption process are created in the same directory as the source file by default. The default output directory can be changed in the Settings Dialog

Application Options
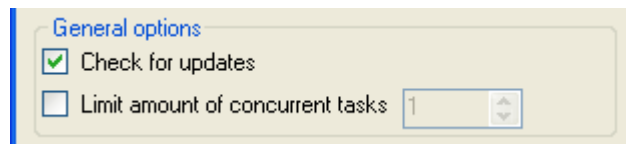
- Limit amount of concurrent tasks - By default ISO encryption and decryption will utilise as many CPU cores as are available. While this improves speed it may also make your PC slow or unresponsive. You may use this option to limit the load on your system, 1 being the minimum load.
- Check for updates – This option will check for, and download, 3K3y Ripper Application updates automatically.

Appendix A: Understanding PS3 Disk Encryption

A PS3 game disk is divided into three or more 'regions', the number of regions is always odd (i.e. 3,5,7 … 115 etc). The regions are numbered from 0, even numbered regions (0,2,4 etc) are not encrypted, odd numbered regions (1,3,5 etc) are encrypted. The key that is used to encrypt the odd regions is unique to that version of a game, it is the so called disk key, D1 or data 1.

A typical PS3 game has 3 regions:
- Region 0 (unencrypted): Contains the game icon & background image amongst other things
- Region 1 (encrypted): Contains the game data
- Region 2 (unencrypted): Contains the ps3update.pup

The disk key is stored on the game disk and read by the PS3 when the game starts, without this key it is not possible to decrypt the game data. If the PS3 cannot decrypt the data correctly you will see the 80010017 error just after the game starts.

3K3y can operate in one of two modes, 3Dump & NoKeys.
- 3Dump mode is able to respond to the PS3's request for the disk key. 3K3y ISOs contain the game's disk key and 3Dump mode uses this information to reply to the PS3's request for the disk key.
- NoKeys cannot is unable to respond to the PS3's request for the disk key and instead, relies on the PS3's BD drive to respond. The disk key in the BD drive's response is read from the original game disk in the BD drive.

ISOs ripped with 3K3y Ripper also contain an odd number of regions, the even regions are unencrypted, the odd regions are encrypted with the disk key.

3Dump mode can send the disk key embedded in the ISO to the PS3 directly and, because the disk key the PS3 received from 3K3y is correct, the encrypted region(s) data can be used directly.

NoKeys mode cannot use the disk key embedded in the ISO, instead it depends on the disk key from the original game disk in the BD drive. This will not match the disk in the ISO.

Previously it was necessary for NoKeys users to decrypt an ISO using the disk key embedded in the ISO and then encrypt it again using the disk key from the original game disk. The 3K3y FPGA handles the transfer of data from the 3K3y microprocessor to the PS3, it is now able to to perform encryption on the data as it passes through the FPGA. This means that there is no need to encrypt the ISO with the disk key from the original

game, instead the encrypted region(s) of the ISO can be decrypted and whenever the PS3 requests data from an encrypted region, the FPGA is instructed to perform the necessary encryption.

This also means that when rebuilding an ISO from a JB rip it is no longer necessary to perform the final encryption step. Since the encryption step is the most time consuming this significantly reduces the time taken to rebuild an ISO.