**Personal Password Management Guideline**

In today’s world, each individual needs to remember many passwords. There is a password for the Windows network logon, e-mail accounts, online passwords (like website member account), SFTP passwords, etc. The list is endless.

For true security, one should use different passwords for different accounts. Because if the same password is used everywhere, and if that password gets leaked due to a hack on one of the accounts, all your accounts become exposed!!! The attacker would have access to your e-mail accounts, websites, bank accounts, etc!!!

People in today’s workplace resort to store their passwords in text files or excel files. They assume that if ‘only they know the specific file’ in which passwords are kept, then the passwords will be safe. However, Security experts advise that ‘Security through Obscurity’ is not a good Security Principle.

Microsoft Excel do offer the ability to encrypt the document. However, Excel encryption is deemed weak (version dependent). Also, Excel is not custom made to support the features that a full-fledged Password Manager should offer.

**KeePass Password Manager**  
  
KeePass is a free open source password manager, which helps to manage passwords in a secure way. It allows to put all your passwords in one database, which is locked with one master key and/or a key file. So you only have to remember one single master password or select the key file to unlock the whole database. The databases are encrypted using the best and most secure encryption algorithms currently known (AES and Twofish).

Main advantages of KeePass are:

* Database is encrypted with industry leading encryption algorithms.
* Provides option for Multiple-factor authentication (supports Key file and Windows User Account Validation, on top of Master Password).
* Contains a Password Generation Mechanism which allows the generation of high entropy passwords from the chosen criteria.
* Passwords can be organized into Groups. Supports better organization of data.
* Passwords are masked out, even when the application is open and unlocked. Supports copying to clipboard without viewing the password.
* The KeePass application is cross-platform and is available in Windows and Linux.

For a full feature list, please see:

<http://keepass.info/features.html> and <http://keepass.info/compare.html>

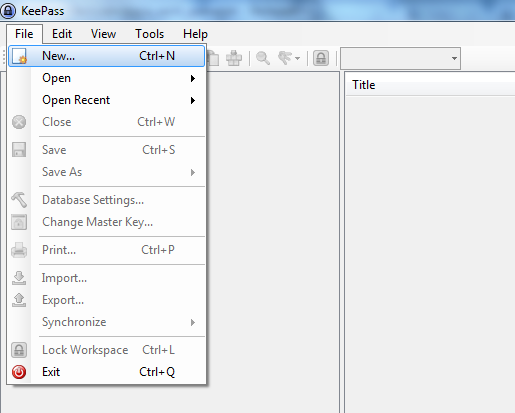
**KeePass Download**

<http://keepass.info/download.html>

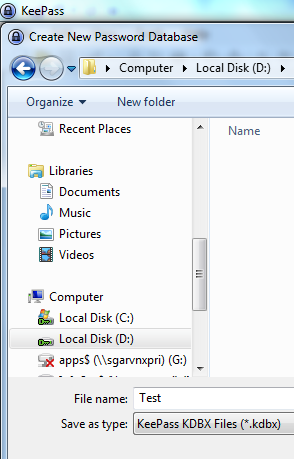
Download the **Portable** **Professional Edition** of KeyPass. After downloading the zip file, please unpack to a location of your preference.

**Creation of New Database**

Open File -> New.



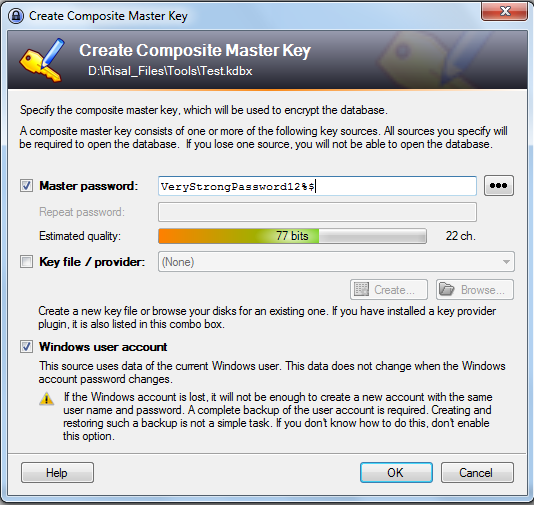
Choose Folder Location where the new database should be stored. Provide a name for the database and click ‘Save’.



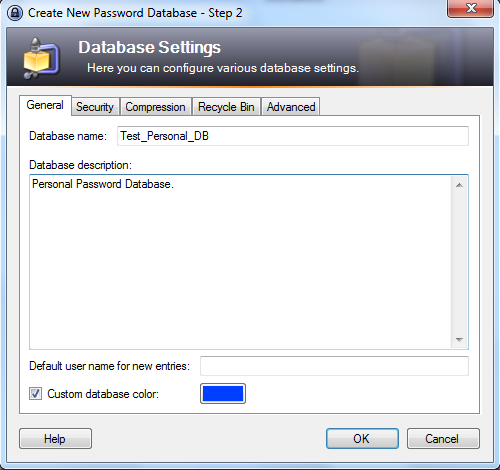
You may either create a Password Database for Personal Use or for Shared Use. Below are the recommended authentication controls for each use case, respectively:

* For **Personal Password Safe creation**:

Please enter a Master Password and select on ‘Windows user account’ option. This ensures that only you, as the current Windows User, along with the pre-set master password can open the database. Click ‘OK’. Follow next steps.

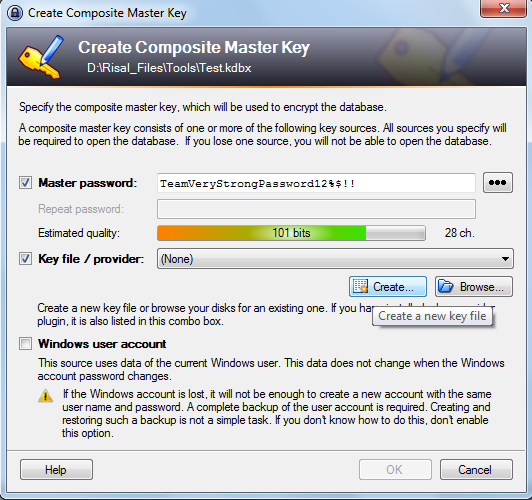


Enter Database Details and click ‘OK’.

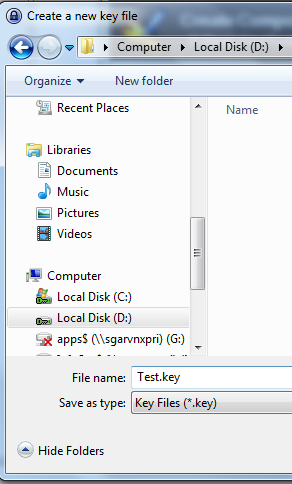


* For a **Shared Password Safe Creation** (example – to be shared between team members):

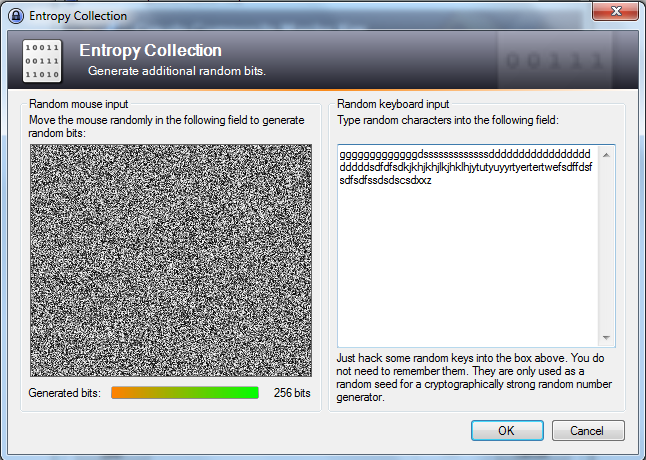
Create a Master password and a Key file. (Do not tick on ‘Windows user account’ option). Tick on ‘Key file / provider’ option and click ‘Create’.



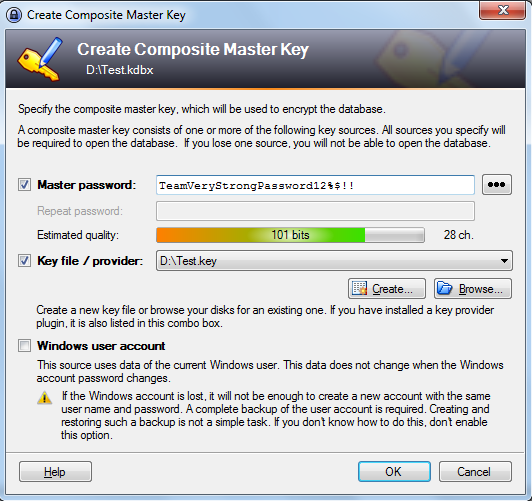
Locate folder where database should be stored. Give a name to the database. Click ‘Save’.



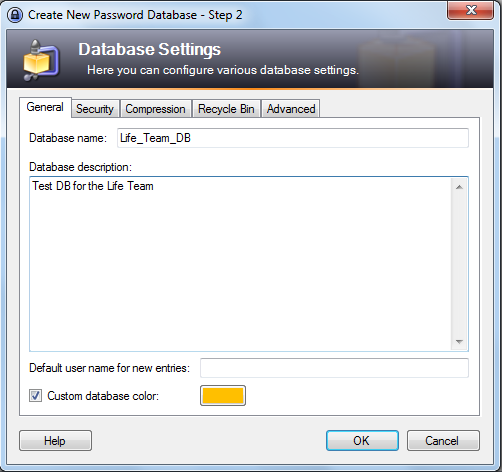
In order to generate randomness to be used for the underlying encryption, move your mouse randomly in the shown area until 256 bits are generated. Also, feed in random keyboard input to the text area.



Once the Key file is generated, and master password is entered, click ‘OK’.



Enter Database Details and click ‘OK’.

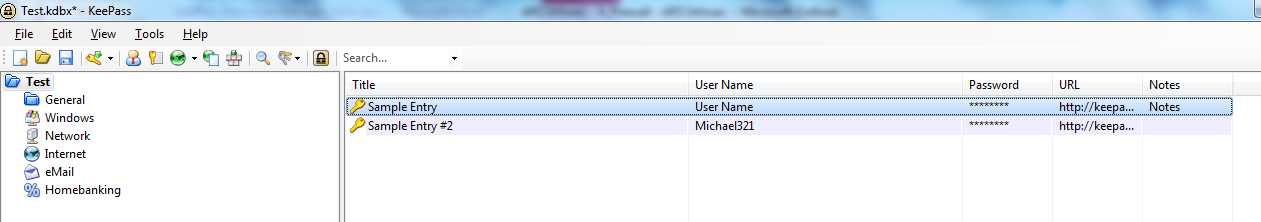


The Key file should be distributed to only those members who need to access shared database. They should keep the Key file in their local drive and point to it **locally** (Even though the database is shared and kept in a ‘shared’ drive, the Key file, should be in a ‘local drive).

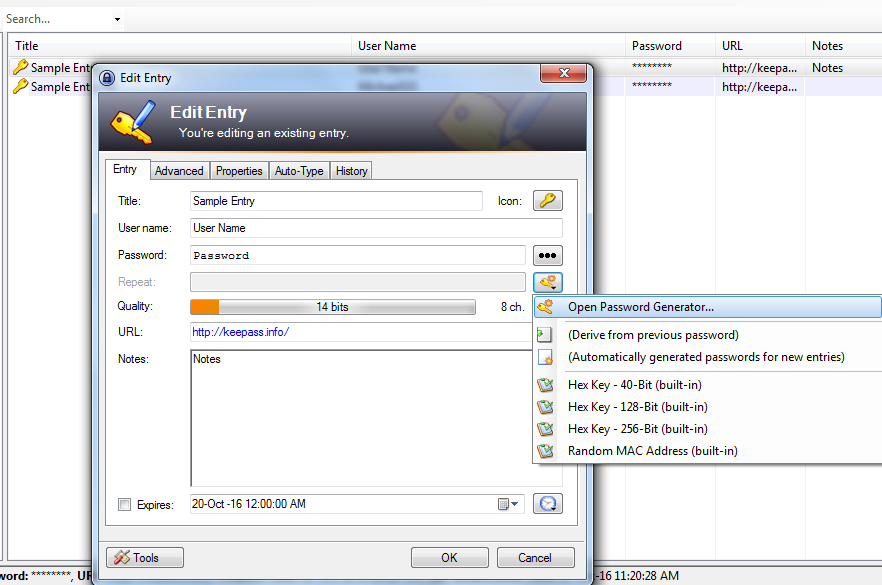
When a member of the team who had access to the database leaves, either the Key file should be regenerated and distributed; or the master password should be changed; or both.

**Password Generator Configurations**

Below screenshot shows sample credential entries.

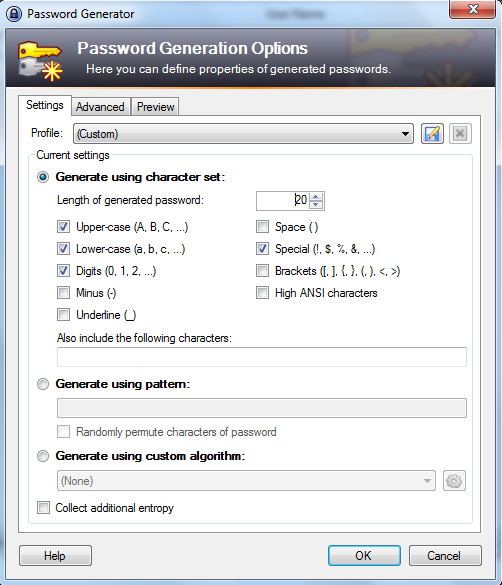


Choose a credential entry by double clicking it. Click on the ‘Key Configuration’ and select ‘Open Password Generator’.



Configure the Password Generation Settings by changing the ‘(Custom)’ profile.

Click on: [1] Upper-Case; [2] Lower-case; [3] Digits; [4] Specials; and select the length of the password as 20 characters. Click ‘OK’.



We can see that a password with the selected complexity and a high quality of entropy is generated.

