keepassxreboot / **keepassxc** Public

Code | Issues 562 | Pull requests 37 | Discussions | Actions | Projects 2 | Wiki | Security | Insights

develop ▾ keepassxc / docs / topics / **SSHAgent.adoc** Go to file ···

tocic Fix typos in docs (#8612) ✓ Latest commit e3a3734 on Oct 20 **History**

5 contributors

180 lines (133 sloc) | 10.3 KB Raw Blame

KeePassXC - SSH Agent integration

SSH Agent integration

SSH (Secure Shell) is a widely used remote secure shell protocol and is considered an industry standard for secure remote access to UNIX-like systems including Linux, BSDs, macOS and more recently even Windows received native support. SSH supports multiple types of authentication and the most widely used ones are either interactive keyboard input with a password or a public-key cryptography pair of keys.

KeePassXC SSH Agent integration is built to manage SSH keys in a secure manner by either storing them completely within your KeePassXC database or by having only the decryption key of a key file that is stored elsewhere. SSH Agent integration *does not* provide an agent itself but works as a client for any agent implementation that is OpenSSH compatible.

OpenSSH agent on Linux

If you are using a modern desktop Linux distribution it is very likely the OpenSSH agent is already configured and running when you have logged in to a graphical desktop session. This should be true for distributions like Debian, Ubuntu (including Kubuntu, Xubuntu and Lubuntu), Linux Mint, Fedora, ElementaryOS and Manjaro.

First, open a terminal and check the output of `ssh-add -l`:

```
$ ssh-add -l
The agent has no identities.
```

If you either got a list of fingerprints or the message above the agent is already running and no further setup is required. If instead you got a message saying "*Could not open a connection to your authentication agent.*" that means the agent is either misconfigured or not running at all.

Since every distribution and desktop environment is configured differently there is no general guide how to properly set it up yourself. The general rule of thumb, however, is that `ssh-agent` needs to be started as part of the startup programs for a session in a way its environment variables are exposed to all processes started by the desktop environment. One of the easiest ways to achieve this is to enable *GNOME Keyring* which should in turn start the agent as part of its services.

There are many guides on the internet how to hack your login shell to start an agent but it is very prone to errors and is not a supported configuration. If you prefer the login shell startup hack you need to set it up with a static socket path and use the *SSH_AUTH_SOCK override* option in SSH Agent settings to match that.

Warning	*GNU Privacy Guard (gpg)* with its SSH agent implementation is **not** compatible with KeePassXC as it does not support *removing* keys that have been added to it making it impossible to use any external tool to manage key lifetime.

Warning	*GNOME Keyring* prior to release 3.27.92 had its own custom implementation of an agent which does not support modern key types and was known to be buggy. It does not support any constraints you may want to configure for an added key. If you are running a modern distribution the custom agent has been removed and replaced with the stock OpenSSH agent which is feature complete.

OpenSSH agent on macOS

Apple has made OpenSSH an integrated part of macOS with automatic agent startup when it is first used. No further configuration is needed.

OpenSSH agent and Pageant on Windows

The SSH Agent integration on Windows supports both *PuTTY Pageant* and *OpenSSH for Windows 10*. Since Pageant is currently still the most widely used implementation and is easily installable on any version of Windows, it is the default on KeePassXC. However, Microsoft includes a native OpenSSH client implementation with Windows 10 since autumn 2018 that can be used instead. If you would like to self-manage your OpenSSH version you can use the builds offered via their official GitHub repository.

Pageant

Download Pageant from the official PuTTY home page at https://www.chiark.greenend.org.uk/~sgtatham/putty/

To use Pageant with KeePassXC, simply start it and it will minimize into the system tray and is ready to use. PuTTY and compatible tools will use Pageant automatically.

OpenSSH

Make sure your Windows version has at least update 1809 installed. For more details consult the official documentation.

To use Windows OpenSSH the *OpenSSH Authentication Agent* service has to be enabled first:

1. Open the Services application via the *Start Menu*, it is located in the *Windows Administrative Tools* section
2. Select the *OpenSSH Authentication Agent* and open its *Properties*
3. Set the *Startup type* to *Automatic* and start the service

Alternatively, you can use a *Windows PowerShell* running as *Administrator* to enable and start the service:

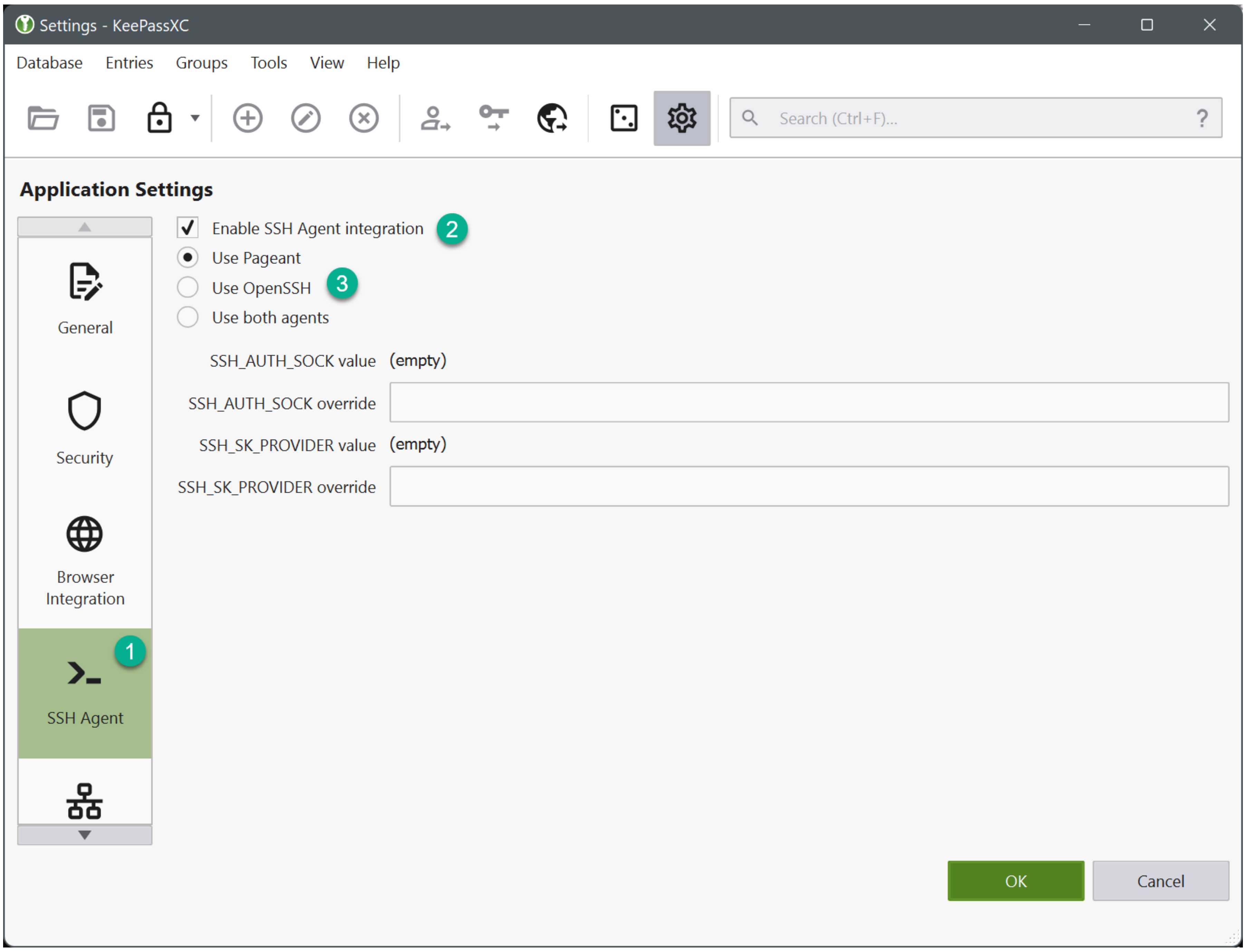
```
PS C:\Users\user> Get-Service ssh-agent | Set-Service -StartupType Automatic
PS C:\Users\user> Start-Service ssh-agent
```

KeePassXC and other compatible tools can now use the Windows OpenSSH agent. To use it with KeePassXC, update the settings explained in Setting up SSH Agent integration.

Setting up SSH Agent integration

By default the SSH Agent integration plugin is disabled. To enable integration, follow the steps below to access the settings:

1. Select *Tools* > *Settings* from the menu
2. Select *SSH Agent* category on the left sidebar



Application Settings

☑ Enable SSH Agent integration

● Use Pageant

○ Use OpenSSH

○ Use both agents

SSH_AUTH_SOCK value (empty)

SSH_AUTH_SOCK override

SSH_SK_PROVIDER value (empty)

SSH_SK_PROVIDER override

Figure 1. SSH Agent Application Settings Page

On the settings page you can enable the integration by checking *Enable SSH Agent integration*. When the integration is enabled coming back to the settings page also shows if connection to the agent is working.

On Windows, you have the option to select *Pageant* and/or *OpenSSH for Windows*. On macOS and Linux, the system ssh-agent will be used automatically and the settings page shows the current value of *SSH_AUTH_SOCK* environment variable which is used to connect to the running agent and an option to manually override the automatically detected path.

If the value of *SSH_AUTH_SOCK* is empty it means the agent is not properly configured and KeePassXC will be unable to connect to it unless you provide a static override path to the socket.

Generating a key to use with KeePassXC

KeePassXC only supports keys in the *OpenSSH* format. On Windows, *PuTTYgen* saves keys in its own format by default and you will need to convert them to OpenSSH format before being used. In this guide we are going to generate a standard RSA key in the default size.

Generating a key on Linux or macOS with *ssh-keygen*

Open a terminal window and type the following command to generate a key:

```
$ ssh-keygen -o -f keepassxc -C johndoe@example
Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in keepassxc
Your public key has been saved in keepassxc.pub
The key fingerprint is:
SHA256:pN+o5AqUmijYBDUrFV/caMus9oIR61+MiWLa8fcsVYI johndoe@example
The key's randomart image is:
+---[RSA 3072]----+
|  =. ..o         |
| o + .+ .        |
|o . .+ o.        |
| o..  Eo. .      |
|  +o .. So       |
|o*o.o+ ..o       |
|Bo=+o.+.o .      |
|+oo+.++o         |
|. ..++ooo        |
+----[SHA256]-----+
```

Now we can see two files were generated:

```
$ ls -l keepassxc*
-rw------- 1 user group 2.6K Apr  5 07:36 keepassxc
-rw-r--r-- 1 user group  569 Apr  5 07:36 keepassxc.pub
```

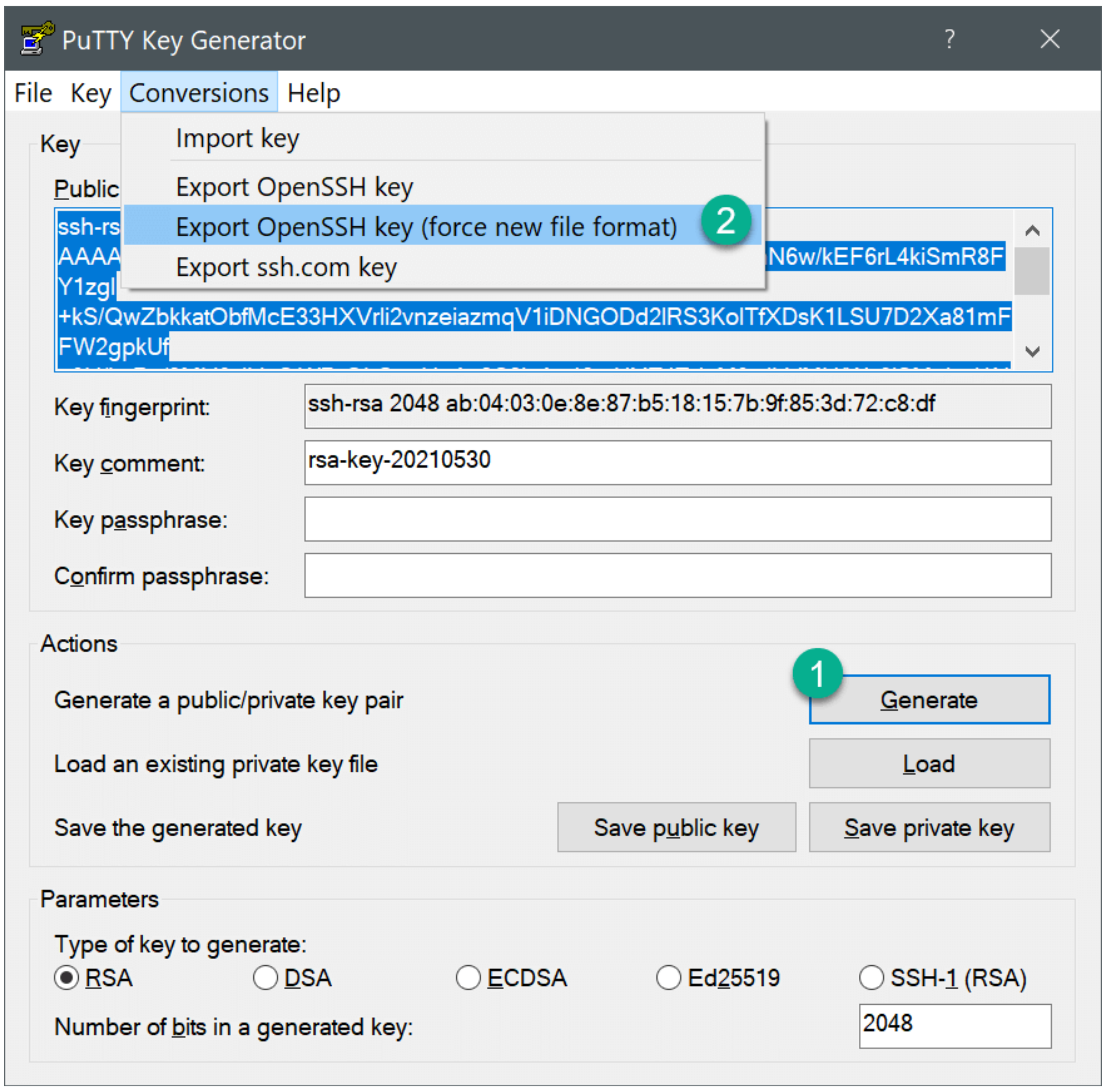
With KeePassXC you only need the first file listed.

Generating a key on Windows

On Windows you can generate key pairs with *PuTTYgen* and with *ssh-keygen*, depending on whether you installed PuTTY and your Windows version.

Using *PuTTYgen*

Please read the manual on how to use *PuTTYgen* for details on generate a key: https://the.earth.li/~sgtatham/putty/0.74/htmldoc/Chapter8.html#pubkey-puttygen. Once generated, you must save the key in the new OpenSSH format, see image below.



ssh-rs
AAAA N6w/kEF6rL4kiSmR8F
Y1zgl
+kS/QwZbkkatObfMcE33HXVrli2vnzeiazmqV1iDNGODd2lRS3KolTfXDsK1LSU7D2Xa81mF
FW2gpkUf

Key fingerprint: ssh-rsa 2048 ab:04:03:0e:8e:87:b5:18:15:7b:9f:85:3d:72:c8:df

Key comment: rsa-key-20210530

Key passphrase:

Confirm passphrase:

Generate a public/private key pair

Load an existing private key file

Save the generated key

Type of key to generate:

● RSA ○ DSA ○ ECDSA ○ Ed25519 ○ SSH-1 (RSA)

Number of bits in a generated key: 2048

Figure 2. Generating a key with *PuTTYgen*

Using *ssh-keygen*

Open *Command Prompt* or *Windows PowerShell* and type the following command to generate a key:

```
PS C:\Users\user> ssh-keygen.exe -o -f keepassxc -C johndoe@example
Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in keepassxc
Your public key has been saved in keepassxc.pub
The key fingerprint is:
SHA256:pN+o5AqUmijYBDUrFV/caMus9oIR61+MiWLa8fcsVYI johndoe@example
The key's randomart image is:
```

```
+---[RSA 3072]----+
|  =. ..o         |
| o + .+ .        |
|o . .+ o.        |
| o..  Eo. .      |
|  +o .. So       |
|o*o.o+ ..o       |
|Bo=+o.+.o .      |
|+oo+.++o         |
|. ..++ooo        |
+----[SHA256]-----+
```

Now we can see two files were generated:

```
PS C:\Users\user> dir keepassxc*
```

```
Directory C:\Users\user
```

```
Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         9/19/2021  12:08 PM           2655 keepassxc
-a----         9/19/2021  12:08 PM            570 keepassxc.pub
```

With KeePassXC you only need the first file listed.

Configuring an entry to use SSH Agent

The last step is to setup an entry to contain the SSH Agent settings and key file you generated.

1. Create a new entry, or open an existing entry in edit mode.

2. Set the password you used for the key file in the password field.

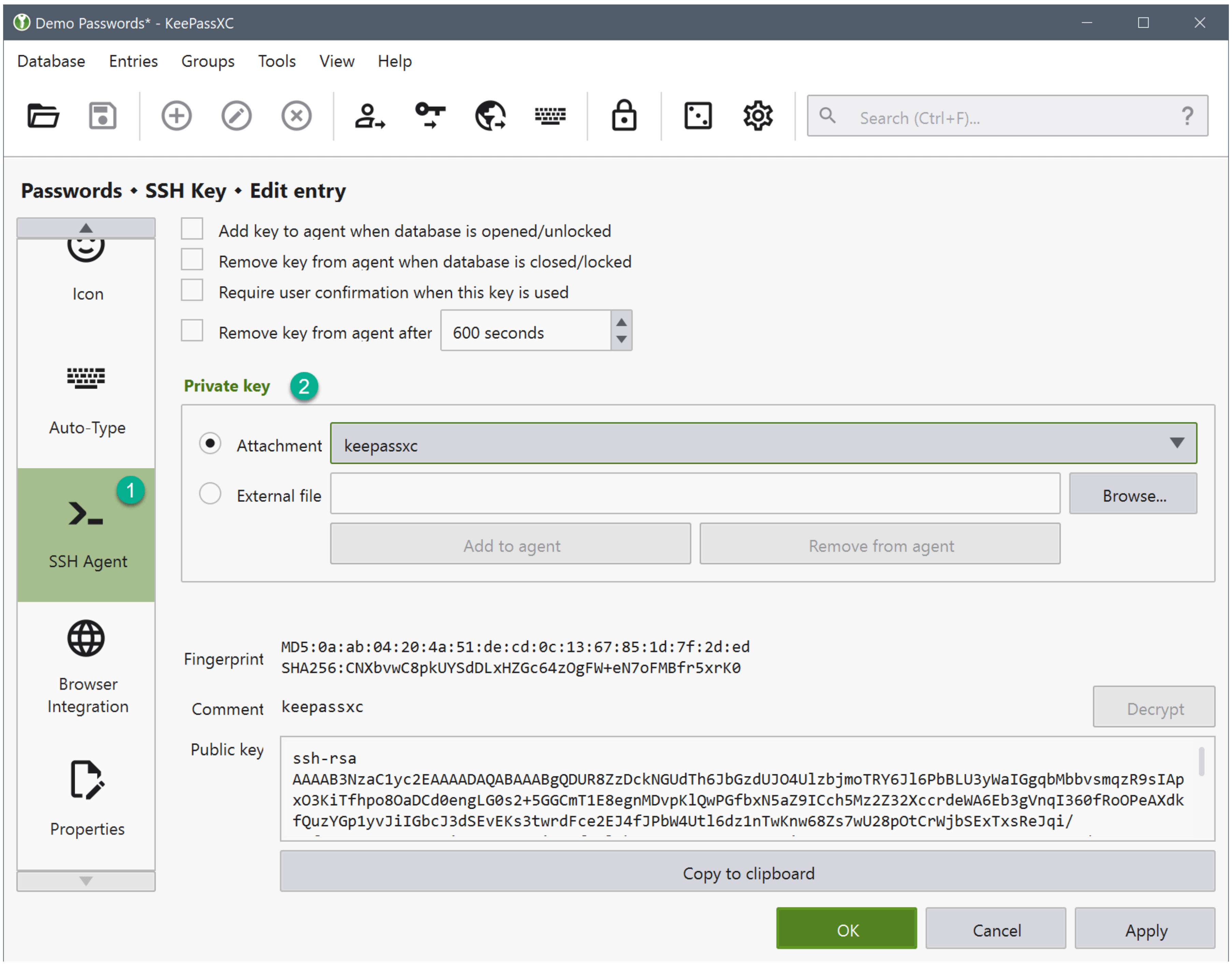
3. Go to the advanced category and attach the key file you generated previously.

4. Go to the SSH Agent category **(1)** and select the attachment from the list **(2)**.

5. Alternatively, you can load an external file dynamically using the file selection.

6. Choose the options for this key.

7. Press **OK** to accept the entry. Depending on the options you chose, KeePassXC will load the key and present it for use.



Passwords • SSH Key • Edit entry

□ Add key to agent when database is opened/unlocked

□ Remove key from agent when database is closed/locked

□ Require user confirmation when this key is used

□ Remove key from agent after 600 seconds

Private key

● Attachment keepassxc

○ External file

Fingerprint MD5:0a:ab:04:20:4a:51:de:cd:0c:13:67:85:1d:7f:2d:ed
SHA256:CNXbvwC8pkUYSdDLxHZGc64zOgFW+eN7oFMBfr5xrK0

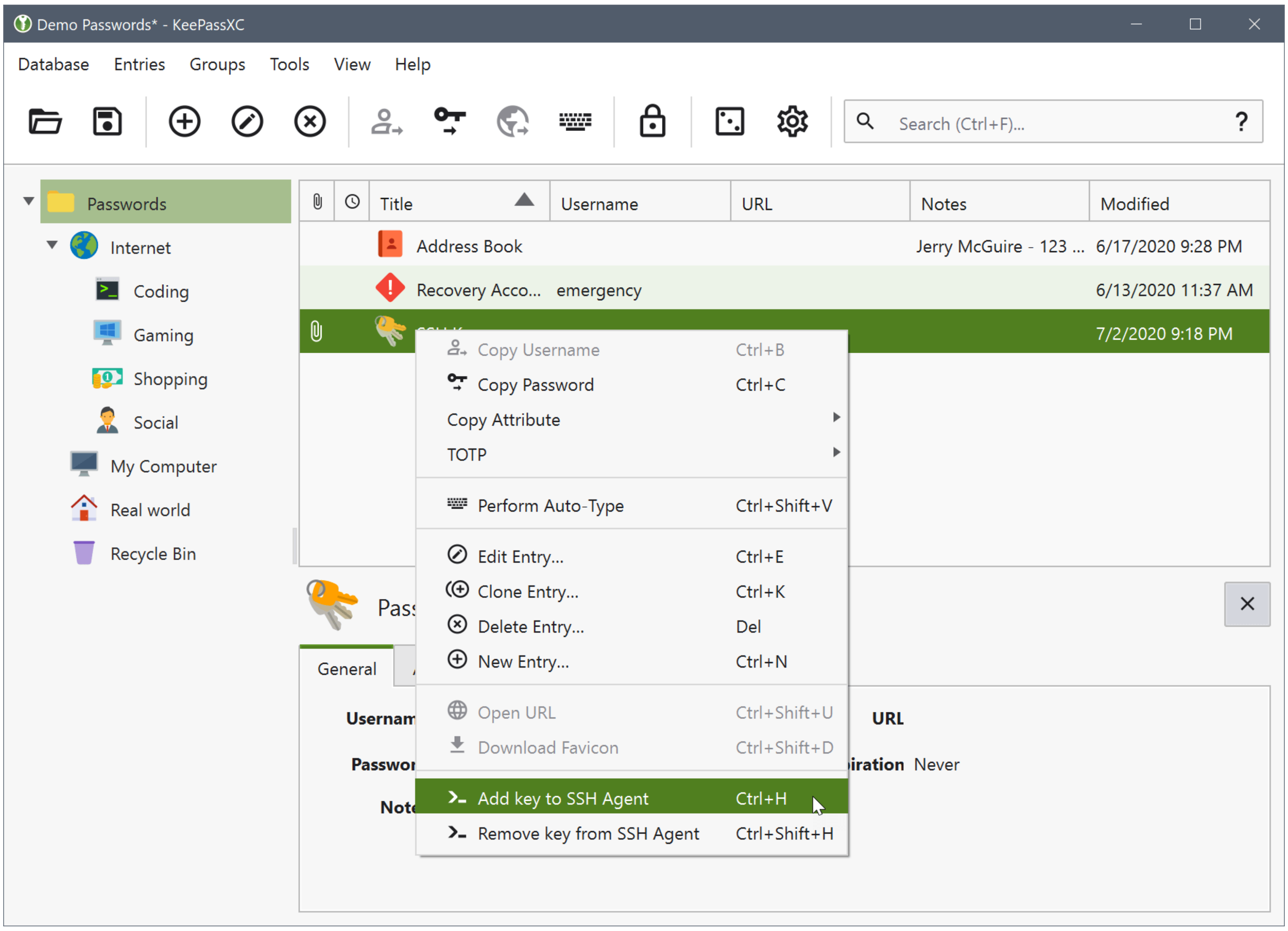
Comment keepassxc

Public key

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABgQDUR8ZzDckNGUdTh6JbGzdUJO4UlzbjmoTRY6Jl6PbBLU3yWaIGgqbMbbvsmqzR9sIAp
xO3KiTfhpo8OaDCd0engLG0s2+5GGCmT1E8egnMDvpKlQwPGfbxN5aZ9ICch5Mz2Z32XccrdeWA6Eb3gVnqI360fRoOPeAXdk
fQuzYGp1yvJiIGbcJ3dSEvEKs3twrdFce2EJ4fJPbW4Utl6dz1nTwKnw68Zs7wU28pOtCrWjbSExTxsReJqi/
```

Figure 3. SSH Agent Entry Settings Page

If you chose to not autoload the key on database unlock, you can manually make the key available by using the context menu from the entry list.



Title	Username	URL	Notes	Modified
Address Book			Jerry McGuire - 123 ...	6/17/2020 9:28 PM
Recovery Acco...	emergency			6/13/2020 11:37 AM
				7/2/2020 9:18 PM

Figure 4. SSH Agent Load Key from Context Menu