BISSANTZ

Umgang mit Passwörtern

Autor	Christoph Behling
Datum der Veröffentlichung	04.01.2019

Bissantz & Company GmbH
Nordring 98 90409 Nürnberg
T +49 911 935536 - 0
F +49 911 935536 - 10
service@bissantz.de

Geschäftsführer
Dr. Nicolas Bissantz
Michael Westphal

USt-IdNr. DE181064258
Steuer-Nr. 241/118/25048
Amtsgericht Nürnberg
HRB 15590

Sparkasse Nürnberg
IBAN DE88 7605 0101 0570 1792 91
BIC SSKNDE77XXX

www.bissantz.de

Abstract

Wir bekommen von Kunden oft sensible Zugangsdaten für ihre EDV-Systeme anvertraut, z. B. Windows-Anmeldungen oder Kennwörter für Netzwerke (VPN, WLAN etc.). Mit diesen Daten müssen wir sorgsam umgehen, kein Dritter soll diese einsehen oder gar verwenden können. Sich alle Passwörter im Kopf zu merken ist wegen der Vielzahl kaum möglich. Daher müssen wir sie irgendwo sicher hinterlegen können. Dieser Beitrag zeigt, wie mit einem Passwortmanager einfach und schnell sensible Informationen verschlüsselt gespeichert und wieder abgerufen werden können.

Umgang mit Kundenpasswörtern

1 Einleitung

Zugangsdaten für Kundensysteme legen wir in der Regel sicher in der TCB unter dem Kundenkey ab („OnlineAccess"). Das Berechtigungssystem der Datenbank schützt die Daten vor unerlaubtem Zugriff. Oft ist man jedoch unterwegs und ein Zugriff auf die zentrale Datenbank ist kompliziert oder gar nicht möglich. Dennoch benötigt man die dort abgelegten Daten z.B. für einen Zugang auf das Kundensystem.

Ein Tabu wäre es jetzt, für jedermann sichtbar eine Textdatei „GeheimePasswörter.doc" zu öffnen, unter den vielen Einträgen den richtigen Kunden zu suchen und sich so die Anmeldedaten zu beschaffen. Das wirkt nicht nur unprofessionell, es verstößt auch gegen die Sorgfaltspflicht.

Wie mit wenigen Schritten eine eigene, sichere Passwortverwaltung mit einem Passwortmanager eingerichtet werden kann, soll dieser Beitrag zeigen.

2 Programmauswahl, Installation und Konfiguration

Es gibt eine Vielzahl von Passwortmanagern. Eine Übersicht findet man im Internet z.B. hier. Ich habe mich nach einigen Tests für die Open Source Software KeePass entschieden[1] und möchte diesen Passwortmanager hier vorstellen.

Das Programm kann auf der Homepage heruntergeladen werden. Nach Ausführung des Setups (die Dialogboxen kann man einfach durchgeklicken) startet man das Programm und legt über „File", „New…" eine neue Datenbankdatei an (vgl. Abb. 1).

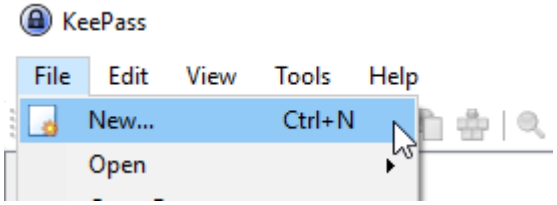

Abb. 1: Erstellen einer neuen Datenbankdatei in KeePass

Diese verschlüsselte Datei wird dann alle sensiblen Daten enthalten - wo die Datei abgelegt wird, ist zunächst Schritt egal. Sie kann später verschoben werden. In Kapitel 4.1 wird gezeigt, wie die Datei in einer Cloud-Umgebung verwendet werden kann.

Im nachfolgenden Eingabefenster (vgl. Abbildung 2) sind zwei Schritte nötig. Zum einen wird die Datei mit einem sicheren Passwort geschützt[2]. Dieses Passwort muss man sich gut merken, es wird bei jedem Öffnen der Datei abgefragt. Ohne das Passwort sind die gespeicherten Informationen nicht mehr zugänglich.

---

[1] Software made in Germany, der Entwickler Dominik Reichl wohnt in Metzingen.

[2] Wie gut ein Passwort ist, kann auf dieser Seite überprüft werden. Dieses Passwort danach *nicht* verwenden, die Verbindung ist nicht verschlüsselt.

Optional kann eine Schlüsseldatei zu der Datei erzeugt werden. Der Zugang zu den Informationen ist so doppelt geschützt. Die zusätzliche Sicherung ist vor allem dann empfehlenswert, wenn die .kdbx-Datei nicht lokal auf dem Rechner abgelegt wird.

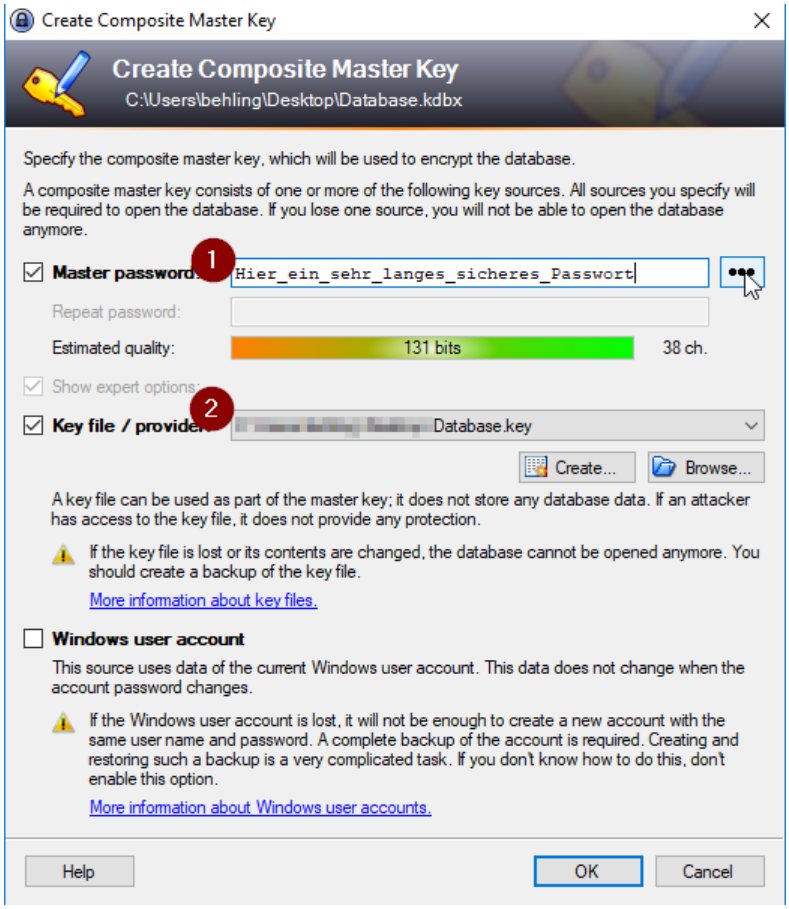

Abb. 2: Konfiguration der verschlüsselten Datenbankdatei

Nach dem Drücken der OK-Schaltfläche kann die Datenbank konfiguriert werden. Weitere Einstellungen sind hier erst einmal nicht nötig und könnten später auch noch geändert werden.

Ein Notfallblatt könnte nun erstellt werden – es wird für Geschäftspasswörter eigentlich aber nicht benötigt. So landet man, nachdem der nächste Dialog übergangen wurde, in der grafischen Benutzeroberfläche des Programms.

3 Anlegen, Speichern und Abrufen von Passwörtern

Zu sehen sind zwei Beispieleinträge – diese können ignoriert oder sogar gelöscht werden. Wir erstellen einen neuen Eintrag über Kontextmenü, „Add Entry…“ (vgl. Abb. 3).

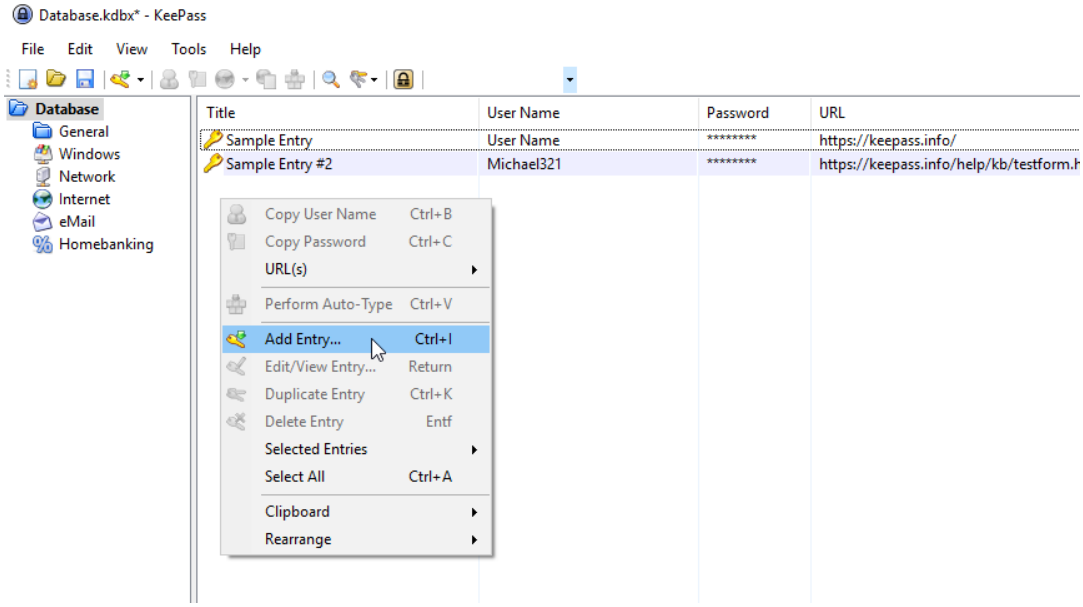

Abb. 3: Neuen Eintrag für ein Passwort erzeugen

Der neue Eintrag kann nun ein Titel bekommen und ein Benutzername kann eingetragen werden (vgl. Abb. 4). Das Passwort kann entweder eingetippt oder in die Zeile kopiert werden. Die Empfehlung ist, bei sehr einfachen Passwörtern diese durch sichere zu ersetzen. Hilfreich dabei ist ein integrierter Passwortgenerator (vgl. Abb. 4, blauer Pfeil).

Außerdem kann eine URL angegeben werden, was später bei der Firefox-Integration (vgl. Kapitel 4.2) gute Dienste leisten kann. In dem Kommentarfeld können noch Serverdaten, Ansprechpartner und andere Informationen des Zugangs hinterlegt werden.

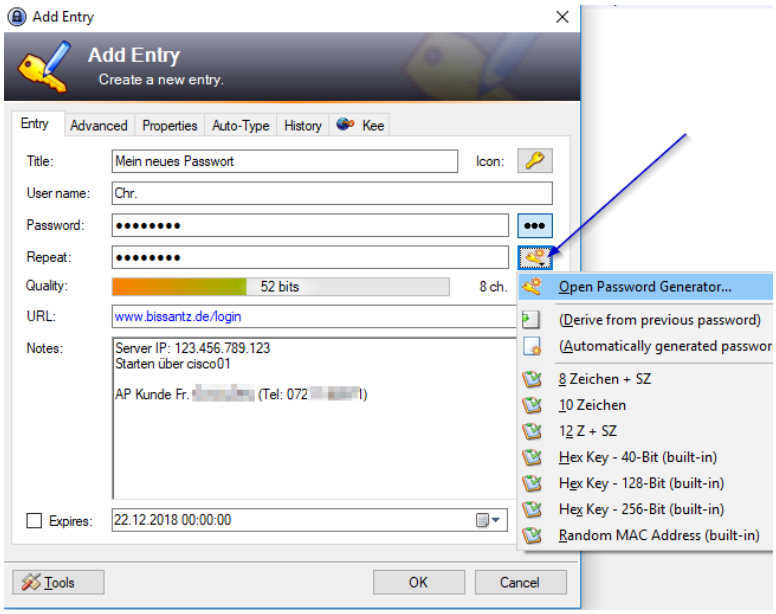

Abb. 4: Ausgefüllter Eintrag

Nach dem Drücken der OK-Taste ist der erste eigene Eintrag vorhanden.

Der Zugriff auf den Benutzernamen und das Passwort ist jetzt sehr einfach: Mit einem Doppelklick auf den benötigten Eintrag wird der Inhalt der Zeile in die Zwischenablage kopiert und kann einfach an der benötigten Stelle wieder eingefügt werden. So wird z.B. für eine Windows-Anmeldung zuerst der Benutzername, dann das Passwort in den Anmeldedialog übertragen; niemand sieht das Passwort, man muss es selbst auch gar nicht mehr wissen. Als kleine Zugabe wird die Zwischenablage automatisch von dem Programm nach 12 Sekunden geleert.

Das oben beschriebene Vorgehen ist die einfachste Einsatzvariante eines Passwortmanagers und reicht in vielen Fällen schon aus, Passwörter sicher abzulegen und unsichtbar wieder aufzurufen.

4 Weitere zusätzliche Einsatzmöglichkeiten

4.1 Ablage der Daten in der Cloud

Damit man auf die Datenbank von überall zugreifen kann und diese z.B. nach einem Festplattenabsturz wiederhergestellt werden kann, sollte man die .kdbx-Datei in einem Cloud-Speicher ablegen (z.B. OneDrive, Dropbox etc.). Zur Erinnerung: neben der Datenbankdatei wurde beim Anlegen zusätzlich eine Schlüsseldatei erstellt. Diese Datei wird natürlich *nicht* zusammen mit der Datendatei in der Cloud abgelegt, sondern auf dem lokalen Gerät gespeichert. Zusätzlich wird eine Sicherheitskopie auf einem anderen Speichermedium angelegt. Ohne die Schlüsseldatei ist die Datenbankdatei wertlos und kann problemlos in der Datenwolke liegen.

Jeder neue Eintrag wird beim Speichern mit der Cloud synchronisiert und somit auch eine perfekte Sicherungskopie erzeugt. Zudem kann man mit weiteren (mobilen) Geräten auf die Datenbank zugreifen (Empfehlung für Android: KeePass2Android[3]; für iOS gibt es auch Apps).

4.2 Browserintegration

Auch eine Browserintegration von KeePass ist möglich. Ich verwende Kee[4] für den Firefox-Browser. Wenn im Datensatz eine URL hinterlegt ist, verbindet er sich bei einer Passwortabfrage im Browser mit der (vorher entsperrten) KeePass-Datenbank und schreibt automatisch die Anmeldeinformationen in den Dialog (vgl. Abb. 5). Sicherer und komfortabler kann man eigentlich nicht arbeiten.

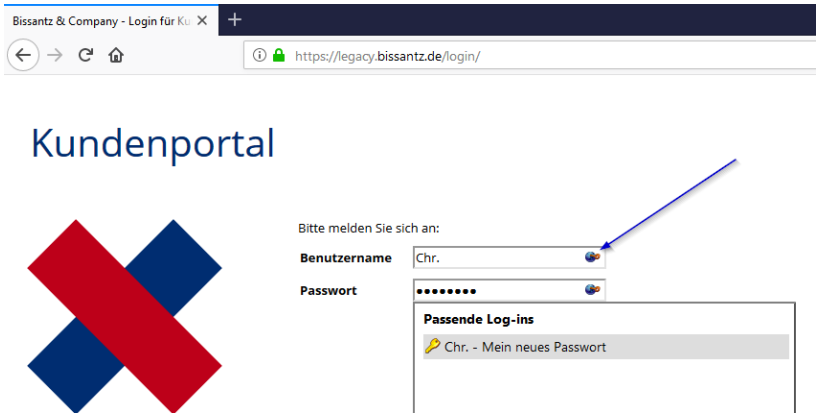

Abb. 5: Beispiel einer Browserintegration

[3] Software made in Germany, Entwicklers Philipp Crocoll, Karlsruhe.

[4] Englischer Entwickler aus Bristol, Chris Tomlinson.