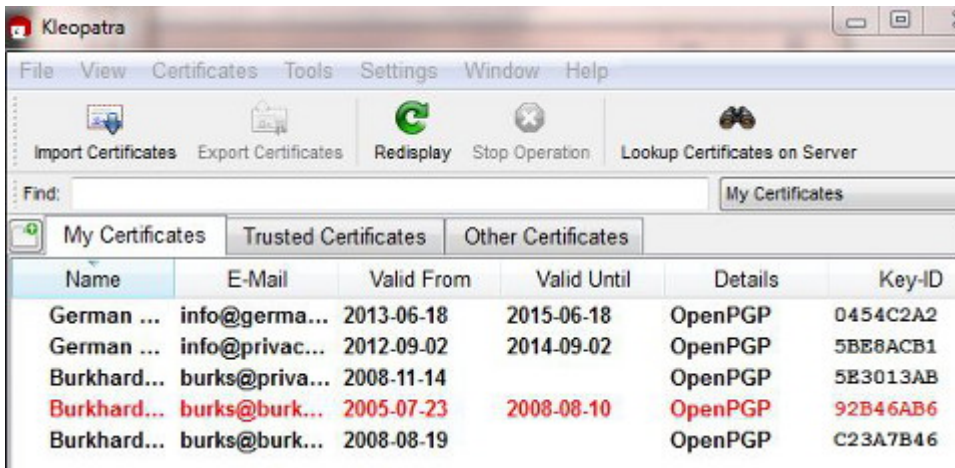


Kleopatra sei Dank!



Hurra, nach stundenlangem Dröseln und Rätselfn habe ich [mein APG-Problem](#) gelöst und kann jetzt auch E-Mails auf meinem Android-Smartphone verschlüsseln. Es gab zwei Probleme:

Erstens: Wie einige kundige Leser schon vermuteten, ist das Format, das [Enigmail](#) (das OpenPGP-Add-on für Thunderbird) zum Export der Schlüssel benutzt, offenbar nicht kompatibel mit der Dateiverwaltung von Android, obwohl der Schlüssel korrekt im ASCII-Format erscheint. Da muss man erst mal drauf kommen. Also [genau das](#), was hier ~~in Klingonisch~~ erwähnt wird:

APG cannot import keys in .asc files terminated with CR/LF (PC-style): it requires LF only (UNIX style). If your PC editor offers you a choice (e.g., TextPad does) save the file as UNIX text.

Ich habe es also mit [Kleopatra](#) versucht und damit die öffentlichen und meine geheimen Schlüssel exportiert und auf das Smartphone gebeamt.

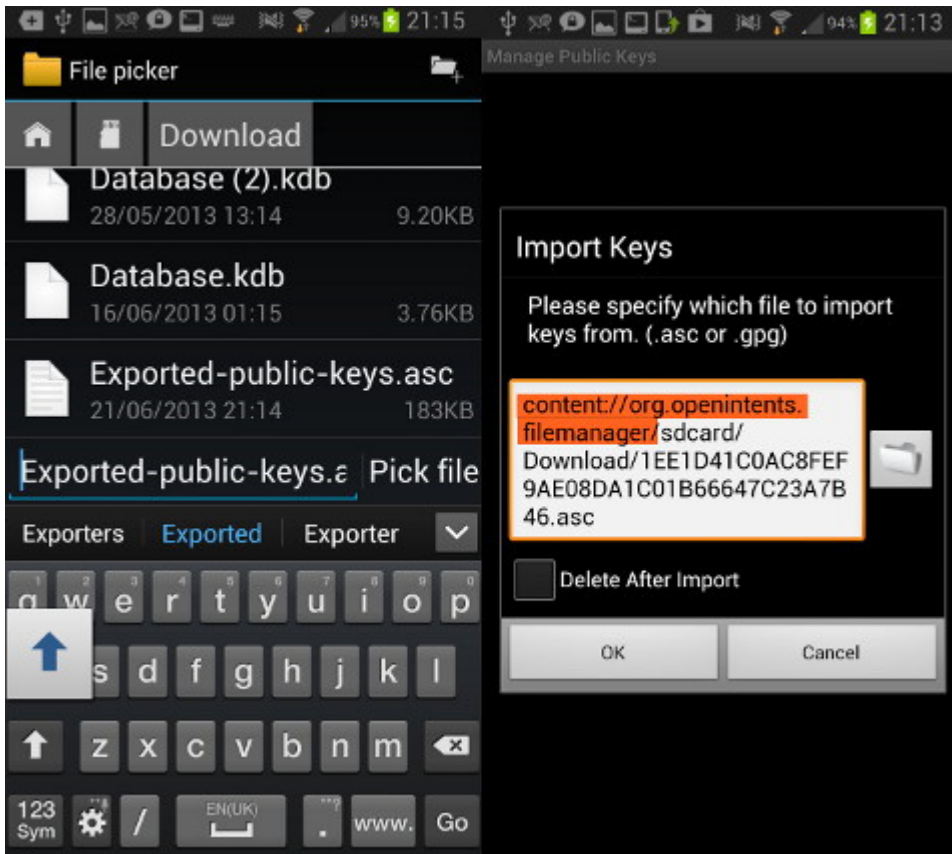
Nun zu uns, Programmierer! Wir sind ja alle dankbar, dass ihr uns das Schöne, Gute und Wahre Open Source und gratis usw. zusammendröselt und zum Downlad anbietet. Ihr habt aber trotzdem einen Knall: Welche Pappnase denkt sich für jeden Softwareschnipsel einen anderen Namen aus? Kleopatra, „ist der bevorzugte Zertifikatsmanager in Gpg4win“. Geht's noch? Sind

wir hier im Gallischen Krieg oder beim großen Latinum? Und das nur, weil ihr Leute abschrecken wollt, das auch zu nutzen?

Was kriegt der Laie in den ersten fünf Minuten alles zu sehen: PGP, OpenPGP, GnuPG, Gpg4Win, GPG, OpenPGP & S/MIME, Algorithmenstärken, öffentliche und private Schlüssel, und jetzt auch noch Kleopatra – glaubt ihr denn, auch nur ein wohlwollender Nutzer und eine geneigte Nutzerin hat da noch Lust, überhaupt anzufangen? Und welche Knalltüte nennt die Schlüssel bei Kleopatra „Zertifikate“? Sogar ich musste erst überlegen und probieren, und ich verschlüssele meine E-Mails seit 1995.

Ich glaube, ich sollte mich mal selbst hinsetzen, der breiten Masse aufs Maul schauen und die Angelegenheit volkstümlich formulieren. So wird das nix.

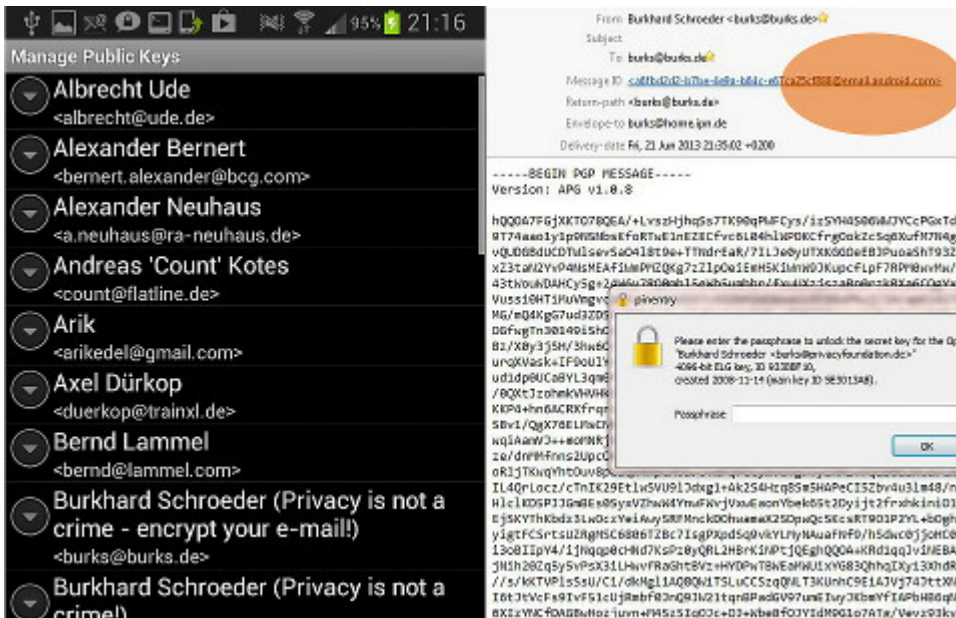
Wie [sagte Paulus](#) über das erfolgreiche Missionieren ganz richtig: „Den Juden bin ich geworden wie ein Jude, auf daß ich die Juden gewinne. Denen, die unter dem Gesetz sind, bin ich geworden wie unter dem Gesetz, auf daß ich die, so unter dem Gesetz sind, gewinne. Denen, die ohne Gesetz sind, bin ich wie ohne Gesetz geworden, auf daß ich die, so ohne Gesetz sind, gewinne. Den Schwachen bin ich geworden wie ein Schwacher, auf daß ich die Schwachen gewinne. Ich bin jedermann allerlei geworden, auf daß ich allenthalben ja etliche selig mache. Den DAUS bin ich geworden ein Dau, auf daß sie ihre E-Mails verschlüsseln lernen und ich sie selig mache.“



Zweitens: Auch der Dateimanager (ich nutze den [OI File Manager](#)) zickt herum.

Auf der linken Seite des oberen Screenshots sind die verschiedenen Dateien im Download-Verzeichnis zur Auswahl zu sehen. Wenn man eine ausgesucht hat und auf „pick file“ drückt, erscheint eben dieselbe, was auf der rechten Seite deutlich wird – hier will ich eine .asc-Datei (so hat Kleopatra meinen öffentlichen Schlüssel genannt) importieren. Niemand sagte mir aber, dass man per Hand (!) zunächst den vorderen Teil des Pfades (alles, was rot markiert ist), eliminieren muss, ansonsten kommt wieder die Meldung „file not found“.

Ich hatte nur so eine vage Ahnung, nachdem ich praktisch alle Forenbeiträge zum Thema der letzten drei Jahre gelesen habe. Das sind gar nicht so viele, und alle natürlich in Englisch. Ich habe einfach herumprobiert. Das kann man niemandem, der das lernen will, zumuten.



Der Verschlüsseln ist auch nicht einfach, und APG stürzt manchmal ab, warum, werde ich herausfinden. Aber immerhin habe ich alle öffentlichen Schlüssel importiert und es geschafft, mir selbst eine verschlüsselte E-Mail zu schicken, die ich mit einem meiner Rechner dann lesen konnte (MessageID email.android.com, oben rechts rot markiert)

Die nächste Aufgabe ist herauszufinden, wo APG die Schlüssel speichert. Ich habe mir für den Fall der Fälle, dass ich sie verstecken will, [EDS](#) installiert.