

Android mit Google-Befreiung, PlayStore-Apps im eigenen Repo

Dieser Workshop zeigt, wie man seine Android-Geräte ohne Google-Konto betreiben kann, ohne auf Apps aus dem Google PlayStore zu verzichten. Er richtet sich an Anwender, die auf Datensparsamkeit achten und Bedenken bezüglich der Verwendung und Weitergabe personenbezogener Daten im Internet haben.

Eines der verbreitetsten Betriebssysteme für Smartphones ist Android, ein von Google speziell angepasstes Linux. Wird das Gerät initial eingerichtet, fordert es dazu auf, ein Google-Konto anzulegen oder ein bestehendes zu nutzen. Das hat den Vorteil, dass man sofort Zugriff auf den Google PlayStore bekommt. Aber es werden ab diesem Zeitpunkt auch eine Reihe von Daten mit Servern bei Google, einem amerikanischen Unternehmen, ausgetauscht. Hierzu zählen Browser und Standortdaten, WLAN-Passwörter, Kalender, Kontakte, Fotos usw. Manches lässt sich abschalten, aber es kommen durch Updates auch neue Verbindungen hinzu oder werden reaktiviert.

Um dem effektiv entgegen zu wirken kann man versuchen, für die verschiedenen Aufgaben alternative Software zu verwenden. Dabei ist es möglich auf ausschließlich freie Software zurückzugreifen und den Google PlayStore nur in Ausnahmefällen zu benutzen. Letztendlich am wirkungsvollsten ist aber, das Gerät ganz ohne Google-Konto zu betreiben und nach Möglichkeit selbst auf die Google Play Services zu verzichten (Erfordert alternative Mods wie Cyanogenmod). Wer dennoch nicht auf Google PlayStore Apps verzichten kann braucht eine Möglichkeit, ohne Google-Konto Zugriff auf den Appstore zu nehmen.

Im Workshop wird eine Lösung vorgestellt und umgesetzt, mit der man mithilfe eines RaspberryPi und eines ausrangierten Smartphones ein Setup herstellt das genau dieses leistet. Mithilfe der freien Software F-Droid wird ein eigenes Repository erzeugt. Die Installationspakete des ausrangierten Smartphones (mit Google-Konto) werden extrahiert und in dem Repository zu Verfügung gestellt. Die Synchronisation erfolgt automatisiert. Das aktuelle Smartphone bekommt dann alle vorbereiteten Apps zur Installation angeboten und Updates werden automatisch angezeigt. Am Ende kann man bei Bedarf seinen Webserver auf dem RaspberryPi über das heimische DSL verfügbar machen und hat damit seinen eigenen Appstore überall dabei.