

# OBTAINIUM WIKI

Das Hauptziel von Obtainium ist es, den Prozess des Herunterladens und Installierens von Android-App-Updates direkt von ihren "Quell"-Websites (Websites, auf denen App-Dateien zum direkten Download verfügbar sind) zu automatisieren. Dieser Prozess muss automatisiert werden, da die Benutzer möglicherweise nicht bereit oder in der Lage sind, sich auf einen App-Store zu verlassen, um eine bestimmte App zu aktualisieren, aber Android-Apps (im Gegensatz zu PC-Apps) gehen in der Regel davon aus, dass sie extern durch einen App-Store aktualisiert werden, und verfügen daher nicht über eine eingebaute Selbstaktualisierungsfunktion.

Dieses Konzept ist zwar einfach, aber die große Vielfalt an unterstützten Quellen, Benutzereinstellungen und APK-Benennungs-, Versions- und Verteilungsmethoden machen die Dinge komplizierter. Dieses Wiki erklärt die verschiedenen in Obtainium verfügbaren App-Quellen und Einstellungen. Es ist nicht vollständig und möglicherweise nicht auf dem neuesten Stand.

## A. Übersicht User Interface (UI)

Die Registerkartenleiste am unteren Rand des Bildschirms ist das Hauptinstrument zur Navigation in Obtainium.

### 1. Apps Seite

Dies ist der Hauptbildschirm; er enthält eine Liste der Apps, die von Obtainium verfolgt werden, und liefert die grundlegenden Informationen zu jeder App.

Auf dieser Seite können Sie mit den Schaltflächen am unteren Rand des Bildschirms verschiedene Vorgänge (wie Löschen, Aktualisieren, Markieren usw.) für mehrere Apps auf einmal durchführen. Sie können dies tun, nachdem Sie eine oder mehrere Apps ausgewählt haben, entweder durch langes Drücken, um in den Mehrfachauswahlmodus zu gelangen, oder über die Schaltfläche "Alle auswählen".

Sie können auch nach bestimmten Apps filtern, indem Sie auf die Schaltfläche Filter klicken. Auf diese Weise können Sie nur die Apps anzeigen, die bestimmte Kriterien erfüllen (z. B. installiert/nicht installiert, aktuell/veraltet, Quellseite usw.).

## 2. App-Seite hinzufügen

Auf dieser Seite können Sie eine Anwendung anhand ihrer Quell-URL hinzufügen.

Bei der Eingabe der Quell-URL erkennt Obtainium automatisch, welche [Quellenlogik](#) verwendet werden soll, und zeigt die entsprechenden Optionen an. Wenn eine URL keiner unterstützten Quelle entspricht, wird die "[HTML](#)"-Quelle als Standard ausgewählt. Diese Quelle ist ein allgemeiner Fallback, der in einigen Fällen funktionieren kann. Wenn Sie der Meinung sind, dass Obtainium die falsche Wahl getroffen hat, können Sie den zu verwendenden Quellentyp manuell festlegen.

Bei den meisten Quellen müssen Sie die URL nicht genau angeben. Die GitHub-Quelle akzeptiert beispielsweise jede GitHub-URL, die die Basis-URL des Projektarchivs enthält (z.B. <https://github.com/ImranR98/Obtainium/releases/latest> wird automatisch auf <https://github.com/ImranR98/Obtainium> gekürzt). Ihre URL muss jedoch präziser sein in zwei Situationen:

1. Bei Verwendung der HTML-Quelle
  - Zum Beispiel ist die Tor Android APK unter <https://www.torproject.org/download/> zu finden, also würde die Eingabe von <https://www.torproject.org/> nicht funktionieren.
2. Bei der manuellen Auswahl einer Quelle (die die HTML-Fallback-Auswahl von Obtainium außer Kraft setzt)

Auf dieser Seite können Sie auch nach Apps in allen Quellen suchen, die diese Funktion unterstützen (die Seite [Import/Export](#) bietet auch ein separates Suchwerkzeug). Nur weil eine App in den Suchergebnissen auftaucht, heißt das nicht, dass sie erfolgreich hinzugefügt wird - sie muss noch alle anderen Kriterien erfüllen.

Schließlich werden auf dieser Seite alle unterstützten Quellen aufgelistet, zusammen mit zusätzlichen Informationen, z. B. ob eine Quelle durchsuchbar ist.

### 3. Import/Export Seite

Auf dieser Seite können Sie Ihre Obtainium-Daten exportieren, damit sie später importiert werden können. Sie können auch automatische Exporte aktivieren, die immer dann erfolgen, wenn sich Daten ändern.

Auf dieser Seite finden Sie auch verschiedene andere Möglichkeiten, eine große Anzahl von Anwendungen zu importieren, unter anderem über die Suche.

Mit dem Suchwerkzeug auf dieser Seite können Sie auch Quellen durchsuchen, bei denen Sie den Quellhost/die Quelldomäne angeben müssen, z. B. F-Droid-Repos von Drittanbietern.

### 4. Einstellungen Seite

Diese Seite bietet verschiedene Einstellungen für die Benutzeroberfläche und das Verhalten, einschließlich der Möglichkeit, mehr Funktionen für bestimmte Quellen zu aktivieren, indem die erforderlichen Anmeldedaten hinzugefügt werden.

## B. Wie Apps nachverfolgt werden

### 1. Grundlagen

Wenn Sie eine App-URL zu Obtainium hinzufügen, müssen Sie eine [App-Quelle](#) auswählen. Quellen definieren, wie App-Informationen und APK-Dateien aus der von Ihnen eingegebenen URL extrahiert werden. In den meisten Fällen wählt Obtainium automatisch die entsprechende Quelle aus - wenn dies nicht möglich ist, wird ein Dropdown-Menü "Quelle überschreiben" angezeigt.

Eine App-Quelle muss mindestens die folgenden Daten für ihre Apps bereitstellen:

- i. **Die App-Version** (oder eine "Pseudo-Version" - eine Kennung, die sich bei jeder neuen Version der App ändert.)
- ii. **Mindestens eine APK-Download-URL**, die der zur Verfügung gestellten Version entspricht

App-Quellen können auch andere Informationen liefern - diese ermöglichen zusätzliche Funktionen oder UI-Vorteile. Zum Beispiel:

- Der Name des Autors der Anwendung
- Die Paket-ID der Anwendung
- Das Veröffentlichungsdatum der neuesten Version
- Infos zu früheren Versionen oder Varianten der App

In einer idealen Welt würde jede App-Quelle alle erforderlichen Informationen auf unkomplizierte Weise bereitstellen - mit einer einzigen App pro angegebener URL, die alle erforderlichen Informationen in einem Standardformat enthält. Dies ist jedoch oft nicht der Fall - es gibt viele verschiedene Arten, wie App-Freigaben gehandhabt werden, sogar von derselben Quelle, so dass es nicht möglich ist, einen festen Satz von Schritten zu haben, um sie alle zu behandeln. Aus diesem Grund werden Ihnen beim Hinzufügen einer App verschiedene zusätzliche Optionen angeboten, mit denen Sie die Art und Weise, wie die App-Informationen extrahiert werden, ändern können. Während die Standardeinstellungen für die meisten Apps funktionieren, sollten Sie diese Optionen kennen, um mit Sonderfällen umgehen zu können - mehr dazu im Abschnitt [App-Quellen](#) weiter unten.

**Hinweis:** Viele Filtereinstellungen in Obtainium (einschließlich vieler quellenspezifischer optionaler Filter) verwenden [reguläre Ausdrücke](#) - Sie sollten mit diesen vertraut sein.

## 2. Erkennung der Version

Wenn Obtainium eine App trackt, die aktuell installiert ist, holt es sich die Version der App von Android und vergleicht sie mit der von der Quelle bereitgestellten Versionszeichenfolge. Dann vergleicht es die beiden, um zu entscheiden, ob ein Update verfügbar ist oder ob sich der Installationsstatus der App geändert hat. Dieser Vergleich kann nur durchgeführt werden, wenn die beiden Versionen das gleiche Format haben, was nicht immer der Fall ist. Es kann zum Beispiel einer der folgenden Fälle eintreten:

1. [Obtainium](#) von GitHub:
  - Für Android gemeldete App-Version: 0.14.21
  - Von der Quelle gemeldete Version: v0.14.21-beta

2. [Cheogram](#) aus einer SourceHut-Instanz:
  - Für Android gemeldete App-Version: 2.12.8-2+free
  - Von der Quelle gemeldete Version: 2.12.8-2
  
3. [Tor](#) von der Tor-Website:
  - Für Android gemeldete App-Version: 102.2.1-Release (12.5.6)
  - Aus der Quelle gemeldete Version: keine  
(diese HTML-Quelle liefert keine Versionszeichenfolge, daher wird stattdessen ein URL-Hash als "Pseudoversion" verwendet)
  
4. [Quotable](#) von GitHub:
  - Für Android gemeldete App-Version: 1
  - Aus der Quelle gemeldete Version: v10

Obtainium speichert eine Liste von „Standardformaten“, die es für diesen Vergleich verwendet (wie **x.y.z** oder **x.y**). Wenn beide zu vergleichenden Versionen demselben Format entsprechen, wird der Vergleich durchgeführt. Wenn nicht, wird die Versionserkennung für diese Anwendung deaktiviert. In einigen Fällen entfernt Obtainium zusätzliche Teile aus dem Quelltext, wenn dies zu einer Standardversion führen würde (wie z.B. **v** und **-beta** aus Obtainium's **v0.14.21-beta**), dann kann es den Vergleich durchführen. Wir versuchen nie, Teile der „echten“, vom Betriebssystem bereitgestellten Version zu entfernen.

Dieses Stück Code definiert, wie die verschiedenen "Standard"-Formate erzeugt werden:

[https://github.com/ImranR98/Obtainium/blob/main/lib/providers/apps\\_provider.dart#L64](https://github.com/ImranR98/Obtainium/blob/main/lib/providers/apps_provider.dart#L64)

Es ist immer möglich, diesen Code zu erweitern, um Unterstützung für weitere Formate hinzuzufügen, aber das erfordert sorgfältige Überlegungen. Wenn Android zum Beispiel meldet, dass die Version einer installierten App **1.2** ist, die Quelle aber sagt, dass die letzte verfügbare Version dieser App **1.2-4** ist, sollten wir dann das **-4** entfernen und sagen, dass die beiden gleich sind (was bedeutet, dass kein Update verfügbar ist)? Das mag in manchen Kontexten in Ordnung sein (wo die **-4** nicht wirklich auf eine Änderung in der Anwendung selbst hinweist), aber nicht in anderen Kontexten. Es wäre also keine gute Idee, diesen speziellen Fall zu unterstützen.

Die Deaktivierung der Versionserkennung sollte in der Regel keine nennenswerten Auswirkungen auf die alltägliche Nutzung haben. Wenn die Versionserkennung für eine Anwendung deaktiviert ist, kann es gelegentlich zu Inkonsistenzen zwischen der tatsächlich auf Ihrem System installierten Version der Anwendung und der in der Obtainium-Benutzeroberfläche angezeigten Version kommen. Dies sollte nur in zwei Fällen vorkommen:

1. Wenn sich die Version einer App aufgrund von Aktionen außerhalb von Obtainium ändert (z.B. wenn sie von Google Play aktualisiert wird)
2. Wenn ein Versuch von Obtainium, [die Anwendung im Hintergrund zu aktualisieren](#), fehlschlägt

In solchen Fällen wäre Obtainium nicht in der Lage zu erkennen, dass sich die tatsächliche Betriebssystemversion der Anwendung geändert hat, und würde daher seine internen Einträge nicht entsprechend aktualisieren - Sie müssten die Inkonsistenz manuell korrigieren.

Siehe <https://github.com/ImranR98/Obtainium/issues/946#issuecomment-1741745587>

### 3. Hintergrund-Updates

Obtainium sucht regelmäßig im Hintergrund nach App-Updates. Sie können die Häufigkeit dieser Aktualisierungsaufgaben auf der Einstellungsseite steuern.

Nach Abschluss der Prüfung einer Hintergrundaktualisierung werden alle verfügbaren Aktualisierungen in 2 Kategorien unterteilt:

1. Aktualisierungen, die im Hintergrund durchgeführt werden können
2. Aktualisierungen, die nicht im Hintergrund durchgeführt werden können

Damit ein Update automatisch im Hintergrund installiert werden kann (auch bekannt als stilles Update), müssen bestimmte Kriterien erfüllt sein:

- Das Betriebssystem muss Android 12 oder höher sein
- Die zu installierende Anwendung muss auf den [jüngsten Android-API-Level](#) ausgerichtet sein
- Die aktuell installierte Version der Anwendung muss von Obtainium installiert worden sein

- Sie müssen die Hintergrundaktualisierung in Obtainium aktiviert haben (sowohl allgemein als auch für diese Anwendung im Besonderen - dies ist die Standardeinstellung)
- Wenn mehrere APKs für das Update zur Verfügung stehen, müssen die zusätzlichen Optionen für diese App so konfiguriert werden, dass Obtainium diese auf eine APK herausfiltern kann

Jedes verfügbare Update wird heruntergeladen und, wenn möglich, installiert, und der Benutzer wird dann entweder über die Verfügbarkeit des Updates oder darüber informiert, dass es im Hintergrund installiert wurde.

Beachten Sie, dass Hintergrundaktualisierungen aufgrund technischer Beschränkungen nur asynchron und nach bestem Bemühen installiert werden können. Wenn also eine Hintergrundaktualisierung nicht installiert werden kann, werden Sie nicht über den Fehler informiert.

## 4. App-Quellen

Die Art und Weise, wie eine App hinzugefügt, auf Aktualisierungen geprüft und installiert wird, hängt von der Quelle und den vom Benutzer für diese App konfigurierten Einstellungen ab.

Die folgenden Optionen sind für alle Anwendungen unabhängig von der Quelle verfügbar:

- **Nur verfolgen:** Wenn Sie diese Option aktivieren, können Sie Aktualisierungsbenachrichtigungen für Apps erhalten, ohne zu versuchen, deren APKs herunterzuladen. Dies ist nützlich, um Updates für Apps zu verfolgen, für die keine APKs verfügbar sind oder die nicht einfach von Obtainium extrahiert werden können. Einige Quellen sind ausschließlich track-only - zum Beispiel [ApkMirror](#). Beachten Sie, dass [die Versionserkennung](#) nicht für Apps funktioniert, die als "Track-only" hinzugefügt wurden.
- **Versionsermittlung:** Siehe den Abschnitt zur [Versionserkennung](#).
- **APKs nach regulären Ausdrücken filtern:** Wenn mehr als eine APK für eine App- Veröffentlichung verfügbar ist, wird der Benutzer aufgefordert, eine manuell auszuwählen. Dies ist nicht nur unbequem, sondern bedeutet auch, dass die App nicht im Hintergrund aktualisiert wird, auch wenn dies sonst möglich gewesen wäre. Mit dieser Option kann der Benutzer einen regulären Ausdruck angeben, mit dem APK herausgefiltert werden können, die nicht darauf passen (idealerweise bis auf eine Option).

- **Versuchen Sie, APKs nach CPU-Architektur zu filtern, wenn möglich:** Mit dieser Option wird Obtainium angewiesen, automatisch alle APKs herauszufiltern, die nicht mit dem Gerät des Benutzers kompatibel zu sein scheinen. Die dafür verwendete Logik ist sehr einfach und basiert auf dem APK-Dateinamen. Sie ist daher nicht immer zuverlässig und der Benutzer muss in vielen Fällen immer noch selbst filtern.
- **App-Name:** Hier kann der Benutzer seinen eigenen Namen für die App angeben, anstatt den von Obtainium extrahierten zu verwenden.
- **Von Hintergrundaktualisierungen ausnehmen:** Mit dieser Option wird verhindert, dass die Anwendung im Hintergrund aktualisiert wird, selbst wenn Hintergrundaktualisierungen aktiviert sind und die Anwendung ansonsten alle Kriterien für eine Aktualisierung im Hintergrund erfüllt hätte.
- **Update-Benachrichtigungen überspringen:** Mit dieser Option wird Obtainium angewiesen, den Benutzer nicht zu benachrichtigen, dass für diese App ein Update verfügbar ist oder dass sie im Hintergrund aktualisiert wurde.

Abgesehen davon hat jede Anwendung zusätzliche quellenspezifische Optionen. Die meisten dieser Optionen sind selbsterklärend und werden häufig aktualisiert, so dass sie in diesem Wiki nicht behandelt werden. Einige Quellen haben jedoch einzigartige Verhaltensweisen, die einer genaueren Erläuterung bedürfen und die im Folgenden behandelt werden.

## 5. GitHub

GitHub legt eine Obergrenze für die Anzahl der API-Anfragen fest, die Sie innerhalb eines bestimmten Zeitraums stellen können. Da Obtainium die GitHub-API nutzt, um Versionsinformationen abzurufen, kann es zu einem "Rate Limit"-Fehler kommen, wenn Sie mehr als ein paar Dutzend GitHub-Apps haben. Sie können dies umgehen, indem Sie ein [persönliches Zugriffstoken](#) von GitHub erhalten und es in den Einstellungen von Obtainium speichern (das Token sollte nur die minimale Leseberechtigung benötigen).

GitHub ermöglicht es Entwicklern auch, mehrere Versionen ihrer Anwendung zu hosten. Dies bedeutet in der Regel ältere Versionen derselben Anwendung, kann aber auch Vorabversionen, Varianten usw. umfassen. - Obtainium bietet daher verschiedene Filter, mit denen Sie durch diese navigieren und genau die Versionen auswählen können, an denen Sie interessiert sind.



## 6. GitLab

---

GitLab-Veröffentlichungen enthalten manchmal APKs, die auf nicht standardisierte Weise angehängt sind, so dass Obtainium nicht einfach an sie herankommt ([siehe diesen Kommentar für eine genauere Erklärung](#)). Die GitLab-API bietet einen weitaus zuverlässigeren Weg, APKs zu extrahieren, aber sie kann nicht ohne einen API-Schlüssel. Während dies für die meisten GitLab-Repos kein Problem darstellen sollte, können Sie in den Einstellungen von Obtainium [Ihr eigenes persönliches Zugriffstoken](#) hinzufügen, um die APK-Extraktion in den Fällen zuverlässiger zu gestalten, in denen sich dies als Problem erweist (der minimale `read_api`-Bereich sollte ausreichend sein).

Wie GitHub ermöglicht auch GitLab Entwicklern das Hosten älterer Versionen derselben App, sodass zusätzliche Optionen zur Verfügung stehen.

## 7. F-Droid Drittanbieter Repo

---

Im Gegensatz zu den meisten anderen Quellen enthalten die F-Droid-Repos Informationen über mehrere Anwendungen, die alle unter der gleichen URL zu finden sind. Das bedeutet, dass Sie zusätzlich zur Repository-URL den Namen oder die ID der gewünschten App separat angeben müssen. Wenn Sie nicht sicher sind, welche Apps in einem bestimmten Repository verfügbar sind, können Sie die Schaltfläche „Quelle suchen“ auf der [Import/Export-Seite](#) verwenden, um dies herauszufinden.

Beachten Sie, dass Obtainium nicht automatisch erkennen kann, dass eine bestimmte URL auf ein F-Droid-Repo eines Drittanbieters verweist. Das bedeutet, dass das Hinzufügen eines F-Droid-Repos eines Drittanbieters zu Obtainium standardmäßig zu einer falschen Verwendung der [HTML-Quelle](#) führt (eine Fallback-Quelle, die Obtainium für jede URL verwendet, die es nicht erkennt). Sie müssen manuell "F-Droid Third Party Repo" aus dem "Override Source" Dropdown Menü auswählen, um dies zu umgehen.

## 8. APKMirror

APKMirror ist eine "Track-Only"-Quelle. Das bedeutet, dass Sie zwar eine APKMirror-URL zu Obtainium für Update-Benachrichtigungen hinzufügen können, aber nicht in der Lage sein werden, die App tatsächlich herunterzuladen und zu installieren. Das liegt daran, dass die Betreuer von APKMirror dies nicht zulassen (siehe [Problem #44](#)).

## 9. HTML

Die "HTML"-Quelle ist eine Fallback-Option für alle App-Quellen, die nicht ausdrücklich von Obtainium unterstützt werden. Aufgrund ihrer Flexibilität ist sie auch eine Möglichkeit, weniger populäre Nischenquellen zu unterstützen, ohne die App aufzublähen.

Die HTML-Quelle funktioniert folgendermaßen:

1. Es sendet eine Anfrage an die vom Benutzer angegebene Quell-URL und parst die Antwort als HTML. Anschließend sucht es nach Links auf der Seite.
2. Es filtert bestimmte Links heraus. Standardmäßig ist dies jeder Link, der nicht auf .apk endet, aber Sie können den „Custom APK Link Filter“ verwenden, um Ihren eigenen Filter festzulegen.
3. Es sortiert die verbleibenden Links. Diese Sortierung ist eine alphanumerische Sortierung für den gesamten Link, aber Sie können auch nur nach dem letzten Segment des Links sortieren. Dieses letzte Segment ist normalerweise der Dateiname, aber möglicherweise nicht, wenn Sie in Schritt 2 Ihren eigenen Filter verwendet haben.
4. Es wendet einen weiteren optionalen benutzerdefinierten Filter auf alle verbleibenden Links an. Der Unterschied zwischen diesem Filter und dem aus Schritt 2 ist, dass es sich hierbei um einen allgemeineren Filter handelt, den alle Quellen haben - er wird von der übergeordneten AppSource-Klasse geerbt (die APK-Filteroption, die in [App-Quellen](#) beschrieben wird). Es ist wahrscheinlich einfacher, dies in einigen Situationen zu verwenden, als einen einzelnen komplizierteren regulären Ausdruck in Schritt 2 zu haben ([siehe diesen Kommentar für ein Beispiel](#)). Der Filter ist auch nützlich, wenn er in Kombination mit der Option Zwischenverknüpfung (intermediate link option) verwendet wird.
5. Von den verbleibenden Links wird der erste ausgewählt (oder der letzte, wenn Sie die Umkehroption aktiviert haben).
6. Da wir nun den endgültigen APK-Link haben, benötigen wir eine eindeutige Release-ID, damit wir wissen, dass ein Update für die App verfügbar ist, wenn sich die ID ändert. Bei anderen Quellen ist die eindeutige Versions-

ID die App-Version, aber bei der HTML-Quelle ist es möglicherweise nicht möglich, einen Versions-String zu extrahieren. Daher ist dies standardmäßig nur ein Hash des Links.

- Allerdings sind in Links oft Versionsstrings eingebettet. Da Obtainium nicht weiß, wie es diese selbst extrahieren kann (verschiedene Websites haben unterschiedliche Methoden dafür), kann der Benutzer einen regulären Ausdruck angeben, der auf den Link angewendet werden kann, um die Version zu extrahieren - dafür ist das Feld „Version Extraction“ gedacht.
- Oftmals ist es jedoch schwierig, einen regulären Ausdruck zu finden, der genau auf die Version passt und gleichzeitig zusätzliche Zeichen ausschließt. Hierfür gibt es die Option "Gruppe abgleichen", mit der der Benutzer angeben kann, welche Gruppe im regulären Ausdruck als Version verwendet werden soll.
- Die Funktion zum Extrahieren der Version ist nicht wirklich notwendig - die Verwendung von Link-Hashes ist einfacher und zuverlässiger. Einige Benutzer möchten sie vielleicht nur haben, weil die echte Version schöner/genauer aussieht und es Obtainium in den meisten Fällen erlaubt, die [Versionserkennung](#) zu verwenden.

Wenn der Filter "Zwischenlink" verwendet wird, funktioniert die HTML-Quelle wie folgt (siehe [Ausgabe #820](#) für eine Situation, in der dies nützlich ist):

1. Es sucht nach Links auf einer HTML-Seite.
2. Er filtert alle Links heraus, die nicht dem Zwischenlinkfilter entsprechen.
3. Er greift den ersten verbleibenden Link auf (hier gibt es keine Umkehrmöglichkeit) und gibt diesen Link als Eingabe für den zuvor beschriebenen normalen HTML- Quellprozess ein.

## 10. Anmerkungen zu verschiedenen anderen Quellen

- **HTML:** Beachten Sie, dass der HTML-Quelltext Standard-Anfrage-Header enthält, die für die meisten Websites geeignet sein sollten. In einigen Fällen (z. B. [SourceForge](#)) müssen Sie sie löschen (und möglicherweise Ihre eigenen angeben).

- **Codeberg:** Diese Quelle ist in ihren zusätzlichen Optionen fast identisch mit GitHub.
- **F-Droid:** Jede App von F-Droid ist wahrscheinlich mit einem anderen Schlüssel [signiert](#) als die Versionen der gleichen App aus anderen Quellen. Dies bedeutet, dass der Versuch, von einer F-Droid-Version einer bestimmten App auf eine Version aus einer anderen Quelle (z. B. GitHub) zu aktualisieren, wahrscheinlich fehlschlagen wird.
- **Jede Quelle, der kein bestimmter Host zugeordnet ist** (wie [F-Droid-Repos](#) von Drittanbietern, Jenkins-Instanzen und SourceHut-Instanzen), wird von Obtainium nicht automatisch erkannt. Sie müssen die richtige Quelle manuell aus dem Dropdown-Menü "Override Source" auswählen.
- **Einige Quellen (wie APKPure)** bieten möglicherweise [XAPK-Dateien](#) anstelle von APK-Dateien an. Die XAPK-Unterstützung von Obtainium ist unvollständig und funktioniert möglicherweise nicht zuverlässig.