

# Das Protokoll des Brew-Upgrades

==> Abschließende Schritte ln -s ../../Cellar/azure-cli/2.68.0/etc/bash\_completion.d/az az ln -s ../../Cellar/azure-cli/2.68.0/bin/az az ln -s ../../Cellar/azure-cli/2.68.0/share/fish/vendor\_completions.d/az.fish az.fish ln -s ../../Cellar/azure-cli/2.68.0/share/zsh/site-functions/\_az\_az ==> Hinweise zsh-Vervollständigungen wurden installiert unter: /opt/homebrew/share/zsh/site-functions ==> Zusammenfassung /opt/homebrew/Cellar/azure-cli/2.68.0: 24.507 Dateien, 580,4MB ==> Ausführung von brew cleanup azure-cli...Entfernen: /opt/homebrew/Cellar/azure-cli/2.67.0\_1...(27.401 Dateien, 647,1MB) Entfernen: /Users/lzwjava/Library/Caches/Homebrew/azure-cli\_bottle\_manifest-2.67.0\_1...(22,5KB) Entfernen: /Users/lzwjava/Library/Caches/Homebrew/azure-cli-2.67.0\_1...(54MB) ==> Hinweise ==> openjdk Damit die System-Java-Wrapper dieses JDK finden, erstellen Sie einen symbolischen Link mit: sudo ln -sfn /opt/homebrew/opt/openjdk/libexec/openjdk.jdk /Library/Java/JavaVirtualMachines/openjdk.jdk

openjdk ist keg-only, d. h., es wurde nicht in /opt/homebrew verlinkt, da macOS ähnliche Software bereitstellt und die parallele Installation dieser Software zu verschiedenen Problemen führen kann.

Wenn openjdk zuerst in Ihrem PATH stehen soll, führen Sie folgenden Befehl aus: echo 'export PATH="/opt/homebrew/opt/openjdk/bin:\$PATH"' > ~/.zshrc

Damit Compiler openjdk finden, müssen Sie möglicherweise Folgendes setzen: export CPPFLAGS="-I/opt/homebrew/opt/openjdk/include" ==> ruby Standardmäßig werden von gem installierte Binärdateien in folgenden Ordner gelegt: /opt/homebrew/lib/ruby/gems/3.4.0/bin

Möglicherweise möchten Sie dies Ihrem PATH hinzufügen.

ruby ist keg-only, d. h., es wurde nicht in /opt/homebrew verlinkt, da macOS diese Software bereits bereitstellt und die Installation einer anderen Version parallel zu verschiedenen Problemen führen kann.

Wenn ruby zuerst in Ihrem PATH stehen soll, führen Sie folgenden Befehl aus: echo 'export PATH="/opt/homebrew/opt/ruby/bin:\$PATH"' > ~/.zshrc

Damit Compiler ruby finden, müssen Sie möglicherweise Folgendes setzen: export LDFLAGS="-L/opt/homebrew/opt/ruby/lib" export CPPFLAGS="-I/opt/homebrew/opt/ruby/include" ==> yt-dlp zsh-Vervollständigungen wurden installiert unter: /opt/homebrew/share/zsh/site-functions ==> redis Um redis nach einem Upgrade neu zu starten: brew services restart redis Oder, wenn Sie keinen Hintergrunddienst benötigen/wollen, können Sie einfach Folgendes ausführen: /opt/homebrew/opt/redis/bin/redis-server /opt/homebrew/etc/redis.conf ==> perl Standardmäßig werden nicht von brew verwaltete cpan-Module im Cellar installiert. Wenn Ihre Module über Updates hinweg erhalten bleiben sollen, empfehlen wir die Verwendung von local::lib.

Sie können dies folgendermaßen einrichten: PERL\_MM\_OPT="INSTALL\_BASE=\$HOME/perl5" cpanlocal :: lib Und fgen Sie Folgendes Ihren Shell – Profil hinzu, z.B. ./profile oder ~/.zshrc : eval "(perl -I\$HOME/perl5/lib/perl5 -Mlocal::lib = \$HOME/perl5)" ==> awscli Das Verzeichnis "examples" wurde installiert unter: /opt/homebrew/share/awscli/examples

zsh-Vervollständigungen und -Funktionen wurden installiert unter: /opt/homebrew/share/zsh/site-functions

==> php Um PHP in Apache zu aktivieren, fügen Sie Folgendes zu httpd.conf hinzu und starten Sie Apache neu: LoadModule php\_module /opt/homebrew/opt/php/lib/httpd/modules/libphp.so

```
<FilesMatch \.php$>
    SetHandler application/x-httpd-php
</FilesMatch>
```

Schließlich prüfen Sie, ob DirectoryIndex index.php enthält: DirectoryIndex index.php index.html

Die php.ini- und php-fpm.ini-Datei finden Sie unter: /opt/homebrew/etc/php/8.4/

Um php jetzt zu starten und beim Anmelden neu zu starten: brew services start php Oder, wenn Sie keinen Hintergrunddienst benötigen/wollen, können Sie einfach Folgendes ausführen: /opt/homebrew/opt/php/sbin/php-fpm -nodaemonize ==> nginx Docroot ist: /opt/homebrew/var/www

Der Standardport wurde in /opt/homebrew/etc/nginx/nginx.conf auf 8080 gesetzt, damit nginx ohne sudo ausgeführt werden kann.

nginx lädt alle Dateien in /opt/homebrew/etc/nginx/servers/.

Um nginx nach einem Upgrade neu zu starten: brew services restart nginx Oder, wenn Sie keinen Hintergrunddienst benötigen/wollen, können Sie einfach Folgendes ausführen: /opt/homebrew/opt/nginx/bin/nginx -g daemon off; ==> git-lfs Aktualisieren Sie Ihre git-Konfiguration, um die Installation abzuschließen:

```
# Globale git-Konfiguration aktualisieren $ git lfs install
```

```
# Systemweite git-Konfiguration aktualisieren $ git lfs install -system
```

zsh-Vervollständigungen wurden installiert unter: /opt/homebrew/share/zsh/site-functions ==> wireshark Diese Formel installiert standardmäßig nur die Befehlszeilen-Dienstprogramme.

Installieren Sie Wireshark.app mit Homebrew Cask: brew install --cask wireshark

Wenn Ihre Liste der verfügbaren Capture-Schnittstellen leer ist (Standardverhalten von macOS), installieren Sie ChmodBPF: brew install --cask wireshark-chmodbpf ==> doctl zsh-Vervollständigungen wurden installiert unter: /opt/homebrew/share/zsh/site-functions ==> azure-cli zsh-Vervollständigungen wurden installiert unter: /opt/homebrew/share/zsh/site-functions