



Claws Mail

ToC

- [Paketinfo](#)
- [Paket erstellen](#)
 - [Makefile und spec.json](#)
 - [mustache](#)
 - [Verzeichnisstruktur](#)
 - [Makefile-Parameter](#)
 - [spec.json](#)
- [Installation](#)
- [Allgemeines](#)
 - [Aufbau des Paketes](#)
 - [Nomenklatur](#)
 - [Unattended-Switches](#)
- [Lizenzen](#)
 - [Dieses Paket](#)
 - [claws-mail](#)
 - [psDetail](#)
 - [Mustache](#)
- [Anmerkungen/ToDo](#)

Paketinfo

Diese OPSI-Paket fuer **Claws Mail** wird fuer das Repository von *O4I* sowie fuer den internen Einsatz im lokalen Repository entwickelt.

Mit dem vorliegenden Code lassen sich verschiedene OPSI-Pakete erstellen, die auf die Besonderheiten der jeweiligen Repositories eingehen. Die Erstellung wird durch ein **Makefile** gesteuert.

Paket erstellen

Dieser Abschnitt beschaeftigt sich mit der Erstellung des OPSI-Paketes aus dem Source-Paket und nicht mit dem OPSI-Paket selbst.

Makefile und spec.json

Da aus den Quellen verschiedene Versionen des Paketes mit entsprechenden Anpassungen generiert werden sollen (intern, O4I; testing/release) wurde hierfuer ein **Makefile** erstellt. Darueber hinaus steuert **spec.json** die Erstellung der Pakete.

Im Idealfall ist beim Erscheinen einer neuen Release von Claws Mail lediglich die **spec.json** anzupassen.

mustache

Als Template-Engine kommt **Mustache** zum Einsatz.

Im Detail wird hier eine Go-Implementierung verwendet. Die Software ist auf [Github](#) zu finden. Binaries fuer Linux und Windows liegen diesem Paket bei.

Das in vorherigen Versionen dieses Paketes (<9) verwendete **pys tache** kommt nicht mehr zum Einsatz.

Verzeichnisstruktur

Die erstellten Pakete werden im Unterverzeichnis **PACKAGES** abgelegt.

Einige Files (control, postinst, setup.opsiscript) werden bei der Erstellung erst aus *.in*-Files generiert, welche sich in den Verzeichnissen SRC/OPSI und SRC/CLIENT_DATA befinden. Die SRC-Verzeichnisse sind in den OPSI-Paketen nicht mehr enthalten.

Fuer den eigentlichen Buildvorgang wird das Verzeichnis **BUILD** verwendet.

Makefile-Parameter

Hinweise zur Verwendung des Makefile liefert der Aufruf von

```
make help
```

Der vorliegende Code erlaubt die Erstellung von OPSI-Paketen fuer die Releases gemaess der Angaben in spec.json. Es kann jedoch bei der Paketerstellung ein alternatives Spec-File uebergeben werden:

```
SPEC=<spec_file>
```

Das Paket kann mit *"batteries included"* erstellt werden. In dem Fall erfolgt der Download der Software beim Erstellen des OPSI-Paketes und nicht erst bei dessen Installation:

```
ALLINC=[true|false]
```

Standard ist hier die Erstellung des leichtgewichtigen Paketes (ALLINC=false).

Bei der Installation des Paketes im Depot wird ein eventuell vorhandenes files-Verzeichnis zu-naechst gesichert und vom postinst-Skript spaeter wiederhergestellt. Diese Verzeichnis beinhaltet die eigentlichen Installationsfiles. Sollen alte Version aufgehoben werden, kann das ueber einen Parameter beeinflusst werden:

```
KEEPFILES=[true|false]
```

Standardmaessig sollen die Files geloescht werden.

OPSI erlaubt des Pakete im Format cpio und tar zu erstellen.

Als Standard ist cpio festgelegt.

Das Makefile erlaubt die Wahl des Formates ueber die Umgebungsvariable bzw. den Parameter:

```
ARCHIVE_FORMAT=<cpio|tar>
```

spec.json

Haeufig beschraenkt sich die Aktualisierung eines Paketes auf das Aendern der Versionsnummern und des Datums etc. In einigen Faellen ist jedoch auch das Anpassen weiterer Variablen erforderlich, die sich auf verschiedene Files verteilen.

Auch das soll durch das Makefile vereinfacht werden. Die relevanten Variablen sollen nur noch in spec.json angepasst werden. Den Rest uebernimmt *make*

Installation

Die Software selbst wird - sofern bei der Paketerstellung nicht anders vorgegeben - nicht mit diesem Paket vertrieben. Fuer die *"batteries included"*-Pakete entfaellt dieser Abschnitt.

Je nach Art des erstellten Paketes erfolgt bei der Installation im Depot durch das postinst-Script der Download der Software vom Hersteller (Windows, 32 und 64 Bit).

Ein manueller Download sollte dann nicht erforderlich sein. Auf dem Depot-Server ist **curl** oder **wget** erforderlich.

Das Gesamtvolumen der herunterzuladenden Dateien betraegt ca. **55 MByte**.

Das 64-Bit-Paket ist von den Claws-Mail-Entwicklern derzeit noch als experimentell gekennzeichnet.

Allgemeines

Aufbau des Paketes

- **variables.opsiinc** - Da Variablen ueber die Scripte hinweg mehrfach verwendet werden, werden diese (bis auf wenige Ausnahmen) zusammengefasst hier deklariert.
- **product_variables.opsiinc** - die produktspezifischen Variablen werden hier definiert
- **setup.opsiscript** - Das Script fuer die Installation.
- **uninstall.opsiscript** - Das Uninstall-Script
- **delsub.opsiinc** - Wird von Setup und Uninstall gemeinsam verwendet. Vor jeder Installation/jedem Update wird eine alte Version entfernt. (Ein explizites Update-Script existiert derzeit nicht.)
- **checkinstance.opsiinc** - Pruefung, ob eine Instanz der Software laeuft. Gegebenenfalls wird das Setup abgebrochen. Optional kann eine laufende Instanz zwangsweise beendet werden.
- **helpers.opsifunc** - Enthaelte eine Reihe von Library Funktionen; wird hier fuer OSPI-Service-Calls verwendet.
- **checkvars.sh** - Hilfsscript fuer die Entwicklung zur Ueberpruefung, ob alle verwendeten Variablen deklariert sind bzw. nicht verwendete Variablen aufzuspuren.
- **bin/** - Hilfsprogramme; hier: **psdetail**
- **images/** - Programmbilder fuer OPSI

Nomenklatur

Praefixes in der Produkt-Id definieren die Art des Paketes:

- **0_** - Es handelt sich um ein Test-Paket. Beim Uebergang zur Produktions-Release wird der Praefix entfernt.
- **o4i_, dfn_** - Das Paket ist zur Verwendung im O4I-Repository vorgesehen.

Die Reihenfolge der Praefixes ist relevant; die Markierung als Testpaket ist stets fuehrend.

Unattended-Switches

Es handelt sich um ein *NSIS*-Paket mit den hier gebraeuchlichen Parametern.

siehe auch: <http://www.silentinstall.org/nsis>

Lizenzen

Dieses Paket

Dieses OPSI-Paket steht unter der *GNU General Public License* **GPLv3**.

Ausgenommen von dieser Lizenz sind die unter **bin/** zu findenden Hilfsprogramme. Diese unterliegen ihren jeweiligen Lizenzen.

claws-mail

Das verwendete Logo steht unter der GPLv3. (siehe: [Wikipedia: ClawsMailLogo.png](#))

Die Variationen fuer das OPSI-Paket wurden von mir unter Verwendung weiterer freier Grafiken erstellt

psdetail

Autor der Software: Jens Boettge <jens.boettge@mpi-halle.mpg.de>

Die Software **psdetail.exe** wird als Freeware kostenlos angeboten und darf fuer nichtkommerzielle sowie kommerzielle Zwecke genutzt werden. Die Software darf nicht veraendert werden; es duerfen keine abgeleiteten Versionen daraus erstellt werden.

Es ist erlaubt Kopien der Software herzustellen und weiterzugeben, solange Vervielfaeltigung und Weitergabe nicht auf Gewinnerwirtschaftung oder Spendensammlung abzielt.

Haftungsausschluss:

Der Autor lehnt ausdruecklich jede Haftung fuer eventuell durch die Nutzung der Software entstandene Schaeden ab.

Es werden keine ex- oder impliziten Zusagen gemacht oder Garantien bezueglich der Eigenschaften, des Funktionsumfanges oder Fehlerfreiheit gegeben.

Alle Risiken des Softwareeinsatzes liegen beim Nutzer.

Der Autor behaelt sich eine Anpassung bzw. weitere Ausformulierung der Lizenzbedingungen vor.

Fuer die Nutzung wird das *.NET Framework ab v3.5* benoetigt.

Mustache

Die verwendete *Mustache Template Engine for Go* steht unter [MIT-Lizenz](#).

Anmerkungen/ToDo

- Die Festlegung von *Claws Mail* als **Default-Mailer** hat im Test nicht funktioniert. Die Umsetzung der entsprechenden Property wurde daher noch nicht implementiert

Jens Boettge <boettge@mpi-halle.mpg.de>, 2021-03-08 13:00:16 +0100