

SDN ONOS Developer Installation



1 Einleitung

1.1 Dieses Dokument und das Ziel

In diesem Dokument geht es darum auf einer vorhandenen Ubuntu 16.04 LTS Installation ein ONOS Developer System zu installieren.

In diesem Dokument geht es nicht darum wie man ONOS konfiguriert oder bedient. Sondern nur um die Installation.

Eine Developer Version von ONOS hat den Vorteil (oder Nachteil) das man es manuell starten muss und wenn man CTRL-C drückt ist ONOS weg. Das eignet sich ganz hervorragend um Tests durchzuführen und um eigene Module zu installieren.

Für eine Produktion ist diese Art nicht geeignet, nicht im Geringsten.

1.2 Voraussetzungen

Für die Installation von ONOS ist eine Maschine nötig.

Als Referenz dient mein Dokument „Installation Ubuntu 1604 Server“ mit etwa 8 bis 12 GB Hauptspeicher und einer 16 GB großen Platte. ONOS basiert auf Java mit einer Art Applikation Server dem Apache Karaf. Von daher brauchen wir hier etwas mehr Bum.

2 Installation

2.1 Software Pakete nachinstallieren

Damit das alles funktioniert wird auf dem Host noch etwas Software benötigt. Wir brauchen das Java 8 mit Maven etwas Python und Zip

```
apt -y install java-common java-wrappers default-jdk default-jre openjfx
apt -y install git git-core
apt -y install python zip
```

2.2 Benutzer erstellen

Für die Installation von ONOS brauchen wir erst einmal einen Benutzer der die Quellen holt und den temporären ONOS startet.

```
useradd -s /bin/bash -c "ONOS Developer Install" -m onos
passwd onos
```

2.3 Login als ONOS User und Quellen holen

Am besten loggt man sich jetzt mit SSH als ONOS User ein, anstatt ständig mit `su` zu arbeiten. Als Benutzer ONOS holen wir uns per GIT nun die Quellen und legen das ganze unter unser Heimatverzeichnis.

```
cd
git clone https://gerrit.onosproject.org/onos
```

2.4 Auschecken der korrekten Version

Im GIT sind viele Versionen inkl. Betas und RCs enthalten. Mit `git tag` kann man die sich alle anschauen. Wir brauchen die Version 1.8.2

```
cd ~/onos
git checkout tags/1.8.2
```

Nachtrag. Ich habe im Verlaufe der nächsten Dokumente die Version auf 1.9.0 geändert, ohne irgendwelche Probleme.

2.5 Einrichten des Environment

Im ONOS Quellbaum gibt es eine Datei die wir in unseren Environment einpflegen können. Es definiert alle Variablen die wir brauchen und die nötig sind. In Dieser Datei stehen auch das Password und der Username für den Zugriff nachher drin. Default ist das der Benutzer `onos` mit dem Password `rocks`. Der hat aber nichts mit unserem UNIX User zu tun.

In der Datei sind auch die Applikationen in `ONOS_APPS` definiert, die ONOS starten soll. Das kann man entsprechend anpassen um z.B. gleich den DHCP Server hochzufahren oder so.

```
cd
echo >>.bashrc
echo '# adding ONOS' >>.bashrc
echo '. ~/onos/tools/dev/bash_profile' >>.bashrc
```

Danach loggen wir uns neu ein, damit das Environment greift.

2.6 SSH Key erstellen und Self-Login

Das ist leider nötig, damit er sich bei der Installation nicht lang macht. Wir brauchen für den Benutzer ein SSH Key und einmal ein Login auf sich selbst, damit der Hostkey gespeichert wird. Tut man das nicht bricht er später mit dem Fehler `do_known_hosts: hostkeys_foreach failed: No such file or directory` ab

```
cd
ssh-keygen -t rsa
ssh onos@localhost
exit
```

2.7 Starten des ONOS

Ja ! Starten des lokalen temporären ONOS ist gleich verbunden mit dem holen der nötigen Software und zusammen schrauben. Das ganze läuft über `buck` und ist ein Build System von Facebook. Da es noch nicht genug Build Systeme gibt, hat man jetzt noch eins, welches über ein eigenes Repository verfügt, toll, egal.

Zum Starten wird hier ganz einfach ein Alias verwendet und der Rest funktioniert wie von Zauberhand.

```
cd $ONOS_ROOT; ok clean
```

Dabei wird `buck` aktualisiert, er holt sich alles was er braucht, baut alles zusammen, startet ein lokalen Karaf Server und dort ein ONOS. Auf der Konsole verfällt er in ein Log. Sobald man hier CTRL-C drückt wird wieder alles beendet. Der Zugriff auf ONOS ist nur über ein anderes Terminal oder über Web möglich.

3 Testen

Wir wollen nur kurz testen ob es geht. Wir loggen uns noch einmal als `onos` User ein und machen ein „Telnet“ auf ONOS. Das sieht dann so aus:

```

onos@sdn-onos-dev-01:~$ onos localhost
Warning: Permanently added '[localhost]:8101' (RSA) to the list of known hosts.
Password authentication
Password:
Welcome to Open Network Operating System (ONOS)!

  _____
 /         \
|   V   V   |
|  / \ / \  |
| /   \   \ |
| \   /   / |
|  \ / \ /  |
|   V   V   |
 \         /
  _____

Documentation: wiki.onosproject.org
Tutorials:    tutorials.onosproject.org
Mailing lists: lists.onosproject.org

Come help out! Find out how at: contribute.onosproject.org

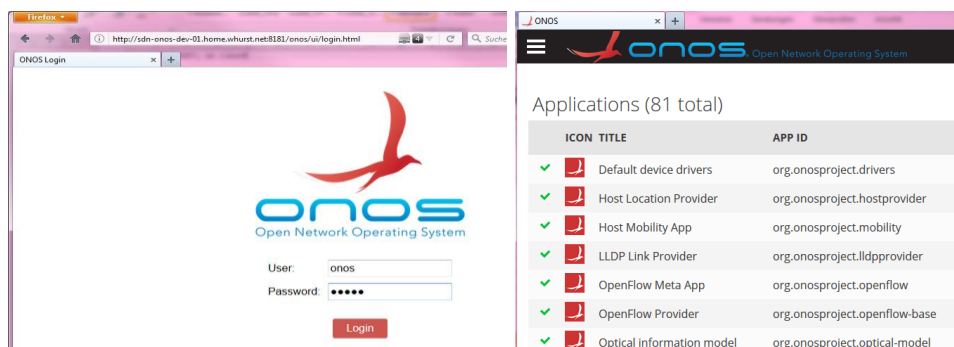
Hit '<tab>' for a list of available commands
and '[cmd] --help' for help on a specific command.
Hit '<ctrl-d>' or type 'system:shutdown' or 'logout' to shutdown ONOS.

onos> apps -a -s
* 14 org.onosproject.drivers          1.8.2   Default device drivers
* 17 org.onosproject.optical-model    1.8.2   Optical information model
* 19 org.onosproject.openflow-base   1.8.2   OpenFlow Provider
* 25 org.onosproject.lldpprovider     1.8.2   LLDP Link Provider
* 44 org.onosproject.mobility         1.8.2   Host Mobility App
* 47 org.onosproject.hostprovider     1.8.2   Host Location Provider
* 48 org.onosproject.openflow         1.8.2   OpenFlow Meta App
* 50 org.onosproject.fwd              1.8.2   Reactive Forwarding App
* 66 org.onosproject.proxyarp        1.8.2   Proxy ARP/NDP App
onos>

```

Ok, rennt. Dann schauen wir uns die Web-UI an. Die URL muss man natürlich anpassen, weil ihr vermutlich nicht den Hostnamen oder die gleiche IP habt, aber der Port passt.

URL: <http://sdn-onos-dev-01:8181/onos/ui/login.html> User `onos` Password `rocks`



4 Nach einem Reboot

4.1 Manueller Start

Nach einem Reboot braucht man sich nur noch als `onos` User anmelden und ONOS starten

```
cd $ONOS_ROOT; ok clean
```

Das war es

4.2 Automatischer Start

Wenn man will das ONOS beim Booten gleich hochkommen soll, gibt es ein Problem. Wir brauchen jetzt `screen`, wir müssen auf die Konsole drauf. Weiterhin ein Start Script das wir benutzen können. Eigentlich drei damit das klappt.

Wir brauchen kurz noch einmal Root Rechte um `screen` zu installieren und um in der `/etc/rc.local` ein Script zu starten.

```
apt -y install screen
echo 'screen -d -m bash /home/onos/start_on_boot.sh' >/etc/rc.local.onos
```

Leider hat „jemand“ am Ende von `/etc/rc.local` ein `exit` platziert, was es uns schwierig macht, wir müssen die Datei editieren und folgende Zeile vor das `exit` hinzufügen:

```
su - onos -c "bash /etc/rc.local.onos"
```

Damit wird als ROOT das Script `/etc/rc.local.onos` aufgerufen aber im Kontext des Users `onos`. Und der startet dann `screen`. `screen` unter Root zu starten wäre nicht zielführend, weil der Benutzer `onos` dann keinen Zugriff drauf hätte.

Als Benutzer `onos` schreiben wir jetzt noch ein Script welches von `screen` gestartet wird, wo wir nur unseren `onos` starten. Jedoch können wir hier nicht `ok` benutzen, weil es ein Alias ist.

```
echo '. ~/.onos/tools/dev/bash_profile' >start_on_boot.sh
echo "cd \"$ONOS_ROOT; NO_BUCKD=1 onos-buck run onos-local -- clean" >>start_on_boot.sh
```

Das testen wir jetzt in dem wir Rebooten und schauen was passiert. Man sollte danach den Prozess sehen. Nun kann man sich als `onos` Einloggen und mittels `screen -r` sich die Session holen. Mit CTRL-A D kann man den `screen` wieder verlassen, ihn aber weiterlaufen lassen. Ganz nett.

5 Updates und Versionswechsel

Durch GIT und durch den Umstand dass wir das „nur“ brauchen für den Start und BUCK alles als Binary holt, können wir extrem schnell die Version von ONOS ändern.

Dazu stoppen wir ONOS vollständig und gehen in das ONOS GIT Verzeichnis. Dort können wir dann mit GIT einfach die Version ändern und ONOS ganz normal starten. Auf Updates ist möglich. Das ist ganz praktisch.

Beispiel Update von 1.8.2 auf 1.9.0 ...

```
onos@sdn-onos-dev-01:~/onos$ git checkout tags/1.9.0
Vorherige Position von HEAD war 4a58d77... Tagging 1.8.2
HEAD ist jetzt bei 1bbaaaa... Tagging 1.9.0
onos@sdn-onos-dev-01:~/onos$ ok clean
Not using buckd because NO_BUCKD is set.
[-] PROCESSING BUCK FILES...FINISHED 1,7s [100%] 🍷 New buck daemon
[-] DOWNLOADING... (0,00 B/S AVG, TOTAL: 0,00 B, 0 Artifacts)
[-] BUILDING...FINISHED 4m43,4s [100%] (743/743 JOBS, 598 UPDATED, 598 [80,5%] CACHE MISS)
tar: Schreibfehler
Running clean installation...
# Host [localhost]:8101 found: line 2
/home/onos/.ssh/known_hosts updated.
Original contents retained as /home/onos/.ssh/known_hosts.old
Creating local cluster configs for IP 127.0.0.1...
Waiting for karaf.log
INFORMATION: Installing and starting initial bundles
```

6 Installationsprotokoll

6.1 Arbeiten als Root

```

whurst@sdn-onos-dev-01:~$ sudo su -
root@sdn-onos-dev-01:~# apt -y install java-common java-wrappers default-jdk default-jre
openjfx
Paketlisten werden gelesen... Fertig
Abhängigkeitsbaum wird aufgebaut.
Statusinformationen werden eingelesen... Fertig
Die folgenden zusätzlichen Pakete werden installiert:
  ca-certificates-java dbus-x11 default-jdk-headless default-jre-headless fonts-dejavu-extra gconf-service gconf-service-backend gconf2
blabla

Holen:4 http://de.archive.ubuntu.com/ubuntu xenial/main amd64 libbonobo2-0 amd64 2.32.1-3 [211 kB]
blabla

Holen:142 http://de.archive.ubuntu.com/ubuntu xenial/universe amd64 va-driver-all amd64 1.7.0-1 [4.620 B]
Holen:143 http://de.archive.ubuntu.com/ubuntu xenial/main amd64 vdpau-driver-all amd64 1.1.1-3ubuntu1 [4.674 B]
Es wurden 93,1 MB in 6 min 36 s geholt (235 kB/s).
Extrahiere Vorlagen aus Paketen: 100%

Noch viel mehr blabla

done.
done.
root@sdn-onos-dev-01:~# apt -y install python zip
Paketlisten werden gelesen... Fertig
Abhängigkeitsbaum wird aufgebaut.
Statusinformationen werden eingelesen... Fertig
»python« ist bereits die neueste Version (2.7.11-1).
python wurde als manuell installiert festgelegt.
Die folgenden NEUEN Pakete werden installiert:
  zip
0 aktualisiert, 1 neu installiert, 0 zu entfernen und 0 nicht aktualisiert.
Es müssen 158 kB an Archiven heruntergeladen werden.
Nach dieser Operation werden 587 kB Plattenplatz zusätzlich benutzt.
Holen:1 http://de.archive.ubuntu.com/ubuntu xenial/main amd64 zip amd64 3.0-11 [158 kB]
Es wurden 158 kB in 0 s geholt (0 B/s).
Vormals nicht ausgewähltes Paket zip wird gewählt.
(Lese Datenbank ... 99667 Dateien und Verzeichnisse sind derzeit installiert.)
Vorbereitung zum Entpacken von .../archives/zip_3.0-11_amd64.deb ...
Entpacken von zip (3.0-11) ...
Trigger für man-db (2.7.5-1) werden verarbeitet ...
zip (3.0-11) wird eingerichtet ...
root@sdn-onos-dev-01:~# useradd -s /bin/bash -c "ONOS Developer Install" -m onos
root@sdn-onos-dev-01:~# passwd onos
Geben Sie ein neues UNIX-Passwort ein:
Geben Sie das neue UNIX-Passwort erneut ein:
passwd: password updated successfully
root@sdn-onos-dev-01:~#

```

6.2 Arbeiten als Benutzer ONOS

```

onos@sdn-onos-dev-01:~$ cd
onos@sdn-onos-dev-01:~$ git clone https://gerrit.onosproject.org/onos
Klone nach 'onos' ...
remote: Counting objects: 71008, done
remote: Finding sources: 100% (54543/54543)
Empfange Objekte: 100% (292771/292771), 96.70 MiB | 208.00 KiB/s, Fertig.
remote: Total 292771 (delta 20643), reused 284778 (delta 20643)
Löse Unterschiede auf: 100% (109565/109565), Fertig.
Prüfe Konnektivität ... Fertig.
onos@sdn-onos-dev-01:~$ cd onos
onos@sdn-onos-dev-01:~/onos$ git checkout tags/1.8.2
Note: checking out 'tags/1.8.2'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by performing another checkout.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -b with the checkout command again. Example:

  git checkout -b <new-branch-name>

HEAD ist jetzt bei 4a58d77... Tagging 1.8.2
onos@sdn-onos-dev-01:~/onos$ cd
onos@sdn-onos-dev-01:~$ echo >>.bashrc
onos@sdn-onos-dev-01:~$ echo '# adding ONOS' >>.bashrc
onos@sdn-onos-dev-01:~$ echo '. ~/onos/tools/dev/bash_profile' >>.bashrc
onos@sdn-onos-dev-01:~$

```


6.3 Starten vom ONOS

```

onos@sdn-onos-dev-01:~$ echo $ONOS_ROOT
/home/onos/onos
onos@sdn-onos-dev-01:~$ cd $ONOS_ROOT; ok clean
Updating Buck...
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100 43.8M  100 43.8M    0     0  187k      0  0:03:59  0:03:59 --:--:-- 214k
Archive:  cache/buck-v2016.12.02.01.zip
  inflating: buck
  extracting: .buck_version
  creating: plugins/
  inflating: plugins/onos.jar
  inflating: plugins/yang.jar
Successfully updated Buck in /home/onos/onos/bin/buck to buck-v2016.12.02.01.zip

Not using buckd because NO_BUCKD is set.
[-] PROCESSING BUCK FILES...FINISHED 2,4s [100%] 🐛 New buck daemon
[-] DOWNLOADING... (0,00 B/S AVG, TOTAL: 0,00 B, 0 Artifacts)
[-] BUILDING...FINISHED 9m15,4s [100%] (743/743 JOBS, 743 UPDATED, 743 [100,0%])
tar: Schreibfehler
Running clean installation...
do_known_hosts: hostkeys_foreach failed: No such file or directory
onos@sdn-onos-dev-01:~/onos$ cd
onos@sdn-onos-dev-01:~/onos$ cd
onos@sdn-onos-dev-01:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/onos/.ssh/id_rsa):
/home/onos/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/onos/.ssh/id_rsa.
Your public key has been saved in /home/onos/.ssh/id_rsa.pub.
onos@sdn-onos-dev-01:~$ ssh onos@localhost
The authenticity of host 'localhost (:::1)' can't be established.
ECDSA key fingerprint is SHA256:2dk0on5dKpyftmLTzj+3yXN2x18IJNHC1Sal6dNzf6U.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
onos@localhost's password:
onos@sdn-onos-dev-01:~$ Abgemeldet
Connection to localhost closed.
onos@sdn-onos-dev-01:~$ cd $ONOS_ROOT; ok clean
Not using buckd because NO_BUCKD is set.
[-] PROCESSING BUCK FILES...FINISHED 2,5s [100%] 🐛 New buck daemon
[-] DOWNLOADING... (0,00 B/S AVG, TOTAL: 0,00 B, 0 Artifacts)
[-] BUILDING...FINISHED 3,0s [100%] (1/1 JOBS, 0 UPDATED, 0 [0,0%] CACHE MISS)
tar: Schreibfehler
Running clean installation...
Host [localhost]:8101 not found in /home/onos/.ssh/known_hosts
Creating local cluster configs for IP 127.0.0.1...
Waiting for karaf.log
Mär 10, 2017 1:31:00 PM org.apache.karaf.main.Main launch
INFORMATION: Installing and starting initial bundles
Mär 10, 2017 1:31:00 PM org.apache.karaf.main.Main launch
INFORMATION: All initial bundles installed and set to start
Mär 10, 2017 1:31:00 PM org.apache.karaf.main.lock.SimpleFileLock lock
INFORMATION: Trying to lock /tmp/onos-1.8.2/apache-karaf-3.0.5/lock
Mär 10, 2017 1:31:00 PM org.apache.karaf.main.lock.SimpleFileLock lock
INFORMATION: Lock acquired

Blabla

2017-03-10 13:31:13,103 | INFO | -message-handler | ApplicationManager | 145 - org.onosproject.onos-core-net - 1.8.2 |
Application org.onosproject.pathpainter has been installed
2017-03-10 13:31:13,116 | INFO | -message-handler | ApplicationManager | 145 - org.onosproject.onos-core-net - 1.8.2 |
Application org.onosproject.mlb has been installed

```