

Lidar Guide

BIBA

Bent Prellberg, bent@uni-bremen.de

Dezember 2022

Inhaltsverzeichnis

1 Einführung	2
1.1 Verbindung von ZXLidar 300 zur Motion Sensor Box	2
1.2 Verbindung zur MSB	3
1.3 Verbindung von MSB zu Contacts Element for IoT	4
1.4 Datenübertragung von der MSB	5
2 Lidar-Script	5
2.1 Problembehebung	6
2.1.1 Datumsfunktion	6
2.1.2 Mehrfachausführung	6
3 Web App	6
4 Lidar ZX300	7
4.1 Problembehebung	7
4.1.1 Status-Lampen bleiben Rot	7
4.2 Tipps und Hinweise	8

1 Einführung

Dieses Dokument dient der Handhabung des ZXLidar 300 und den zugehörigen Programmen, die für die Datenerfassung und Visualisierung geschrieben wurden. Dieses Dokument stellt eine zusätzliche Anleitung für die jeweiligen Skripte und den Aufbau des Lidar dar. Zusätzliche Informationen finden Sie auf dem GitHub Repository für das Projekt WindIO.

1.1 Verbindung von ZXLidar 300 zur Motion Sensor Box

Das ZXLidar verfügt über eine Ethernet-Schnittstelle. Über diese Ethernet-Schnittstelle ist die MSB (Motion Sensor Box) mit dem ZXLidar verbunden. Die Ethernet-Schnittstelle an der MSB wurde im Laufe des Projekts implementiert und ist wasserfest verbaut. Das Kabel führt zu einem Ethernet-Adapter innerhalb der MSB. Beim Öffnen sowie beim Verschließen der MSB muss darauf geachtet werden, dass keine Kabel oder der Adapter selbst eingeklemmt werden. Zudem sollte darauf geachtet werden, dass beim Verbinden der MSB mit dem ZXLidar, die wasserdichte Abdeckung nicht zu stark angezogen wird, so dass zum einen das Ethernet-Kabel nicht beschädigt wird, zum anderen, dass die Gummidichtung nicht überbeansprucht wird. Beim Verbinden ist ebenfalls darauf zu achten, dass das Plastikgewinde “gerade” angezogen wird, um die Wasserdichtigkeit zu gewähren.

Falls eine neue MSB zum Einsatz kommen soll, müssen folgende Dinge beachtet werden. Zum einen muss ein Durchgang mit einem Lochbohrer gebohrt werden. Anschließend sollte der Durchgang mit einer Feile nachbehandelt werden. Darauffolgend sollte die Schnittstelle verbaut und der Mikro-USB-Adapter an den Raspberry Pi angeschlossen werden.



Abbildung 1: Ethernet-Schnittstelle innen



Abbildung 2: Ethernet-Schnittstelle außen

Bevor nun aber Daten von dem ZXLidar empfangen werden können, muss für die MSB eine statische IP festgelegt werden.. Hierfür benutzen Sie den Befehl **ifconfig**, um sich den Namen von dem Ethernet-Adapter anzeigen zu lassen. Dieser lautete bei meiner Einrichtung **enx00e04a3bd32c**. Um die statische IP-Adresse einzustellen, muss anschließend der Befehl **sudo ip addr add 10.10.8.2 dev enx00e04a3bd32c** verwendet werden.

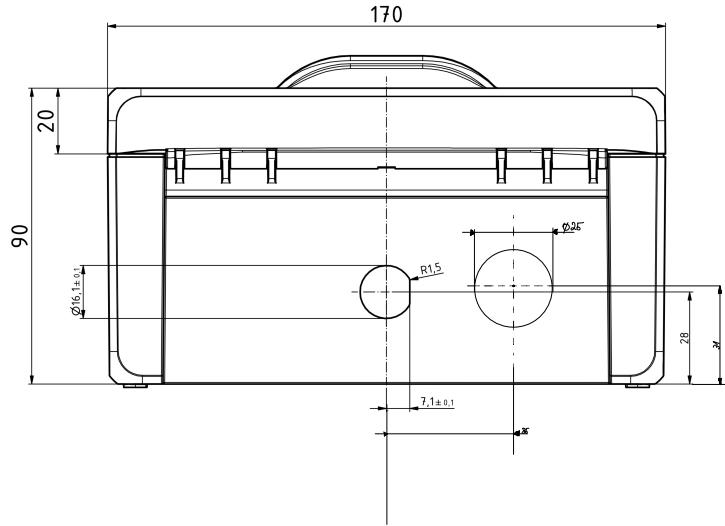


Abbildung 3: Bohrung für den Ethernet-Adapter (26 mm Durchmesser)

1.2 Verbindung zur MSB

Um auf die MSB zugreifen zu können wird ein reverse SSH (Secure Shell) verwendet, eine Anmeldung mit Benutzername und Passwort ist derzeit leider **nicht möglich**. Um sich mit der MSB verbinden zu können, muss man zunächst eine SSH-Verbindung mit der BIBA-Maschine herstellen. Siehe hierfür letzte Seite. Auf der BIBA-Maschine ist ein weiterer SSH-Key hinterlegt, mit dem man auf die MSB zugreifen kann. Sollte man sich lokal (gleiches Netzwerk) mit der MSB verbinden wollen, so muss der SSH-Key lokal gespeichert sein. Siehe hierfür erneut letzte Seite. Es kann jedoch vorkommen, dass die Verbindung zum einen nicht hergestellt werden kann, zum anderen kann es auch vorkommen, dass die Verbindung ungewollt unterbrochen wird. Oftmals hilft hierbei das erneute Aufbauen der Verbindung. Außerdem ist es sinnvoll, nicht genutzte Services, die auf der MSB aktiviert sind, zu deaktivieren, da diese die Performance der MSB schmälern. Hierzu navigiert man mit `cd` zum Pfad `home/motion_sensor_box/scripts/stop_services.sh`, in dem das Shell-Skript liegt. Um das Shell-Skript zu starten, folgt der Befehl `./stop_services.sh`. Beachten Sie, dass nach dem reboot der MSB, die Services automatisch wieder starten.

Wenn man sich im gleichen Netzwerk wie die MSB befindet, kann man sich ohne den reversen SSH direkt mit der MSB verbinden. Der Befehl hierfür lautet `ssh pi@msb-0005-a.local -i Pfad/zum/SSH-KEY`. Es bietet sich also an, wenn

man mit der MSB arbeitet und diese momentan in keinem Netzwerk ist, einen eigenen HotSpot zu schaffen, mit dem man sein Gerät und die MSB verbindet. Der Name des HotSpots und das Passwort des HotSpots sollten einem schon voreingestellten Netzwerk entsprechen. Die Einstellungen des BIBA-Wlans oder des Hackspace bieten sich hierfür an (Netzwerk: hackspace; Passwort: biba4ever).

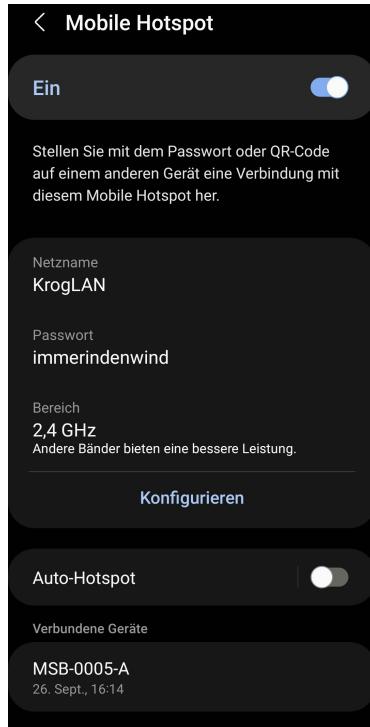


Abbildung 4: HotSpot-Verbindung

1.3 Verbindung von MSB zu Contacts Element for IoT

Die grundlegende Architektur ist auf GitHub unter dem Repository **lidar-data** zu finden. Manchmal kann es jedoch vorkommen, dass der Client von Elements for IoT nicht richtig funktioniert und deswegen keine Daten angezeigt werden. Um zu überprüfen, ob das Problem beim Lidar liegt oder der Client nicht richtig funktioniert, ist es sinnvoll, mit Hilfe von simulierten Lidar-Daten und dem MQTT-Skript Testdaten an Elements for IoT zu senden. Falls diese nicht angezeigt werden sollten, ist es ratsam, sich mit Ian Altmann (Ian.Altmann@contact-software.com) von Contact-Software in Verbindung zu setzen. Oftmals kann dieser das Problem schnell beheben. Ein Aussetzen der Daten kann auch durch einen kurzen Abbruch der Internet-Verbindung entstehen. Um wieder Daten zu versenden, empfiehlt es sich, die jeweiligen Skripte (siehe Kapitel 2.1.2) zu

stoppen und nach einer Wartezeit von ca. 5 Sekunden erneut zu starten.

```
C:\Users\pre\PycharmProjects\lidar-data\venv\Scripts\python.exe C:/Users/pre/PycharmProjects/lidar-data/src/lidar_mqtt.py
('print': True, 'url': 'windio-contact.northeurope.cloudapp.azure.com', 'port': 1883, 'user': 'urn:uni-bremen:bik:wio:0:0:wns:0002@3', 'password':
Working with user: urn:uni-bremen:bik:wio:0:0:wns:0002@3
Traceback (most recent call last):
  File "C:/Users/pre/PycharmProjects/lidar-data/src/lidar_mqtt.py", line 604, in <module>
    client.connect(url, port)
  File "C:/Users/pre/PycharmProjects/lidar-data/venv\lib\site-packages\paho\ mqtt\client.py", line 914, in connect
    return self.reconnect()
  File "C:/Users/pre/PycharmProjects/lidar-data/venv\lib\site-packages\paho\ mqtt\client.py", line 1044, in reconnect
    sock = self._create_socket_connection()
  File "C:/Users/pre/PycharmProjects/lidar-data/venv\lib\site-packages\paho\ mqtt\client.py", line 3685, in _create_socket_connection
    return socket.create_connection(addr, timeout=self._connect_timeout, source_address=source)
  File "C:/Users\pre\AppData\Local\Programs\Python\Python39\lib\socket.py", line 844, in create_connection
    raise err
  File "C:/Users\pre\AppData\Local\Programs\Python\Python39\lib\socket.py", line 832, in create_connection
    sock.connect(sa)
ConnectionRefusedError: [WinError 10061] Es konnte keine Verbindung hergestellt werden, da der Zielcomputer die Verbindung verweigerte

Process finished with exit code 1
```

Abbildung 5: Connection Refused

1.4 Datenübertragung von der MSB

Mit Hilfe des Befehls **scp** (secure copy) ist es möglich, Dateien von der MSB auf einen PC oder umgekehrt zu übertragen. Wenn man lokal mit der MSB verbunden ist (gleiches Netzwerk), muss für einen Datentransfer von MSB zu PC der Befehl

- **scp -i Pfad/zum/SSH-Key pi@msb-0005-a.local:/Dateipfad/auf/MSB “Dateipfad/auf/PC”**

und von PC zur MSB der Befehl

- **scp -i Pfad/zum/SSH-Key Dateipfad/auf/PC pi@msb-0005-a.local: “Dateipfad/auf/MSB”**

folgen.

Wenn die Verbindung über den reversen SSH-Tunnel erfolgt, so muss bei der Datenübertragung von MSB zur BIBA-Maschine der Befehl

- **scp -i Pfad/zum/SSH-Key -P 65005 pi@localhost:/Dateipfad/auf/MSB “Dateipfad/auf/PC”**

und bei Datenübertragung von BIBA-Maschine zur MSB der Befehl

- **scp -i Pfad/zum/SSH-Key -P 65005 “Dateipfad/auf/PC”pi@localhost: “/Dateipfad/auf/MSB”**

folgen.

2 Lidar-Script

Das Lidar-Script wurde in Python geschrieben und ist genauso wie alle folgenden Skripte auf GitHub zu finden. Die Daten werden mit Modbus erhoben, aufbereitet und anschließend mit ZeroMQ an ein anderes Skript versendet.

2.1 Problembehebung

Da die Software selbst entwickelt wurde und während der Entwicklung beziehungsweise während des Einsatzes Bugs aufgetaucht sind und behoben werden konnten, dient dieses Kapitel dazu, häufig auftretende Fehler schneller beheben zu können.

2.1.1 Datumsfunktion

Da Modbus auf 16-Bit basiert, wurde von dem Hersteller ZXLidar die Entscheidung getroffen, einen etwas unkonventionellen Timestamp einzusetzen. Wie dieser aufgebaut ist beziehungsweise wie die Umrechnung funktioniert, ist umfangreich auf GitHub beschrieben. Obwohl die Datumsfunktion getestet wurde, kann es dazu kommen, dass dennoch unentdeckte Fehler bei Datumsverschiebungen, wie einem Schaltjahr, auftauchen. Ist dies der Fall, so muss die Funktion weiter entwickelt und diese Fehler behoben und auf GitHub dokumentiert werden.

2.1.2 Mehrfachausführung

Falls ungewollt das Skript mehrfach ausgeführt wird, kann es dazu kommen, dass eine Fehlermeldung auftaucht und das Skript abbricht. Ursache hierfür ist, dass wenn mehrere Skripte parallel laufen, die Daten nicht richtig übermittelt werden. Um sicherzugehen, dass das Skript nicht im Hintergrund ausgeführt wird, ist es sinnvoll sich mit dem Befehl **htop** die laufenden Prozesse anzeigen zu lassen. Falls das Skript mehrfach ausgeführt wird, kann dieses mit dem Befehl **sudo systemctl stop msb-lidar** (für das Lidar-Skript) bzw. **sudo systemctl stop msb-mqtt_update** (für das Mqtt-Skript) beendet werden. Mit dem Befehl **sudo systemctl status service name** kann außerdem der Status des services abgefragt werden.

```
pi@MSB-0005-A:~/motion-sensor-box/src/lidar/config $ sudo systemctl status msb-lidar
● msb-lidar.service - Lidar Unit Service
  Loaded: loaded (/etc/systemd/system/msb-lidar.service; enabled; vendor preset: enabled)
  Active: active (running) since Tue 2022-06-07 15:04:01 BST; 3min 29s ago
    Main PID: 3227 (python)
       Tasks: 2 (limit: 725)
      CGroup: /system.slice/msb-lidar.service
              └─3227 /usr/bin/python /home/pi/motion-sensor-box/src/lidar/src/lidar_data.py

Jun 07 15:04:01 MSB-0005-A systemd[1]: Started Lidar Unit Service.
pi@MSB-0005-A:~/motion-sensor-box/src/lidar/config $
```

Abbildung 6: Status von msb-lidar

3 Web App

Die Web App dient zur Visualisierung und dem Downloaden der Daten. Außerdem können alle Parameter des Lidar live beobachtet werden. Für die Web App

wurde Django als Framework gewählt. Die Web App wurde primär für den Einsatz auf der MSB von Andreas Haselsteiner und mir entwickelt. Alle Details zur Funktionsweise finden Sie auf GitHub. Für die Web App gibt es einen Admin-Account. Die Login-Daten sind folgende: **Benutzername: admin; Passwort: biba4ever.**

4 Lidar ZX300

Das ZX300 ist in der Lage, auf bis zu elf verschiedenen Höhen (bis 200m) die horizontalen und vertikalen Windgeschwindigkeiten sowie die Windrichtungen zu messen. Wichtig ist, dass das Lidar nicht alle Höhen zum selben Zeitpunkt misst, sondern alle Höhen mit einer Sekunde Unterschied. Dies ist besonders zu beachten, wenn die Winddaten analysiert werden.

4.1 Problembehebung

4.1.1 Status-Lampen bleiben Rot

Nach dem Einschalten des Lidar fährt das Gerät hoch (zu erkennen an dem laufenden Lüfter). Laut der Lidar eigenen Software (Waltz) trat der Fehler **Laser Fault** auf (siehe Abb. 5). Außerdem kann man erkennen, dass die Statuslampe rot blinkt. Dieser Fehler trat zum einen bei Testläufen auf dem Uni-Gelände auf aber auch während der Messkampagne auf dem Gelände des Fraunhofer IWES in Bremerhaven.

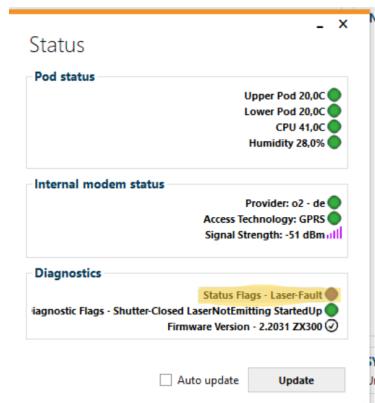


Abbildung 7: Laser Error



Abbildung 8: Statuslampe rot

Auf dem Uni-Gelände konnte der Fehler behoben werden, indem man das Lidar ausschaltete und nach einigen Minuten wieder anschaltete. Die Art der Fehlerbehandlung war in Bremerhaven leider nicht möglich, auch nach mehreren Neustarts leuchtete die Statuslampe rot. Daraufhin wurde der Support von ZXLidar benachrichtigt. Nach dem Auslesen der internen Log-Dateien, konnte

ein Fehler, der durch das interne Modem ausgelöst wurde, identifiziert werden. Daraufhin wurde, da das interne Modem momentan nicht genutzt wird, dieses abgeschaltet. Es ist jedoch nicht eindeutig erkennbar, weshalb der Fehler im internen Modem einen Laser-Error erzeugt. Nach dem Neustart des Geräts war der Fehler behoben. Sollte wider Erwarten der Fehler nochmals auftreten, so ist es sinnvoll den Support von ZXLidar, entweder telefonisch (+44 (0) 1531 651 000; Hinweis! Support von ZXLidar sitzt im Vereinigten Königreich) oder über E-Mail (support@zxlidars.com), zu kontaktieren. Der Support Mitarbeiter, der den Laser-Error behandelt hat, heißt James Dunn.

4.2 Tipps und Hinweise

Bevor man das Lidar an einem neuen Ort für eine Messkampagne einsetzen möchte, sollte man sich vorher mit dem Abbau beziehungsweise wieder Aufbau von der Met-Station vertraut machen. Des weiteren ist stets zu beachten, dass jegliche Anpassungen der Messhöhen und Parameter nur mit Hilfe der ZXLidar eigenen Software **Waltz** durchführbar sind. Für den Einsatz der Software muss ein Laptop mit einem Ethernet-Kabel mit dem Lidar verbunden werden. Eine spätere Anpassung über das Internet ist **nicht** möglich.



Abbildung 9: Anschluss
(Ethernet-Seite)



Abbildung 10: Anschluss (On-Off-Seite)

Beim Aufstellen des Lidars an der Messstelle sollte darauf geachtet werden, dass das Lidar mit Hilfe der auf der Oberseite des Lidars angebrachten Wasserwaagen gerade aufgestellt wird. Außerdem ist es ratsam, das Lidar in einem Abstand von ca. 5m zu anderen hohen Strukturen wie beispielsweise Bäumen zu platzieren, da sonst die Messung gestört wird. Nicht nur Objekte, sondern auch Wetterumstände können Messungen verhindern und so zu Messlücken in den Datensätzen führen. Deshalb ist es sinnvoll, ein größeres Zeitintervall für eine Feldmessung anzustreben. Bei ungeklärten Fragen, bietet das User-Manual von ZX-Lidar viele weitere Informationen. Das User-Manual ist in der Nextcloud des BIBAs unter dem Dateipfad /499_LIDAR_und_Stromversorgung/020_documentation hinterlegt.



Abbildung 11: ZX300

SSH-Key für BIBA-Maschine:

-----BEGIN OPENSSH PRIVATE KEY-----

```
b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEb9uZQAAAAAAAAABAAABlwAAAAdzc2gtcn
NhAAAAAwEAAQAAAYEAuB/5VYAlBMom4P/SGI//sWtQmR6dfLmqtG7oZHj9afAj/fumnUR0
CfzGYFN8bPA9WEJ+1U3UVtOUchUfpSwoePopbfVCzi7euR/iBy14vKcgHtqFM6b76PawzC
HOn/7VMDIIG5yVw7IKfrex2ICvXIDd5Cd2LXp0sb+XWml63+XB0NqhEd6SQ6/YT/Q4lt0R
8J1lqlkOEpOJspsNEApw+BOuaXy0i5uewb2MQHeaOvlUo+BgoHhiMJI9PmbeZy10bUVx
EMpUwQ9jhT2fSIWeJZ1ymrmTBH0ivTzMKDIsd0s8lf8Bjt+nc6UdxsHu7HWaySx14jOtm
N6PMn7kr9/P9sfqAI3nHm2DKbkHKFPiyM2Rrinzq0T/xbQzX4Usew1bOk2rsWCw2gz95fD
8/x458O5F3txii6D2QiVN8DzrfOsrKC9A2+/eujBMCSyW3fTqqzTbxfc8pUl8npaeDEPBY
Ju6lyNs8sWTqPpJbTlbGLExFWmdwv4rcSP2NMPSIAAAFiDnQrlU50K5VAAAAB3NzaC1yc2
EAAAGBALgf+VWAJQTkJuD/0hov/7FrUJkenXy5qrRu6GR4/Wnwl/37pp1EdAn8xmBTfGzw
PVhCftVN1FbTlHVH6UsKHj6KW31Qs4u3rkf4gcteLynlB7ahTOm++j2sMwhzp/+1TA5SB
uclcO5Sn63sdpQr1yA3eQndi16dLG/l1ppet/lwaDaoRHekkOv2E/0OCLdEfCdZao5ZDhK
TibKbDRAD8PgTrml8tlubnsG9jEB3mjr5VKPqYKB4YjCSJfT5m3mctdG1FcRDKVMEPY4U9
n0pVniWdcpq5kwR9lr08zCgyEndLPJRFAY7X/p3OlHcb7ux1msksdelrzjejz+5K/fz
/bH6gJd5x5tgym5ByhT4sptka4p86tE/8W0GceFLHsNWzpNq7FgsNoM/eXw/P8eOfDuRd7
cYoug9kIlTfA863zrKygvQNvv3rowTAKslt306qmU2733PKVCPJ6WngxDwWCupcjbPLFk
6j6SW0yGxixMRVpncL+K3Ej9jTD7JQAAAAMBAEAAAGAGEdhD5jq6LeGSmd5Syo2n2y87z
/I+N8bpixzzKHWmpxHhrNi/DxdR77c4rFHiC7ii+uSwWo2nG9gVO27o0Ql0mtwlnTog/0
nvkZCyTAFbi87kyWwWGcuKk6/nuPtMxxd7bxwZOckW50L1EtrqqcPpuJ+pbLzKFRjXzDY
E4qmLnth/2Gr7cTxAWKC7WlyS++5kpoXIzYYHMzdklzPsAiS3f+6NqAjSMrWzU1FAAhZnL
fqg9JdoeBPRHcbvHmAjjtJKIWIB00sXiFAiYHQMyrr/CVfWli9bQm1JvEtnVUIZzkaD9
oYbd7hvqodUm6UBnVbKFZREgEeOfsB5f82CXleqaLzK8lyNbRi4Uf5LQRXBS6URuM0UFRa
fGcxmsEf+1PoEBb3mrqh43t1RFN/tE1xypkDDtuRLwtu8mBuHBLb+pn0HWFqRHdh7XLcc
gnVzJXqhCuyiPxz56CF19EOVfsOYhP/V7yEv3Z/qxihIKW/X1jA6fF2lliQMVRMnZAAA
wHuyzhlpGLKdXpQRtRDN3qalMK78GdJKaclaBL63CDgf2NUKBRs2jJmFiSjt7G4/jk+fk
wyw+m1BqEtXtGt4qHXyX5xqs4/OXTEb9epiqZR1edamS4rfmVPSSnq8DV7p3F5vDnRW1XMP
9i7TpLroEtZQSbBKl+2zlvuPlniyW8UPNIDksX5QwrgGkXPafzWw2d925UqryFqpY1O+7
xz1vMjbQfMoqWDVBcsQ3CXqebEiVNj4I1WOM3DpJapM22nWAAAAMEAvQ1jjp2HTHzBPEm
NQ2QZARiSTMrrDfy+iGSnrC9zBIQV4Wzb0Si7rHtrvFGqLw3JY4eQKQUx9xA0zqCnkVnmB
ItncPtgaZwM0DeIb2Pw2VSxMX3ttm+mxXHRGt+meHNInwDkJNY2QCsWPMTK7nwK8uK2EAs
FALxOGVf3WXKepnbMGjs0IK30HaXO462GSTAhaidae7gOSgK84ajjq+ovW5VJthvLty9R/
grgQZ4kVN817HXpurSno7eszJ+Xjp5AAAAAwQD5U+O09RXDReT/StUcW66eqCVRBF7Ta3YH
7DXrJbLNbeK5T8azkeOU63Co2zkXu6agcAaRSsW0EH6X7WCE/iFyhKLixBANw7N629pTw4
rMhzDH04+5ypS66wUdM8dEZdanlwZY64lncwoRMIFFqsLCYJFDUaybOktHztCvRNLuVhYk
UXbt8MI+Fb1O7B8AlBY7WfKavhtQv0Nn50rI7gxU7IGPKM4b5Ozkd3DLezyUFl4chg/h8k
sMmof/MMjXWw0AAAQc2FuQHNhbi10aGlua3BhZAECAw==
```

-----END OPENSSH PRIVATE KEY-----

SSH-Key für lokalen Zugriff auf MSB:

-----BEGIN OPENSSH PRIVATE KEY-----

```
b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEb9uZQAAAAAAAAABAAAMwAAAAtzc2gtZW
QyNTUx0QAAACDHZptfvUfZjLR30PLBTECqVzXEEPIIQE1FpG7Wi+MwzAAAAJjfAgTL3wIE
ywAAAAAtzc2gtZWQyNTUx0QAAACDHZptfvUfZjLR30PLBTECqVzXEEPIIQE1FpG7Wi+MwzA
AAAEAXdmv8cWBscayl4kjQKHZqZVPy9ITXi6Pvvb2n847DJ8dmm1+9R9mMtHfQ8sFMQKpX
NcQQ+UhATUWkhtaL4zDMAAAAEmlbnRADW5pLWJyZW1lbi5kZQECAw==
```

-----END OPENSSH PRIVATE KEY-----