

Server-Installationsanleitung



Xesar 3.0

Installationsanleitung auf Windows Server 2016 /Datacenter/Hypervisor

Inhalt

| | | |
|---|---|---|
| 1 | Vorwort | 2 |
| 2 | Voraussetzungen: | 3 |
| 3 | Ubuntu einrichten | 4 |
| 4 | Ubuntu Updates installieren | 6 |
| 5 | Windows 10 Pro Admin PC einrichten..... | 6 |
| 6 | Xesar 3.0 Installation | 9 |

1 Vorwort

Diese Anleitung zeigt die Vorbereitung der Xesar 3.0 Installation auf einem Windows-Server mit dem Betriebssystem Versionen Windows Server 2016 Standard oder Datacenter als Hypervisor.

Die Herstellung der notwendigen IT und Serverumgebung ist nicht Teil dieser Installationsanleitung. Diese muss kundenseitig zur Verfügung gestellt werden und liegt nicht in der Verantwortung von EVVA.

Vor der Installation ist zu prüfen und zu bestätigen, dass die Xesar 3.0 Systemvoraussetzungen laut Projektcheckliste und Systemhandbuch erfüllt sind.

Die aktuelle Projektcheckliste können Sie von der Xesar Homepage unter <https://www.evva.com/at-de/produkte/elektronische-schliesssysteme-zutrittskontrolle/xesar/> herunterladen.

Wir empfehlen dringend, die Xesar 3 Installation nur in enger Zusammenarbeit mit dem zuständigen IT Administrator des Kunden durchzuführen.

Anleitung zur Installation von Xesar 3.0 unter Windows Server 2016 Datacenter.

2 Voraussetzungen:

Ein physischer Server wird mit Microsoft Windows Server 2016 aufgesetzt und als Hypervisor konfiguriert. Auf diesem wird eine VM mit aktuellem Ubuntu LTS Server installiert auf welchem in weiterer Folge Docker mit Xesar 3 läuft.

Folgende Voraussetzungen sind für eine erfolgreiche Installation von Xesar 3.0 auf einem Server mit dem Betriebssystem Windows Server 2016 notwendig:

1. Ein physischer Server mit installiertem **Windows Server 2016 /Datacenter** Betriebssystem ab Version 1607
2. Konfiguration als **Hypervisor** für **Virtuelle Maschine (VM) für Ubuntu LTS Server für Docker**
3. Der Anwender muss Windows Server und Netzwerk Verwaltungs Know How besitzen.
4. Der Anwender muss Lokale Administrationsrechte besitzen.
5. Es muss ein bestehendes DHCP Service vorhanden sein.
6. Die Server Zeitzone muss als UTC konfiguriert sein.
7. Eine Hyper-V Unterstützung muss vorhanden sein, sowie ein Virtueller Switch mit Möglichkeit zur Verbindung und Zugriff auf das Internet
8. Ein Zugriff auf Internet (Docker Trusted Registry mit Notary Service und Lizenzservice, Port 443, 4443, 8072) muss vorhanden sein.
9. Eventuell muss der Treiber für die Codierstation installiert werden. HID Omnikey 5422 wird meistens automatisch erkannt.

Wichtig: Aufgrund der Ressourcenverfügbarkeit in Verbindung mit Windows Server empfehlen wir mindestens 8GB besser noch 16GB Speicher für den physischen Server. Für die VM werden mindestens 4GB Speicher benötigt. Grundsätzlich gilt je größer die Anlage und umso mehr Personen / Traffic und Online Wandlerer umso mehr Speicher sollte zur Verfügung stehen.

3 Ubuntu einrichten

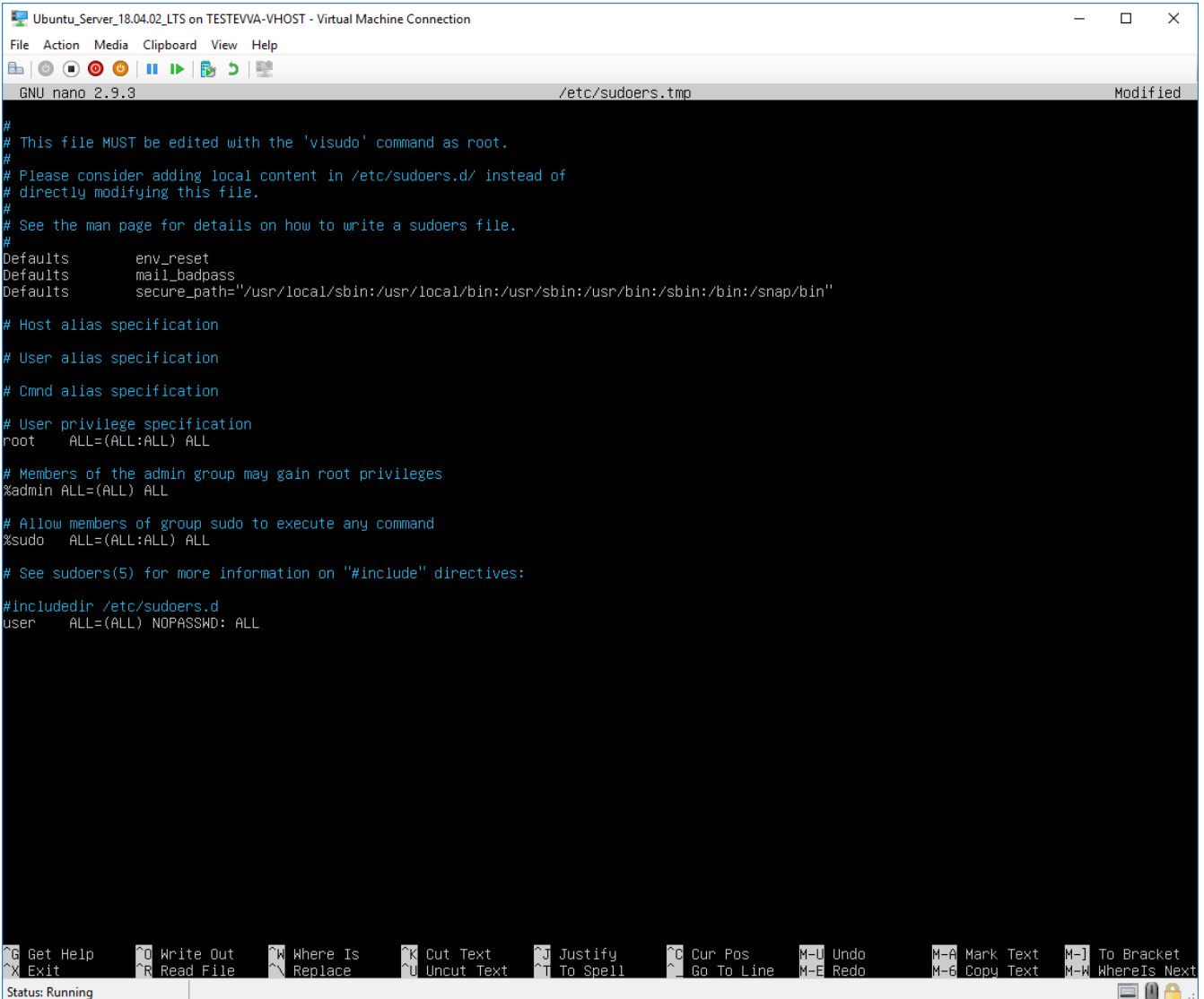
1. Passwortabfrage für sudo wird mit folgendem Befehl bearbeitet:

sudo visudo

Der nun geöffneten Datei wird am Ende folgenden Zeile hinzugefügt:

user ALL=(ALL) NOPASSWD: ALL

Der unterstrichene Bereich muss durch den Namen des Benutzers ersetzt werden, welcher bei der Installation angegeben wurde.



```
Ubuntu_Server_18.04.02_LTS on TESTEVA-VHOST - Virtual Machine Connection
File Action Media Clipboard View Help
GNU nano 2.9.3 /etc/sudoers.tmp Modified
#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"
# Host alias specification
# User alias specification
# Cmnd alias specification
# User privilege specification
root    ALL=(ALL:ALL) ALL
# Members of the admin group may gain root privileges
%admin  ALL=(ALL) ALL
# Allow members of group sudo to execute any command
%sudo  ALL=(ALL:ALL) ALL
# See sudoers(5) for more information on "#include" directives:
#include_dir /etc/sudoers.d
user    ALL=(ALL) NOPASSWD: ALL
^G Get Help      ^O Write Out    ^W Where Is    ^K Cut Text     ^J Justify     ^C Cur Pos     ^U Undo        ^M Mark Text    ^_ To Bracket
^X Exit          ^R Read File   ^N Replace     ^U Uncut Text  ^T To Spell    ^G Go To Line  ^- Redo        ^- Copy Text   ^- WhereIs Next
Status: Running
```

Mit STRG+O und anschließend Enter wird die Datei gespeichert und mit STRG+X geschlossen

2. Ein SSH Schlüsselpaar wird mit folgendem Befehl erstellt:

ssh-keygen

Name und Passwort können leer gelassen werden und mit Enter bestätigt werden.

```
user@localhost:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/user/.ssh/id_rsa):
Created directory '/home/user/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/user/.ssh/id_rsa.
Your public key has been saved in /home/user/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:eiT3PzDcMKpfIdpa9tUGAcg3CkQemJKeB/ZORGXick4 user@localhost.localdomain
The key's randomart image is:
+---[RSA 2048]----+
|  .  ==.0..  |
| 0 00.=0 o.  |
| .o. E. o ..  |
| 0 * . . .   |
| . + S . .   |
| +. B 00. o   |
| 0 .+ =.0+.0  |
| ..=.0 0*.   |
| .+0. . .+   |
+----[SHA256]-----+
user@localhost:~$ _
```

3. Den SSH Public Key zu den authorized Keys hinzufügen:

cd /home/user/.ssh/

cat id_rsa.pub > authorized_keys

Der unterstrichene Bereich muss durch den Namen des Benutzers ersetzt werden, welcher bei der Installation angegeben wurde.

```
user@localhost:~$ cd /home/user/.ssh/
user@localhost:~/ssh$ ls
id_rsa id_rsa.pub
user@localhost:~/ssh$ cat id_rsa.pub > authorized_keys
```

4 Ubuntu Updates installieren

- Mit den folgenden Befehlen werden aktuelle Updates heruntergeladen und installiert und anschließend neu gestartet:

sudo apt-get update

sudo apt-get upgrade

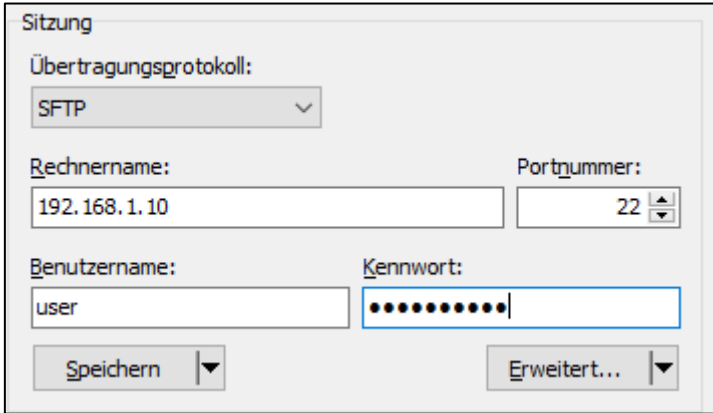
sudo apt-get dist-upgrade

sudo apt-get autoremove

sudo reboot now

5 Windows 10 Pro Admin PC einrichten

1. Zur Übertragung des SSH Schlüssel wird WINS SCP von <https://winscp.net/eng/download.php> heruntergeladen und installiert.
2. Beim Starten von WINS SCP wird nach dem Rechnernamen, Port, Benutzername und Kennwort des zuvor erstellten Ubuntu Servers gefragt.



Sitzung

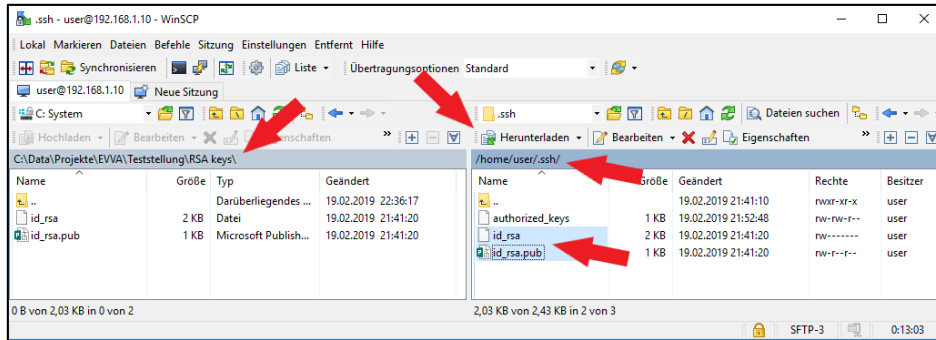
Übertragungsprotokoll:
SFTP

Rechnername: 192.168.1.10 Portnummer: 22

Benutzername: user Kennwort: ●●●●●●●●

Speichern | Erweiterter...

3. Durch die Tastenkombination STRG+ALT+H werden in WINS SCP die versteckten Dateien und Ordner angezeigt. Auf der linken Seite zu einem Ordner auf dem lokalen Windows PC wechseln. Auf der rechten Seite in den Ordner „.ssh“ am Ubuntu Server wechseln, die Dateien id_rsa und id_rsa.pub auswählen und mit der Schaltfläche „Herunterladen“ auf den Windows PC laden.



- Anschließend wird die aktuelle Version von Docker CE von <https://docs.docker.com/docker-for-windows/release-notes/> heruntergeladen und installiert. Nach einem Neustart kann die Installation überprüft werden.

```
PS C:\Users\tatshar> docker version
Client: Docker Engine - Community
Version:      18.09.2
API version:  1.39
Go version:   go1.10.8
Git commit:   6247962
Built:        Sun Feb 10 04:12:31 2019
OS/Arch:     windows/amd64
Experimental: false

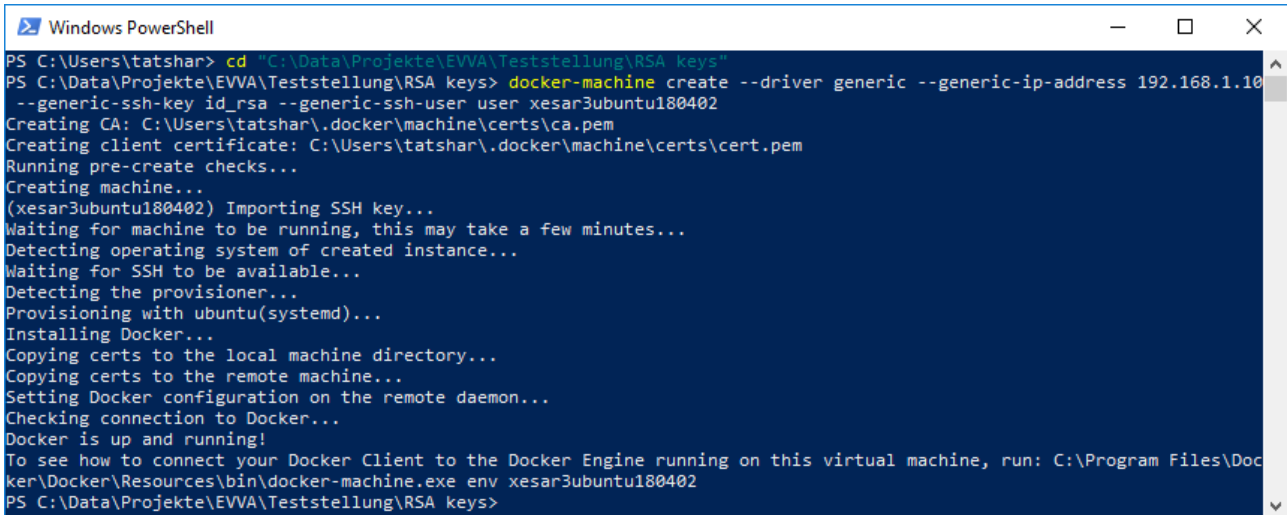
Server: Docker Engine - Community
Engine:
Version:      18.09.2
API version:  1.39 (minimum version 1.12)
Go version:   go1.10.6
Git commit:   6247962
Built:        Sun Feb 10 04:13:06 2019
OS/Arch:     linux/amd64
Experimental: false

PS C:\Users\tatshar> docker-machine version
docker-machine.exe version 0.16.1, build cce350d7
PS C:\Users\tatshar> docker-compose version
docker-compose version 1.23.2, build 1110ad01
docker-py version: 3.6.0
CPython version: 3.6.6
OpenSSL version: OpenSSL 1.0.2o  27 Mar 2018
```

- Zum Erstellen der Docker Maschine werden folgende Befehle in die Powershell oder Windows Konsole eingegeben:

cd "C:\Data\Projekte\EVVA\Teststellung\RSA keys" docker-machine create --driver generic --generic-ip-address 192.168.1.10 --generic-ssh-key id_rsa --generic-ssh-user user xesar3ubuntu180402

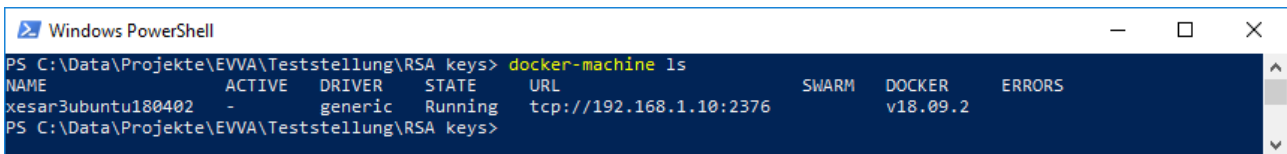
Der grün markierte Bereich muss durch den Pfad ersetzt werden in welchen zuvor die Dateien mit WINSOCP kopiert wurden, der blau markierte Bereich ist die IP Adresse des Ubuntu Servers die bei der Installation statisch vergeben wurde, der braun markierte Bereich ist der Benutzername des Ubuntu Servers welcher bei der Installation angelegt wurde und der grau markierte Bereich ist der Name, den die Docker Maschine erhalten soll.



```
Windows PowerShell
PS C:\Users\tatshar> cd "C:\Data\Projekte\EVVA\Teststellung\RSA keys"
PS C:\Data\Projekte\EVVA\Teststellung\RSA keys> docker-machine create --driver generic --generic-ip-address 192.168.1.10
--generic-ssh-key id_rsa --generic-ssh-user user xesar3ubuntu180402
Creating CA: C:\Users\tatshar\.docker\machine\certs\ca.pem
Creating client certificate: C:\Users\tatshar\.docker\machine\certs\cert.pem
Running pre-create checks...
Creating machine...
(xesar3ubuntu180402) Importing SSH key...
Waiting for machine to be running, this may take a few minutes...
Detecting operating system of created instance...
Waiting for SSH to be available...
Detecting the provisioner...
Provisioning with ubuntu(systemd)...
Installing Docker...
Copying certs to the local machine directory...
Copying certs to the remote machine...
Setting Docker configuration on the remote daemon...
Checking connection to Docker...
Docker is up and running!
To see how to connect your Docker Client to the Docker Engine running on this virtual machine, run: C:\Program Files\Docker\
ker\docker\resources\bin\docker-machine.exe env xesar3ubuntu180402
PS C:\Data\Projekte\EVVA\Teststellung\RSA keys>
```

6. Mit folgendem Befehl kann überprüft werden ob die Docker Maschine läuft:

docker-machine ls



```
Windows PowerShell
PS C:\Data\Projekte\EVVA\Teststellung\RSA keys> docker-machine ls
NAME          ACTIVE DRIVER  STATE  URL          SWARM  DOCKER  ERRORS
xesar3ubuntu180402 -      generic Running tcp://192.168.1.10:2376 v18.09.2
PS C:\Data\Projekte\EVVA\Teststellung\RSA keys>
```

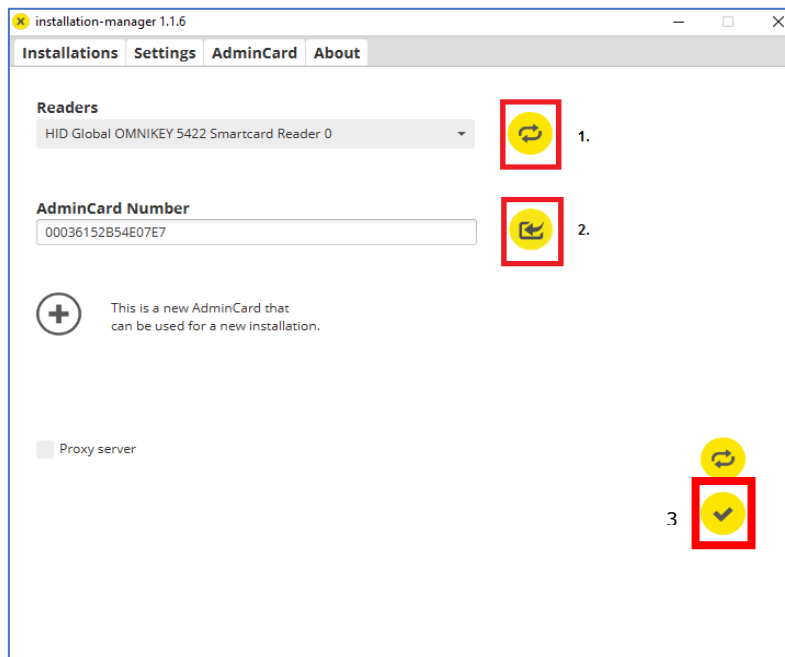
7. Schließen Sie die **Codierstation** über USB an ihrem Admin PC an und stecken Sie die **AdminCard** in den Kartenslot der Codierstation.

6 Xesar 3.0 Installation

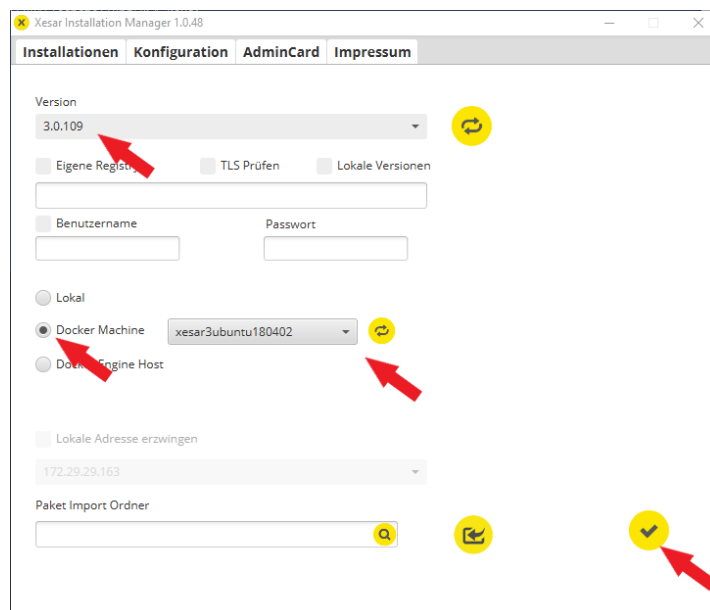
1. Laden Sie die aktuelle Xesar 3.0 Software mit dem Installation Manager von der EVVA Homepage:

<https://www.evva.com/at-de/produkte/elektronische-schliesssysteme-zutrittskontrolle/xesar/xesar-software-download/>

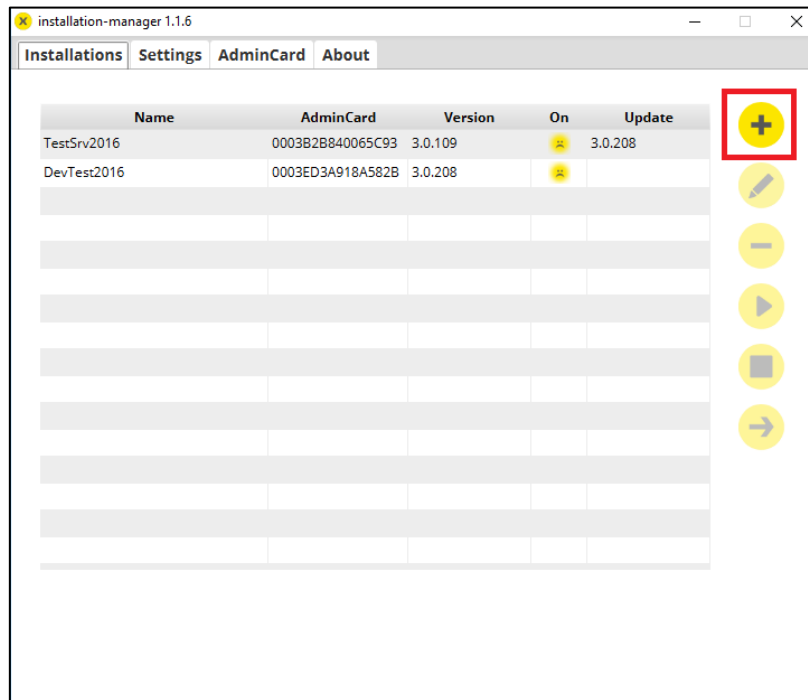
2. Öffnen Sie den Installation Manager. Wählen Sie den Tab AdminCard und laden Sie den Kartenleser (1). Anschließend laden Sie die AdminCard (2) und bestätigen die Eingabe (3)



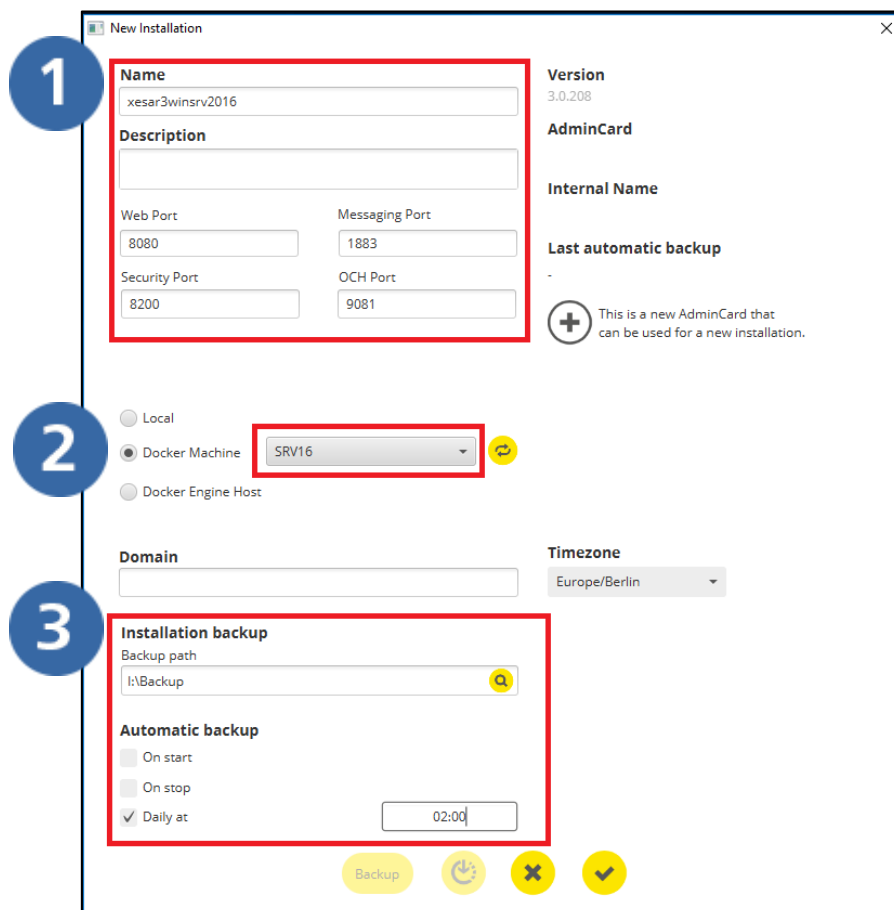
3. Wählen Sie den Tab **Konfiguration** den Abschnitt „**Docker Machine**“ und die zuvor erstellte Docker Maschine. Anschließend die Xesar Software Version auswählen und mit dem Haken speichern.



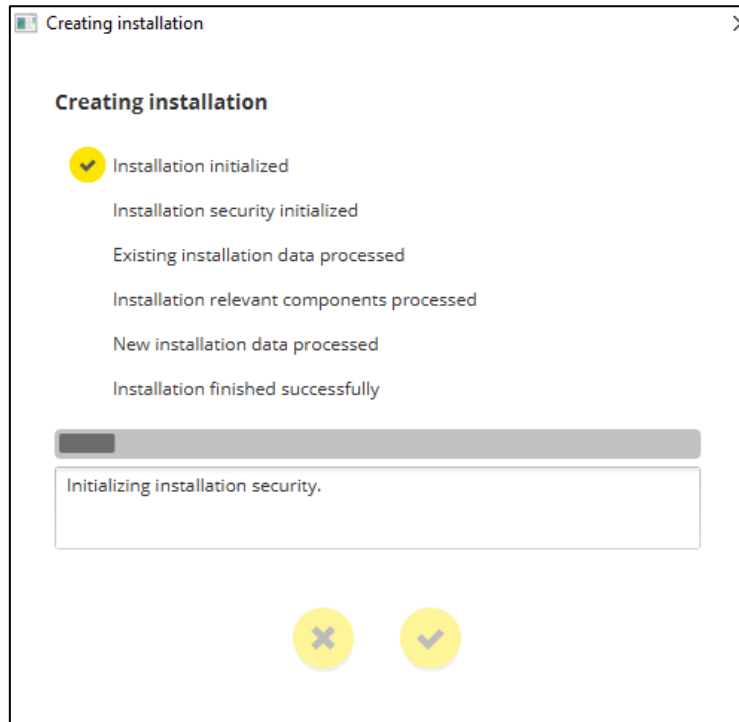
4. Wählen Sie den Tab **Installations** und fügen Sie mit „+“ eine neue Anlage hinzu.



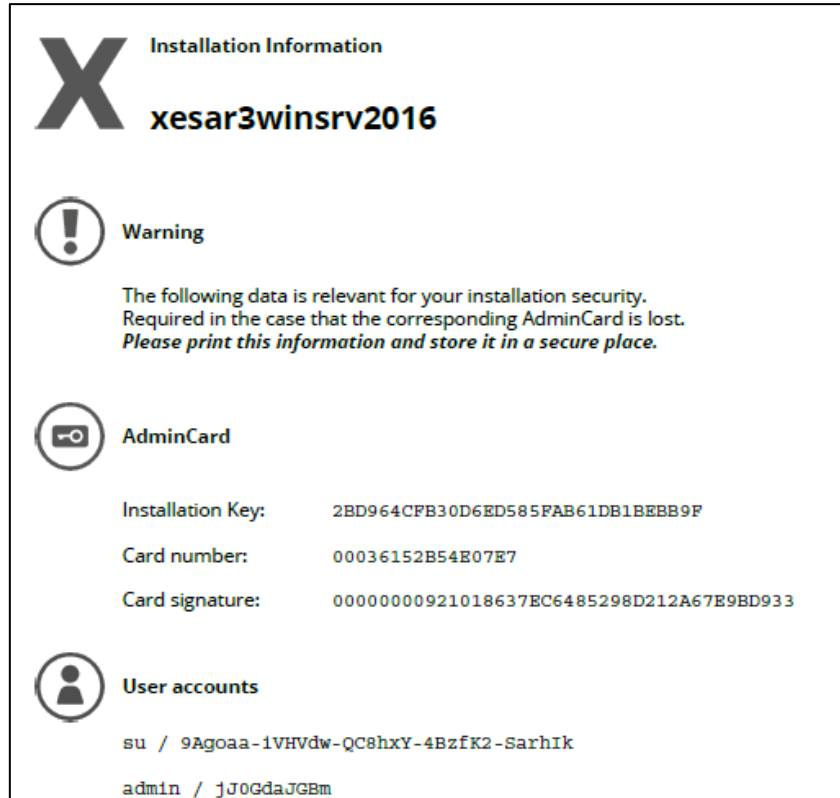
5. Tragen Sie alle Anlagen-Daten (1) ein, wählen Sie die Docker Machine (2) und definieren Sie die gewünschte automatische Sicherungsfunktionen (Backup) (3).



6. Die Anlage wird erstellt und die Installations-Informationen werden angezeigt.



7. Die wichtigen Anlagendaten werden im Dokument „Installationsinformationen“ ausgegeben.



Wichtig: Diese Anlageninformationen sind bei Verlust oder Defekt der AdminCard die einzige Möglichkeit, die Anlage wieder zu bedienen. Daher muss dieses File ausgedruckt und sicher verwahrt werden.

