

# NagiosQL 3.2

## Installation

2012 – Version 2.0  
Martin Willisegger

# Inhaltsverzeichnis

1	Installation.....	3
1.1	Voraussetzungen.....	3
1.2	Vorbereitung Verzeichnisstruktur.....	3
1.3	Nagios Konfigurationsdateien.....	4
1.4	Vorbereitung Apache Konfiguration.....	5
1.5	PHP Module installieren (SSH).....	6
1.6	NagiosQL installieren.....	6
1.7	NagiosQL Installer (Neuinstallation).....	7
1.8	NagiosQL Update.....	12
2	NagiosQL Konfiguration.....	15
2.1	Grundkonfiguration.....	15
2.2	Domänenkonfiguration (Pfade).....	16
2.2.1	Datendomänen.....	16
2.2.2	Konfigurationsdomäne.....	18
2.3	Fernzugriff einrichten (FTP oder SSH/SFTP).....	20
2.3.1	Fernzugriff via FTP.....	20
2.3.2	Fernzugriff via SSH/SFTP.....	20
3	Nagios Konfiguration.....	22
3.1	Nagios Konfigurationsdatei.....	22
3.2	Zusätzliche Berechtigungen.....	22
3.2.1	Prüfbefehl.....	22
3.2.2	Konfigurationseditor.....	23
3.2.3	Restartbefehl.....	23
3.3	Supportseite.....	23
3.3.1	Allgemeine Prüfungen.....	23
3.3.2	Konfigurationsprüfungen.....	24

# 1 Installation

## 1.1 Voraussetzungen

NagiosQL benötigt als Webapplikation eine lauffähige Webserverumgebung mit PHP sowie einen Datenbankserver zum Betrieb.

- Apache Webserver
- MySQL Server Version 4.1 oder höher
- PHP Version 5.2 oder höher
- Nagios 2.x oder 3.x
- Spezielle PHP Module:
  - Session
  - MySQL (php5-mysql)
  - gettext
  - filter
  - ftp (optional – für FTP Verbindung zu entfernten Servern)
  - SSH (PECL Modul – optional – für SSH/SCP Verbindung zu entfernten Servern)

## 1.2 Vorbereitung Verzeichnisstruktur

NagiosQL legt die Nagios-Konfigurationsdateien in einer eigenen Verzeichnisstruktur ab. Diese sollte vorgängig durch einen Superuser (root) angelegt werden. Wo genau dies zu tun ist, bleibt dem Administrator überlassen, in dieser Anleitung werden wir jeweils das Verzeichnis **/etc/nagiosql** verwenden. Zu beachten gilt, dass diese Verzeichnisstruktur später auch korrekt in der Nagios Konfigurationsdatei **nagios.cfg** eingetragen werden muss, damit Nagios selbst seine Konfigurationsdateien darin findet.

Diese Verzeichnisstruktur muss auf dem Server eingerichtet werden, auf dem Nagios selbst installiert ist.

```
# mkdir /etc/nagiosql
# mkdir /etc/nagiosql/hosts
# mkdir /etc/nagiosql/services
# mkdir /etc/nagiosql/backup
# mkdir /etc/nagiosql/backup/hosts
# mkdir /etc/nagiosql/backup/services
```

NagiosQL wird so alle Konfigurationsdateien nach **/etc/nagiosql** speichern. Hostkonfigurationen werden nach **/etc/nagiosql/hosts** gespeichert und Servicekonfigurationen nach **/etc/nagiosql/services**. In die Backupstruktur wird die jeweils alte Version einer Konfigurationsdatei abgelegt.

Da NagiosQL als Webapplikation läuft muss sichergestellt werden, dass der Webserver-Demon Schreibrechte für diese Verzeichnisse erhält, sofern es sich um eine lokale Installation handelt. Ebenfalls muss der Nagios-Demon die Dateien lesen können. Falls die Benutzer der beiden Demons nicht bekannt sind, können diese folgendermassen abgefragt werden:

```
# ps -ef | grep http
root      2999      1  0  Jul04  ?    00:00:10 /usr/sbin/httpd2-prefork -f /etc/apache2/httpd.conf
wwwrun    3123    2999  0  Jul04  ?    00:00:00 /usr/sbin/httpd2-prefork -f /etc/apache2/httpd.conf
```

Der Benutzer unter welchem der Webserver-Demon läuft heisst hier **wwwrun**.

```
# ps -ef | grep nagios
nagios    2902      1  0  Jun22  ?    00:05:39 /usr/sbin/nagios -d /etc/nagios/nagios.cfg
```

Der User unter welchem der Nagios-Demon läuft heisst hier **nagios**. Wir benötigen allerdings den Name der Gruppe vom User **nagios**.

```
# id nagios
uid=111(nagios) gid=111(nagios) groups=111(nagios)
```

Die Gruppe des Benutzers unter welchem der Nagios-Demon läuft heisst hier ebenfalls **nagios**.

So können wir die Rechte der Verzeichnisstruktur entsprechend setzen:

```
# chown -R wwwrun.nagios /etc/nagiosql
```

Alternativ können noch die Rechte der Verzeichnisstruktur auf 750 (Verzeichnisse) und 640 (cfg-Dateien) gesetzt werden.

Bei entfernten Nagios Installationen, welche via FTP oder SSH/SCP in NagiosQL verwaltet werden muss diese Verzeichnisstruktur auf dem entfernten System erstellt werden. Dort muss der zugreifende FTP oder SSH/SCP Benutzer anstelle des Benutzers vom Webserver-Demon Schreibrechte auf die Verzeichnisstruktur erhalten.

### 1.3 Nagios Konfigurationsdateien

NagiosQL benötigt auch Schreibrechte auf einige Konfigurationsdateien von Nagios selbst. Es sind dies folgende Dateien:

Nagios Hauptkonfigurationsdatei: **nagios.cfg** (Schreibrechte)  
Nagios CGI Konfigurationsdatei: **cgi.cfg** (Schreibrechte)  
Nagios Prozessdatei: **nagios.pid** oder **nagios.lock** (Leserechte)  
Nagios Commanddatei: **nagios.cmd** (Schreibrechte)

Den Speicherort dieser Dateien findet man mit dem „find“ Befehl:

```
# find / -name nagios.cfg
/etc/nagios/nagios.cfg
# find / -name cgi.cfg
/etc/nagios/cgi.cfg
# find / -name nagios.pid
/var/run/nagios.pid
# find / -name nagios.lock
/var/spool/nagios/nagios.lock
# find / -name nagios.cmd
/var/spool/nagios/nagios.cmd
```

Hier ist speziell, das sowohl der Webserver-Demon als auch der Nagios-Demon die benötigten Rechte bekommen. Dies kann auf verschiedene Arten über Benutzer/Gruppenrechte erreicht werden. Hier ein mögliches Beispiel für die lokale Installation:

```
# chown -R wwwrun.nagios /etc/nagios/nagios.cfg
# chown -R wwwrun.nagios /etc/nagios/cgi.cfg
# chown -R wwwrun.nagios /var/spool/nagios/nagios.cmd
# chmod 640 /etc/nagios/nagios.cfg
# chmod 640 /etc/nagios/cgi.cfg
# chmod 660 /var/spool/nagios/nagios.cmd
```

Leserechte auf die Prozessdatei sind in der Regel durch die Standardrechte 644 gegeben. Hier interessiert aber der korrekte Pfad für später. Die Pfade sollten alle notiert werden um sie später einfach in der NagiosQL Konfigurationsdomäne eintragen zu können.

## 1.4 Vorbereitung Apache Konfiguration

Als nächstes muss ein Webserververzeichnis vorbereitet werden in dem später die NagiosQL-Webapplikation installiert werden kann. In dieser Anleitung wird dies unter **/opt/nagiosql** geschehen.

```
# mkdir /opt/nagiosql
# chown wwwrun /opt/nagiosql
```

Folgende zwei Optionen sollten zudem im **php.ini** File für den Apache kontrolliert werden (z.Bsp. bei openSuSE/SLES unter **/etc/php5/apache/php.ini** zu finden):

```
file_uploads = On
session.auto_start = 0
date.timezone = "Europe/Zurich" (bzw. "Europe/Berlin" oder "Europe/London" etc.)
```

Nun wird eine Konfigurationsdatei für den Webserver erstellt. Wo diese Datei abgelegt wird unterscheidet sich von System zu System. Unter openSuSE/SLES werden diese Dateien z.Bsp. in **/etc/apache2/conf.d** gespeichert.

```
# touch /etc/apache2/conf.d/nagiosql.conf
```

Die Datei kann nun mit einem Editor (vi) modifiziert werden und bekommt folgenden Inhalt :

```
# NAGIOSQL SAMPLE CONFIG SNIPPETS FOR APACHE WEB SERVER
# Last Modified: 03-24-2011
#
# This file contains examples of entries that need
# to be incorporated into your Apache web server
# configuration file.  Customize the paths, etc. as
# needed to fit your system.
```

```
Alias /nagiosql "/opt/nagiosql"
```

```
<Directory "/opt/nagiosql">
    Options None
    AllowOverride None
    Order allow,deny
    Allow from all
    # Order deny,allow
    # Deny from all
    # Allow from 127.0.0.1
    # AuthName "NagiosQL Access"
    # AuthType Basic
    # AuthUserFile /etc/nagiosql/auth/nagiosql.users
    # Require valid-user
</Directory>
```

Soll später die Apache Authentifizierung verwendet werden, so sind die roten Linien zu aktivieren und die blauen Linien zu deaktivieren. Der Pfad zum AuthUserFile ist entsprechend anzupassen. Ausser Basisauthentifizierung kann natürlich jede vom Apache unterstützte Authentifizierungsmethode verwendet werden (LDAP/Kerberos/MySQL/etc.) – die entsprechenden Optionen dazu sind in der Apache Dokumentation zu finden.

Zusätzlich ist noch zu überprüfen, ob die Apache Konfiguration diese \*.conf Dateien überhaupt einliest. Dies könnte z.Bsp. via http.conf oder in einer passenden VirtualHost Konfiguration geschehen:

```
Include /etc/apache2/conf.d/nagiosql.conf
```

Nicht vergessen das Webverzeichnis zu erstellen und die neue Konfiguration zu testen sowie zu laden:

```
# mkdir /opt/nagiosql
# rcapache2 configtest
# rcapache2 restart
```

## 1.5 PHP Module installieren (SSH)

Wenn PHP Module fehlen (Voraussetzungen oder Installer), so müssen diese nachinstalliert werden. In den meisten Fällen ist dies problemlos mit den Werkzeugen möglich, welche von der Distribution mitgeliefert werden (YaST/apt-get/rpm). Ein Spezialfall stellt hier das SSH Modul dar. Nachfolgend eine Anleitung, wie SSH für NagiosQL eingerichtet wird.

Um das SSH Modul zu installieren wird das Systemtool PECL benötigt. Dieses kann normalerweise aus der Distribution nachinstalliert werden (YaST/apt-get/rpm).

Ist PECL installiert wird dieses wie folgt aufgerufen:

```
# pecl
```

Ohne Parameter wird die Hilfe ausgegeben. Sofern sich das System hinter einem Proxyserver befindet, muss dieser in PECL evtl. noch konfiguriert werden:

```
# pecl config-set http_proxy http://proxy.myserver.tld:8080
# pecl config-show
```

Nun kann das SSH Modul installiert werden:

```
# pecl install ssh2
```

Anschliessend muss das Modul noch in die php.ini eingebunden und Apache neu gestartet werden :

```
extension = ssh2.so
# rcapache2 restart
```

Die Einrichtung eines Benutzers für SSH erfolgt später in dieser Anleitung.

## 1.6 NagiosQL installieren

Nun kann NagiosQL im neuen Webverzeichnis installiert werden, dazu muss es nach /opt (nicht /opt/nagiosql) kopiert und dort entpackt werden:

```
# cd /opt
# gunzip -c nagiosql_320.tar.gz | tar xf -
# mv nagiosql32 nagiosql
# chown -R wwwrun /opt/nagiosql
# chmod 750 /opt/nagiosql/config
```

Nun sollte NagiosQL via Browser aufrufbar sein:

**<http://localhost/nagiosql>**

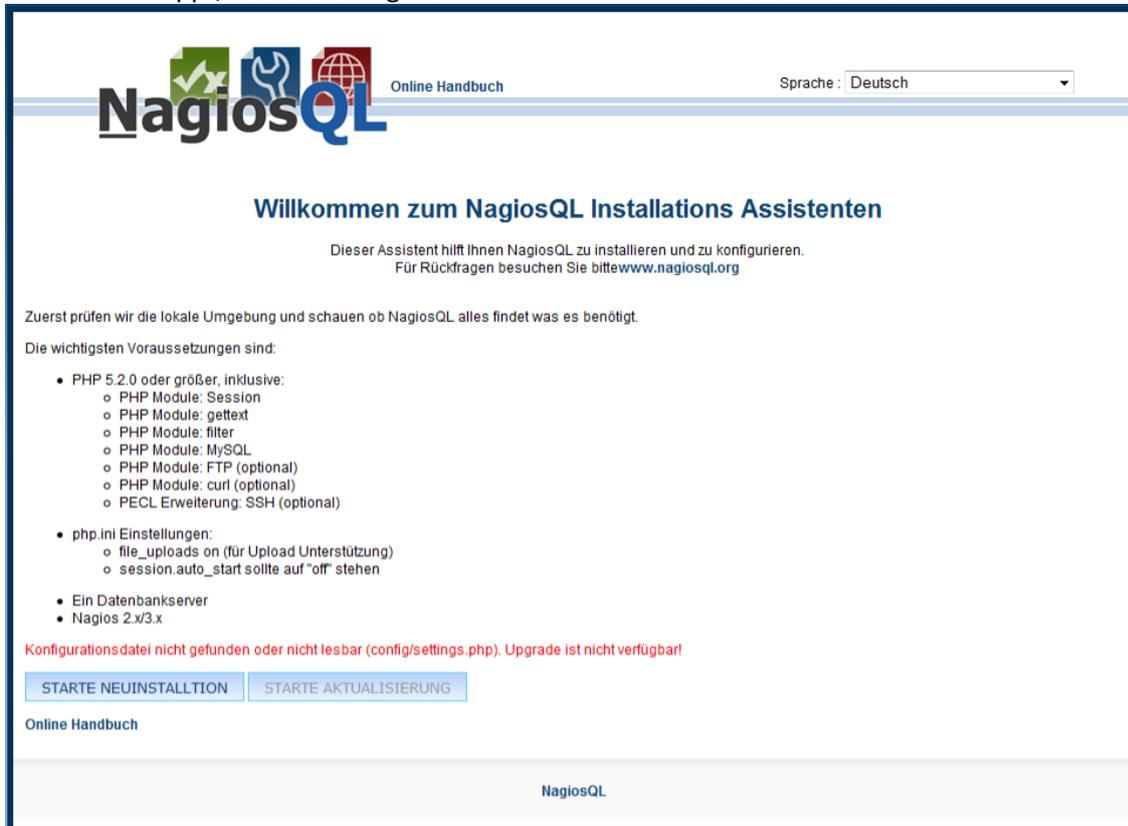
**<http://mein.virtualhost.tld/nagiosql>**

## 1.7 NagiosQL Installer (Neuinstallation)

Normalerweise wird der Installer beim ersten Aufrufen von NagiosQL automatisch gestartet. Ein manueller Aufruf ist aber ebenso möglich via:

<http://localhost/nagiosql/install/index.php>

Wenn alles klappt, sollte nun folgendes Bild im Browser erscheinen:



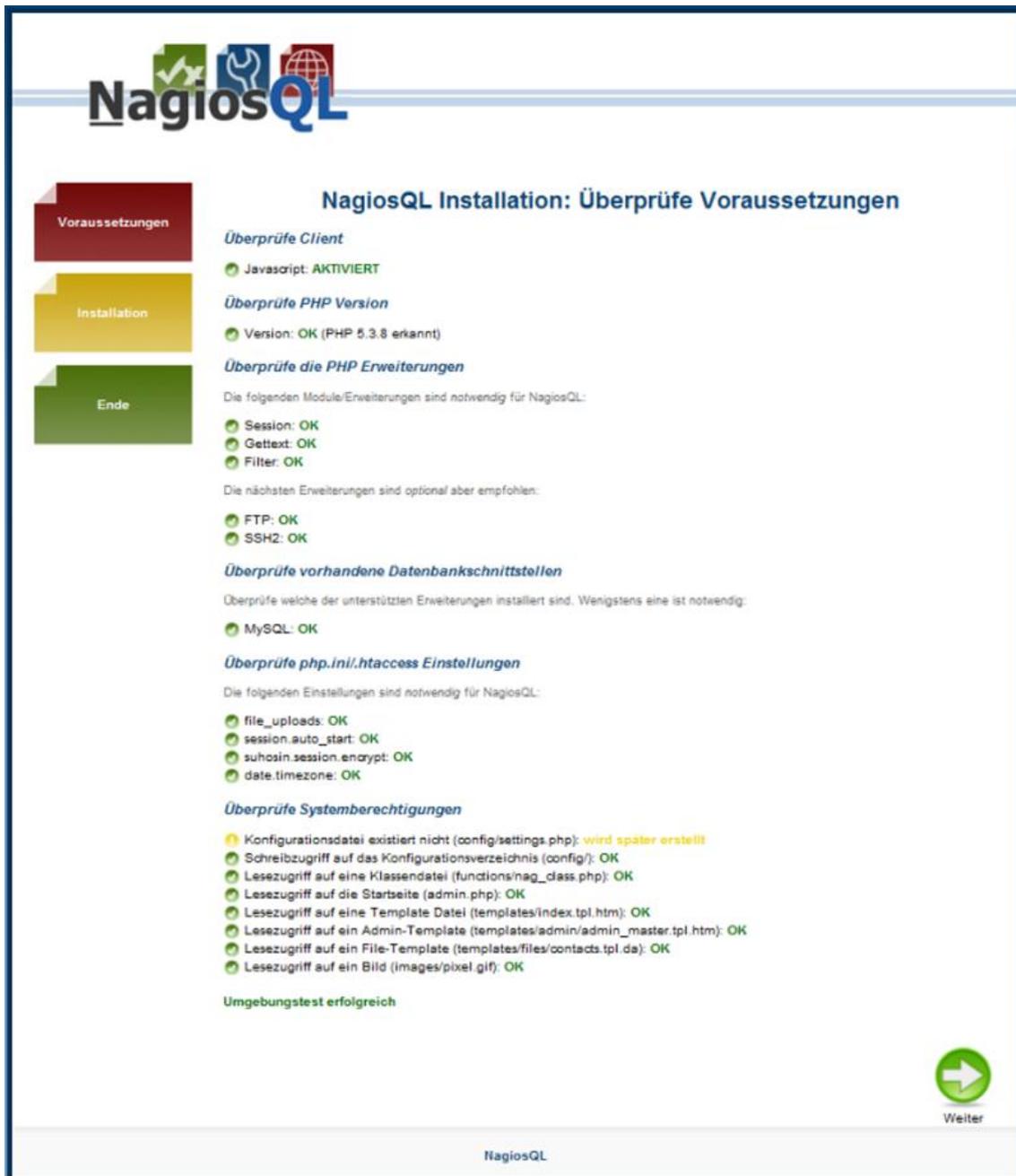
Diese Seite zeigt lediglich eine kurze Liste über die aktuellen Voraussetzungen für die Installation von NagiosQL. Zusätzlich ist es möglich die Sprache auszuwählen.

**STARTE AKTUALISIERUNG** ist nur verfügbar, wenn die gültige settings.php einer Vorgängerversion ins Verzeichnis /opt/nagiosql/config kopiert wurde. Die Aktualisierung wird später unter Punkt 1.8 beschrieben.

Mit **STARTE NEUINSTALLATION** geht es weiter.

Anschließend erfolgen die Umgebungstests, diese sind in 7 Bereiche unterteilt.

Folgendes Bild zeigt eine lauffähige Umgebung:



### Überprüfe Client:

Hier wird geprüft, ob der Browser JavaScript verarbeitet und ob dieses aktiviert ist. Einige Browser AddOns können JavaScript deaktivieren – sollten solche installiert sein, muss JavaScript für die NagiosQL Seite explizit aktiviert werden.

### Überprüfe PHP Version:

Hier wird getestet, ob die installierte PHP Version aktuell genug ist. Sollte diese Prüfung fehlschlagen, muss das System mit einer aktuelleren PHP Version aktualisiert werden. Siehe Systemdokumentation.

### Überprüfe PHP Erweiterungen:

Die ersten PHP Erweiterungen sind Pflicht. Fehlende müssen zwingend nachinstalliert werden um NagiosQL zu benutzen.

Die optionalen Erweiterungen müssen nur installiert werden, wenn die entsprechenden Optionen in NagiosQL genutzt werden.

### Überprüfe Datenbankschnittstelle:

NagiosQL benötigt zwingend eine Datenbankschnittstelle innerhalb PHP (Z.Bsp. Modul MySQL)

### Überprüfe php.ini/.htaccess Einstellungen:

Es wird geprüft, ob die notwendigen php.ini Einstellungen korrekt sind (siehe Punkt 1.4).

### Überprüfe Systemberechtigungen:

Es wird geprüft, ob alle NagiosQL Dateien lesbar/beschreibbar sind. Allenfalls sind hier noch Berechtigungen zu korrigieren. Wenn genau nach dieser Anleitung vorgegangen wurde, sollten aber alle Berechtigungen korrekt sein.

Aktualisieren kann man den Umgebungstest indem die [install/index.php](#) Seite aufgerufen und **STARTE NEUINSTALLATION** erneut gewählt wird.

Wenn alles richtiggestellt wurde, kann mit „Weiter“ die nächste Seite aufgerufen werden:

**NagiosQL**

## NagiosQL Installation: Einrichtung

Bitte vervollständigen Sie das u.a. Formular. Pflichtfelder sind mit einem \* markiert.

**Datenbank Konfiguration**

Datenbanktyp	mysql
Datenbankserver *	localhost
Lokaler Hostname oder IP-Adresse *	localhost
Datenbankserver Port *	3306
Datenbankname *	db_nagiosql_v32
NagiosQL Datenbank Benutzer *	nagiosql_user
NagiosQL Datenbank Passwort *	.....
Administrativer Datenbankbenutzer *	root
Administratives Datenbankpasswort *	.....
Lösche Datenbank, sofern sie bereits existiert?	<input checked="" type="checkbox"/>

**NagiosQL Benutzer Einrichtung**

Erster NagiosQL Benutzer *	admin
Erstes NagiosQL Passwort *	.....
Bitte Passwort wiederholen *	.....

**Nagios Konfiguration**

Importiere Nagios Beispiel Konfiguration?

**NagiosQL Verzeichnisdaten**

Erstelle NagiosQL Verzeichnisse?

NagiosQL Konfigurationsverzeichnis	/etc/nagiosql
Nagios Konfigurationsverzeichnis	/etc/nagios

Beide Verzeichnisdefinitionen werden in der Datenbank beim Konfigurationsziel "localhost" eingetragen. Wenn die Option "Erstelle NagiosQL Verzeichnisse" ausgewählt ist müssen Sie sicherstellen, dass das NagiosQL Konfigurationsverzeichnis existiert und der Webserver Schreibrechte darauf besitzt. Dann wird der Installer die benötigten Unterverzeichnisse im Dateisystem von "localhost" erstellen (hosts, services, backup etc.)

**Weiter**

Auf dieser Seite werden die Daten zum Datenbankserver eingetragen.

**Datenbanktyp:** Auswahl des Datenbanktyps (in 3.2.0 noch inaktiv)

**Datenbankserver:** Hostname des Datenbankservers

**Lokaler Hostname oder IP-Adresse:** Hostname oder IP des Webservers

*Wichtig: wenn der Datenbankserver nicht auf demselben Host läuft wie der Webserver wird der Eintrag „Lokaler Hostname“ verwendet um den Zugriff (woher) beim Datenbankserver festzulegen!*

*(Zugriff erlauben für: nagiosq\_user@webserver)*

**Datenbankserver Port:** TCP Port des Datenbankserver Demons

**Datenbankname :** Name der Datenbank, welche für NagiosQL erzeugt werden soll

**NagiosQL Datenbank Benutzer:** Benutzername für die NagiosQL Datenbank

**NagiosQL Datenbank Passwort:** Passwort für den NagiosQL Datenbank Benutzer

*Der NagiosQL Datenbankbenutzer wird bei einer Neuinstallation noch nicht existieren und wird anschliessend entsprechend den hier eingetragenen Daten erzeugt.*

**Administrativer Datenbankbenutzer:** Admin Benutzername

**Administratives Datenbankpasswort:** Admin Passwort

*Der administrative Datenbankbenutzer muss existieren und genug Rechte besitzen um eine neue Datenbank anzulegen sowie Benutzer einzurichten die auf die neue DB Zugriff erhalten (z.Bsp. root User). Die Daten des administrativen Datenbankbenutzers werden in NagiosQL nicht gespeichert – sie werden nur einmalig benötigt um die Datenbank anzulegen.*

**Lösche Datenbank:** Eine gleichnamige Datenbank zuerst löschen (**Vorsicht**)

**Erster NagiosQL Benutzer:** NagiosQL Adminbenutzername (admin)

**Erstes NagiosQL Passwort:** NagiosQL Adminpasswort

*Administrativer Benutzer für das Login in die NagiosQL Webapplikation. Dieser Benutzer kann später nicht mehr gelöscht werden*

„Importiere Nagios Beispiel Konfiguration“ würde die Datenbank bereits mit einigen Beispielesdaten abfüllen.

„Erstelle NagiosQL Verzeichnisse“ erstellt eine Verzeichnisstruktur wie unter Punkt 1.2 beschrieben, was im Fall dieser Anleitung nicht mehr nötig ist. Voraussetzung dafür wäre, dass das NagiosQL Konfigurationsverzeichnis für den Webserver-Demon beschreibbar ist. Anpassung von UID/GID sind aber nicht möglich und müssen später von Hand vorgenommen werden.

**NagiosQL Konfigurationsverzeichnis:** Basisverzeichnis für die Konfigurationsdaten

**Nagios Konfigurationsverzeichnis:** Basisverzeichnis für die Konfiguration von Nagios  
(enthält nagios.cfg)

*Diese Daten werden verwendet um die „localhost“ Domäne vorzukonfigurieren.*

Wurden alle Daten korrekt eingetragen und die Datenbank konnte erzeugt werden erfolgt folgende Ausgabe:

**NagiosQL**

**NagiosQL Installation: Fertigstellen**

**Erstelle NagiosQL MySQL Benutzer**

Datenbankserver Verbindung (privilegierter Benutzer)	funktioniert
Datenbankserver Version	5.5.16-log unterstützt
Datenbankserver Unterstützung	unterstützt
Bestehende NagiosQL 3 Datenbank löschen	erledigt (db_nagiosql_v32)
Erstelle neue Datenbank	erledigt (db_nagiosql_v32)
Erstelle NagiosQL Datenbankbenutzer	erledigt (Nur Rechte zum bestehenden Benutzer hinzugefügt: nagiosql_user)
Installiere NagiosQL Datenbanktabellen	erledigt
Setze den NagiosQL Administrator	erledigt
Datenbankserver Verbindung (NagiosQL Benutzer)	funktioniert

**Setze NagiosQL Einstellungen**

Schreiben der Einstellungen in die Datenbank	erledigt
Schreibe Datenbank Konfiguration in settings.php	erledigt
Import Nagios Beispiel Daten	erledigt
Erstelle und/oder speichere NagiosQL Verzeichnisse	erledigt (Überprüfen Sie die Rechte der Verzeichnisse!)

Bitte löschen Sie den Installationsordner bevor Sie fortfahren!

Ende

NagiosQL

Anschließend sollte das Verzeichnis `/opt/nagiosql/install` gelöscht werden.

Nach erfolgreicher Installation müssen innerhalb NagiosQL einige Grundkonfigurationen überprüft oder gesetzt werden. Das nächste Kapitel beschäftigt sich mit diesem Thema.

## 1.8 NagiosQL Update

Beim Update wird identisch vorgegangen wie bei der Installation unter Punkt 1.7. Zu beachten dabei ist lediglich, dass die Datei **config/settings.php** der alten NagiosQL Installation erhalten bleibt und in das config Verzeichnis der neuen Installation kopiert wird.

**Niemals dürfen die Dateien einer neuen Version in das vorhandene Verzeichnis mit den alten Dateien der vorgängigen NagiosQL Version kopiert werden!**

**Sehr wichtig bei einer Aktualisierung ist es, die bestehende Datenbank vorgängig zu sichern!** Schlägt beim Update etwas fehl, kann das Update **nicht mehr wiederholt werden** und die Datenbank ist defekt in einem undefinierten Zustand. Dann hilft meist nur noch eine Neuinstallation mit anschliessendem Import aller Konfigurationsdaten.

Ein Backup kann man mit gängigen Tools wie phpMyAdmin sehr einfach erstellen. Alternativ gibt es auch Commandline Tools wie mysqldump.



Mit **STARTE AKTUALISIERUNG** geht es weiter.

Anschliessend erfolgen dieselben Umgebungstests wie bei der Neuinstallation. Die Seite wird daher hier nicht erneut behandelt.

Wenn in den Umgebungstests alles richtiggestellt wurde, kann mit „Weiter“ die nächste Seite aufgerufen werden:

**NagiosQL Update: Einrichtung**

Bitte vervollständigen Sie das u.a. Formular. Pflichtfelder sind mit einem \* markiert.

**Bitte sichern Sie Ihre Datenbank bevor Sie fortfahren!**

**Datenbank Konfiguration**

Datenbanktyp	mysql
Datenbankserver *	localhost
Lokaler Hostname oder IP-Adresse *	localhost
Datenbankserver Port *	3306
Datenbankname *	db_nagiosql_v31
Administrativer Datenbankbenutzer *	root
Administratives Datenbankpasswort *	

[Weiter](#)

NagiosQL

Damit das Update wie gewünscht startet, müssen hier lediglich die Daten des administrativen Datenbankbenutzers eingetragen werden. Die anderen Daten sind zwar auch änderbar, sollten aber nur dann angefasst werden, wenn man ganz sicher ist, was man tut.

**Administrativer Datenbankbenutzer:** Admin Benutzername

**Administratives Datenbankpasswort:** Admin Passwort

*Der administrative Datenbankbenutzer muss existieren und genug Rechte besitzen um eine neue Datenbank anzulegen sowie Benutzer einzurichten die auf die neue DB Zugriff erhalten (z.Bsp. root User). Die Daten des administrativen Datenbankbenutzers werden in NagiosQL nicht gespeichert – sie werden nur einmalig benötigt um die Datenbank anzulegen.*

Wurden alle Daten korrekt eingetragen und die Datenbank konnte aktualisiert werden erfolgt folgende Ausgabe:

**NagiosQL Update: Fertigstellen**

**Voraussetzungen**

**Update**

**Ende**

**Aktualisiere bestehende NagiosQL Datenbank**

Datenbankserver Verbindung (privilegierter Benutzer)	funktioniert
Installierte NagiosQL Version	unterstützt (3.1.1)
Aktualisiere von Version 3.1.1 nach 3.2.0	erledigt
Konvertiere Datenbank zu UTF8	erledigt
Konvertiere Datenbanktabellen zu UTF8	erledigt
Konvertiere Datenbankfelder zu UTF8	erledigt
Datenbankserver Verbindung (NagiosQL Benutzer)	funktioniert

**Setze NagiosQL Einstellungen**

Schreiben der Einstellungen in die Datenbank	erledigt
Schreibe Datenbank Konfiguration in settings.php	erledigt

**Bitte löschen Sie den Installationsordner bevor Sie fortfahren!**

**Ende**

NagiosQL

Anschließend sollte das Verzeichnis `/opt/nagiosql/install` gelöscht werden.

# 2 NagiosQL Konfiguration

## 2.1 Grundkonfiguration

Die NagiosQL Grundkonfiguration ist in der Weboberfläche unter **Verwaltung – Einstellungen** zu finden. Hier sind auch einige der Daten zu finden, welche zuvor schon im Installer eingetragen wurden. Über das blaue Fragezeichensymbol kann jeweils eine kurze Hilfe zum entsprechenden Feld aufgerufen werden.

The screenshot shows the NagiosQL configuration page titled "NagiosQL Systemeinstellungen konfigurieren". The interface is in German. On the left is a navigation menu with "Verwaltung" selected. The main content area is divided into several sections:

- Pfad:** Temporäres Verzeichnis \* (set to /tmp), Server Protokoll \* (set to http).
- Sprache:** Sprache (set to Deutsch), Encoding (set to utf-8).
- Datenbank:** MySQL Server \* (set to localhost), MySQL Server Port \* (set to 3306), Datenbankname \* (set to db\_nagiosql\_v32), Datenbankbenutzer \* (set to nagiosql\_user), Datenbankpasswort (masked with dots).
- Sicherheit:** Automatisches Ausloggen nach (set to 3600), Authentifikation\* (set to NagiosQL).
- Allgemeines:** Zeilen pro Seite (set to 15), Auswahl methode (set to NagiosQL 3), Warnungen bei Templatewerten (radio buttons for Einschalten and Ausschalten), Online Versionsüberprüfung (radio buttons for Einschalten and Ausschalten), Proxyserver (radio buttons for Einschalten and Ausschalten), Proxyadresse \* (empty), Proxy Benutzername (optional) (empty), Proxy Passwort (optional) (empty).

At the bottom, there are "Speichern" and "Abbrechen" buttons, and a note "\* Pflichtfelder". The version "NagiosQL 3.2.0" is displayed at the bottom center.

*Erwähnenswert sind hier folgende Werte:*

- Temporäres Verzeichnis:** Das Verzeichnis in dem NagiosQL temporäre Daten ablegt. In der Regel wird die **/tmp** sein.
- Server Protokoll:** Wenn NagiosQL auf einer SSL Umgebung installiert wird kann hier entsprechend **https** gewählt werden. Achtung, eine Falscheingabe kann die Funktion beeinträchtigen/beenden!
- Sprache:** Hier kann die NagiosQL Oberfläche generell auf eine andere Sprache umgestellt werden. Manchmal muss der Browsercache geleert, der Browser geschlossen /geöffnet oder gar der Apache neu gestartet damit die neue Sprache durchgängig angezeigt wird.
- Authentifikation:** Soll die Apache Authentifikation verwendet werden, so muss diese hier generell eingeschaltet werden.
- Zeilen pro Seite:** Hier kann die Anzahl Zeilen für die Listenansichten definiert werden.

## Online Versionsprüfung:

Es wird innerhalb NagiosQL Supportseite angezeigt, ob eine neue NagiosQL Version verfügbar ist.  
In Umgebungen mit Proxyserver müssen allenfalls zusätzlich die Daten des Proxyservers anschliessend eingetragen werden.

## 2.2 Domänenkonfiguration (Pfade)

Die einzelnen Domänen und Pfade der gesteuerten Nagios-Installationen werden über die Domänenadministration festgelegt. Diese befindet sich unter **Verwaltung**.

Es existieren zwei Arten von Domänen:

### 2.2.1 Datendomänen

Datendomänen werden zur Datenhaltung verwendet, in Ihnen werden die Konfigurationsobjekte definiert. Das Auswahlfeld oben links in der Ansicht bezieht sich immer auf die gerade aktive Datendomäne.

	Datendomäne	Beschreibung	Aktiv	Funktion
<input type="checkbox"/>	common	administrative global domain	Ja	  
<input type="checkbox"/>	localhost	Local installation	Ja	  

Hinzufügen Markierte:

*Es gibt zwei vorgegebene Datendomänen die nicht gelöscht oder umbenannt werden können:*

#### Common Datendomäne

Eine allgemeine Datendomäne die in jeder anderen Datendomäne eingebunden werden kann. Dies kann zum Beispiel für Kontakte, Zeitperioden oder Befehle sinnvoll sein, welche in jeder Datendomäne identisch bleiben. Hier stehen nur sehr wenige Einstellungen zur Verfügung.

#### Localhost Datendomäne

Die lokale Datendomäne des eigenen Servers. Ausser der Tatsache, dass diese Datendomäne nicht gelöscht oder umbenannt werden kann gibt es hier keinen Unterschied zu einer anderen Datendomäne.

## Konfigurationsoptionen

Verwaltung

Verwaltung -> Verwaltung -> Datendomäne

Domäne: localhost

Eingeloggt: admin [Abmelden](#)

### Datendomänen

Domäne \* localhost ?

Beschreibung \* Local installation

Konfigurationsziel \* localhost ?

Nagios Version 3.0 + ?

Benutze die globale Domäne Einschalten ?

Zugriffsgruppe Uneingeschränkter Zugriff ?

Aktiv

[Speichern](#) [Abbrechen](#) \* Pflichtfelder

[Menu ausblenden]

NagiosQL 3.2.0

- Domäne:** Interner Name der Datendomäne (wird in den Konfigurationsübersichten so angezeigt)
- Beschreibung:** Nähere Beschreibung der Datendomäne
- Konfigurationsziel:** Auswahl der zugeteilten Konfigurationsdomäne (siehe 2.2.2)
- Nagios Version:** Version der Nagiosinstallation (Version 2.x oder 3.x werden unterstützt) – je nach Version werden zusätzliche Optionen in den Übersichten angezeigt.
- Globale Domäne:** Hier kann die globale Datendomäne ein- oder ausgeschaltet werden. Sobald die globale Datendomäne eingeschaltet ist und dort Objekte definiert sind, werden diese Objekte auch in die Konfigurationsdateien der aktuellen Datendomäne geschrieben.
- Zugriffsgruppe:** Hier kann definiert werden, welche Benutzergruppe Zugriff zu dieser Datendomäne bekommt
- Aktiv:** Datendomäne aktivieren oder deaktivieren

## 2.2.2 Konfigurationsdomäne

Konfigurationsdomänen werden zur Definition einer Nagiosinstallation (Site) verwendet, in Ihnen werden die Pfade und Verbindungseinstellungen definiert.

### Konfigurationsoptionen

- Domäne:** Interner Name der Konfigurationsdomäne (wird in der Datendomäne - Konfigurationsziel so angezeigt)
- Beschreibung:** Nähere Beschreibung der Domäne
- Konfigurationsziel:** Auswahl der zugewiesenen Konfigurationsdomäne (siehe 2.2.2)
- Servername:** Hostname des Systems auf welchem sich die Nagios Installation dieser Domäne befindet - muss auf eine IP Adresse auflösbar sein
- Methode:** Die Verbindungsart wie NagiosQL auf die Konfigurationsdateien zugreifen soll. Zur Auswahl stehen:
- Fileaccess Die Dateien befinden sich im lokalen Dateisystems des Servers oder sind über dieses zugreifbar (NFS/SMB/CIFS etc.)
  - FTP Der Zugriff auf die Konfigurationsdateien erfolgt über FTP
  - SSH/SFTP Der Zugriff auf die Konfigurationsdateien erfolgt über SSH/SFTP
- Für FTP und SSH/SFTP folgt im nächsten Kapitel (2.3) eine Kurzanleitung zur Einrichtung dieser Verbindungsarten*
- Benutzername:** Benutzername zum Login auf dem Zielsystem (Nur bei FTP oder SSH/SFTP)
- Passwort:** Passwort zum Login auf dem Zielsystem (Nur bei FTP oder SSH/SFTP)
- Schlüsselpaar:** Lokales Verzeichnis in dem der SSH Key abgelegt ist (Nur bei SSH/SFTP)

<b>Basisverzeichnis:</b>	Basisverzeichnis für Nagios Konfigurationsdateien. Alle Konfigurationsdateien ausser für Hosts und Services werden in dieses Verzeichnis geschrieben. <i>/etc/nagiosql</i>
<b>Hostverzeichnis:</b>	Verzeichnis für die Nagios Host Konfigurationsdateien <i>/etc/nagiosql/hosts</i>
<b>Serviceverzeichnis:</b>	Verzeichnis für die Nagios Service Konfigurationsdateien <i>/etc/nagiosql/services</i>
<b>Backupverzeichnis:</b>	Verzeichnis in welches NagiosQL Backupversionen von Nagios Konfigurationsdateien (ausser Hosts/Services) ablegt <i>/etc/nagiosql/backup</i>
<b>Host Backupverz.:</b>	Verzeichnis für NagiosQL Backupversionen von Hostkonfigurationsdateien <i>/etc/nagiosql/backup/hosts</i>
<b>Service Backupverz.:</b>	Verzeichnis für NagiosQL Backupversionen von Servicekonfigurationsdateien <i>/etc/nagiosql/backup/services</i>
<b>Nagios Basisverz.:</b>	Basisverzeichnis der Nagiosinstallation. Die Datei <b>nagios.cfg</b> und die Datei <b>cgi.cfg</b> ist in diesem Verzeichnis abgelegt und werden von NagiosQL dort erwartet! <i>/etc/nagios</i>
<b>Importverzeichnis:</b>	Alternatives Importverzeichnis. Konfigurationsdateien in diesem Verzeichnis werden zusätzlich zum Import angeboten <i>/etc/nagios/objects</i>
<b>Nagios Bilderverz.:</b>	Alternatives Bilderverzeichnis. Üblicherweise ist der Bilderpfad fest in Nagios definiert (z.Bsp. <i>/usr/local/nagios/share/images/logos</i> ). Hier können ab diesem Pfad zusätzliche feste Unterverzeichnisse ab diesem Verzeichnispunkt angegeben werden (relativer Pfad), welche dann in den einzelnen Konfigurationen nicht mehr jedes Mal eingetragen werden müssen.
<b>Nagios Command:</b>	Pfad zur Nagios Command Datei ( <i>nagios.cmd</i> ). Diese Datei wird benötigt um Befehle an den Nagiosdaemon zu senden. Achtung, die Datei wird nur erzeugt /gelesen, wenn <b>check_external_commands=1</b> in der <b>nagios.cfg</b> definiert ist! <i>/var/nagios/rw/nagios.cmd</i>
<b>Nagios Binary:</b>	Pfad zur Nagios Binarydatei <i>/opt/nagios/bin/nagios</i>
<b>Nagios Prozessdatei:</b>	Pfad zur Nagios Prozessdatei (PID oder LOCK File) <i>/var/nagios/nagios.lock</i>
<b>Nagios Konfiguration:</b>	Pfad zur <i>nagios.cfg</i> <i>/etc/nagios/nagios.cfg</i>
<b>Nagios Version:</b>	Version der Nagiosinstallation (Version 2.x oder 3.x werden unterstützt) – je nach Version werden zusätzliche Optionen in den Übersichten angezeigt.
<b>Zugriffsgruppe:</b>	Hier kann definiert werden, welche Benutzergruppe Zugriff zu dieser Domäne bekommt
<b>Aktiv:</b>	Domäne aktivieren oder deaktivieren

## 2.3 Fernzugriff einrichten (FTP oder SSH/SFTP)

### 2.3.1 Fernzugriff via FTP

Auf dem lokalen System muss das PHP Modul **ftp** installiert sein.

Auf dem Zielsystem muss ein FTP Server installiert werden. Zusätzlich wird ein eigener Benutzer für NagiosQL eingerichtet.

Die Dateistruktur für NagiosQL wie unter Punkt 1.2 beschrieben wird auf dem Zielsystem erstellt. Ebenso müssen die Berechtigungen der Nagios Konfigurationsdateien auf dem entfernten System entsprechend Punkt 1.3 gesetzt werden. Anstelle des Webserver-Demon muss hier der NagiosQL FTP Benutzer Schreibrecht bekommen.

Die Benutzerdaten des NagiosQL FTP Benutzers werden in der Domänenkonfiguration (Punkt 2.2) bei Benutzernamen und Passwort eingetragen.

Methode	<input type="text" value="FTP"/>
Benutzername*	<input type="text" value="nagiosql"/>
Passwort*	<input type="password" value="....."/>

Es wird empfohlen die FTP Verbindung vorgängig manuell mit einem FTP Clienten zu testen.

Via FTP sind die Funktionen unter „**Werkzeuge – Nagios steuern**“ nur eingeschränkt nutzbar. „**Konfigurationsdateien prüfen**“ und „**Nagios neu starten**“ funktioniert unter FTP nur bei einem Server der REMOTE EXEC unterstützt – dies ist aus Sicherheitsgründen heute fast nirgends mehr gegeben.

### 2.3.2 Fernzugriff via SSH/SFTP

Auf dem lokalen System muss das PHP Modul **ssh2** installiert sein (Punkt 1.4).

Auf dem Zielsystem muss ein SSH Server installiert werden. Allfällige Zugriffsbeschränkungen (Firewall etc.) sind für den Zugriff vom lokalen System zu entfernen.

Auf dem Zielsystem ist ein neuer Benutzer für NagiosQL zu erstellen, als Beispiel wird hier `nagiosql_usr` verwendet. Danach wird wie folgt ein ssh Schlüsselpaar erstellt.

#### Lokales System:

1. Es wird ein Pfad für die SSH Keys erstellt:  

```
# mkdir /etc/nagiosql/ssh
```
2. Nun wird ein Schlüsselpaar erstellt – nur `rsa` wird unterstützt:  

```
# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa): /etc/nagiosql/ssh/id_rsa
-> Passwort leer lassen!
Der Name des Schlüssels muss zwingend id_rsa lauten!
```
3. Der Schlüssel wird nun auf das Zielsystem kopiert:  

```
# ssh-copy-id -i /etc/nagiosql/ssh/id_rsa/id_rsa.pub nagiosql_usr@remote-system
```
4. Der Schlüssel unter `/etc/nagiosql/ssh` muss für den Webserver-Demon lesbar sein!  

```
# chown -R wwwrun.nagios /etc/nagiosql/ssh
# chown -R wwwrun.nagios /etc/nagiosql/ssh/id_rsa
```
5. Nun sollte das Login manuell geprüft werden:  

```
# ssh -i /etc/nagiosql/ssh/id_rsa nagiosql_usr@remote-system
-> Login muss ohne Passwort möglich sein!
```

Der Benutzername des NagiosQL SSH Benutzers (nagiosql\_usr) wird in der Domänenkonfiguration (Punkt 2.2) bei Benutzername eingetragen. In das Feld für den SSH Verzeichnisschlüssel wird das Verzeichnis mit dem Schlüssel eingetragen (/etc/nagiosql/ssh/).

Methode	SSH/SFTP 
Benutzername*	nagiosql_usr
Passwort*	<input type="password"/>
Verzeichnis zum SSH Schlüsselpaar*	/etc/nagiosql/ssh/ 

Das Feld Passwort kann bei Authentifizierung mittels Key frei bleiben – es wird nicht verwendet.

Eine Authentifizierung über Benutzername und Passwort (ohne Key) ist auch möglich, aber explizit nicht empfohlen, da diese generell störungsanfälliger ist. Sollte diese Variante trotzdem verwendet werden, muss in der **sshd\_config** des Zielsystems folgender Parameter gesetzt werden:

```
PasswordAuthentication yes
```

# 3 Nagios Konfiguration

## 3.1 Nagios Konfigurationsdatei

Die Nagios Konfigurationsdatei **nagios.cfg** muss soweit angepasst werden, dass Nagios die von NagiosQL geschriebenen Konfigurationen findet und verwenden kann. Dies kann mit dem integrierten Editor von NagiosQL angepasst werden, sofern die Zugriffsrechte in Punkt 1.3 richtig gesetzt wurden.

Wichtig sind die Pfade der Standardkonfigurationsdateien:

```
cfg_file=/etc/nagiosql/contacttemplates.cfg
cfg_file=/etc/nagiosql/contactgroups.cfg
cfg_file=/etc/nagiosql/contacts.cfg
cfg_file=/etc/nagiosql/timeperiods.cfg
cfg_file=/etc/nagiosql/commands.cfg
cfg_file=/etc/nagiosql/hostgroups.cfg
cfg_file=/etc/nagiosql/servicegroups.cfg
```

```
cfg_dir=/etc/nagiosql/hosts
cfg_dir=/etc/nagiosql/services
```

Sowie optionale Konfigurationsdateien, wenn diese verwendet werden:

```
cfg_file=/etc/nagiosql/hosttemplates.cfg
cfg_file=/etc/nagiosql/servicetemplates.cfg
cfg_file=/etc/nagiosql/servicedependencies.cfg
cfg_file=/etc/nagiosql/serviceescalations.cfg
cfg_file=/etc/nagiosql/hostdependencies.cfg
cfg_file=/etc/nagiosql/hostescalations.cfg
cfg_file=/etc/nagiosql/hostextinfo.cfg
cfg_file=/etc/nagiosql/serviceextinfo.cfg
```

Sämtliche vorhandenen Zeilen, die mit **cfg\_file=** oder **cfg\_dir=** beginnen sind **zu entfernen** oder mit einem führenden **# zu deaktivieren!** Ausser den oben angegebenen dürfen keine zusätzlichen **cfg**-Zeilen enthalten sein.

Damit NagiosQL einen Restart Befehl an Nagios schicken kann, muss Nagios selbst externe Befehle akzeptieren. Dazu muss folgende Zeile existieren:

```
check_external_commands=1
```

## 3.2 Zusätzliche Berechtigungen

Für die vollumfängliche Funktion von NagiosQL müssen einige Dateien von Nagios für NagiosQL berechtigt werden (siehe Punkt 1.3). Bei lokalen Installationen ist hier der Benutzer des Webserver-Demons gemeint – bei entfernten Installationen der NagiosQL FTP oder SSH/SCP Benutzer. Nachfolgend spricht die Anleitung vereinfacht von „Benutzer“.

Die Pfade zu diesen Dateien müssen in der Domänenkonfiguration (Punkt 2.2) eingetragen sein. Bezüglich der Rechte gibt es mehrere Möglichkeiten (Dateirechte, sudo, acl). Zudem unterscheiden sich die Installationen individuell.

### 3.2.1 Prüfbefehl

Um die aktuelle Konfiguration zu prüfen, wird NagiosQL die interne Prüfmethode von Nagios verwenden:

```
# nagios -v nagios.cfg
<nagios-binary> -v <nagios Konfigurationsdatei>
```

Der Benutzer muss Rechte zu Ausführen des Nagiosbinary besitzen und Leserechte auf die Konfigurationsdatei **nagios.cfg**.

### 3.2.2 Konfigurationseditor

Die Dateien **nagios.cfg** und **cgi.cfg** können in einem vereinfachten Editor direkt aus NagiosQL editiert werden.

Der Benutzer muss Schreibrechte auf die Konfigurationsdateien **nagios.cfg** und **cgi.cfg** besitzen.

### 3.2.3 Restartbefehl

Um eine neue Konfiguration direkt aus NagiosQL aktivieren zu können werden Rechte auf die Kommanddatei **nagios.cmd** und die Prozessdatei **nagios.lock** benötigt.

Der Benutzer muss Schreibrechte auf die Kommanddatei **nagios.cmd** besitzen und Leserechte auf die Prozessdatei **nagios.lock** bzw. **nagios.pid**.

## 3.3 Supportseite

Seit NagiosQL 3.2.0 existiert eine Supportseite, welche die wichtigsten internen Prüfungen vornimmt und das Resultat übersichtlich darstellt. Allfällige Fehlkonfigurationen können so schnell festgestellt und korrigiert werden.

### 3.3.1 Allgemeine Prüfungen

The screenshot shows the NagiosQL Support page. The navigation menu on the left includes: Hauptseite, Überwachung, Alarmierung, Befehle, Spezialitäten, Werkzeuge, Verwaltung (selected), Neues Passwort, Benutzer, Gruppen, Menu Zugriff, Datendomäne, Konfigdomäne, Logbuch, Einstellungen, Hilfe Editor, Support, and [Menu ausblenden]. The main content area is titled 'NagiosQL Supportseite' and contains sections for support information, a donation link, translation services, a version check table, and an environment test table.

Installiert	Verfügbar	Information
3.2.0	3.2.0	Sie haben bereits die letzte stabile Version installiert

Umgebungsprüfung			
Javascript	OK	PHP ini Einstellungen: file_uploads	OK
PHP Version	OK (5.3.8)	PHP ini Einstellungen: session.auto_start	OK
PHP Modul: session	OK	PHP ini Einstellungen: suhosin.session.encrypt	OK
PHP Modul: gettext	OK	PHP ini Einstellungen: date.timezone	OK (Europe/Zurich)
PHP Modul: ftp	OK	Schreib-/Leserechte: settings.php	OK
PHP Modul: ssh	OK		
MySQL Version	OK (5.5.16-log)		

Hier werden hilfreiche Links angegeben und grundlegende Umgebungsprüfungen durchgeführt. Sollte nach einem Systemupdate etwas nicht mehr korrekt funktionieren findet man hier vielleicht einen Hinweis.

Sofern eingeschaltet wird hier auch geprüft, ob die aktuellste Version von NagiosQL installiert ist.

### 3.3.2 Konfigurationsprüfungen

#### Überprüfung der Konfigurationsdomäne

Die nachstehenden Überprüfungen basieren auf den eingestellten Werten bei der Daten- und Konfigurationsdomäne. Die Datendomäne kann über das Pull-Down-Menü oben rechts geändert werden. Die Konfigurationsdomäne ist innerhalb der jeweiligen Datendomäne definiert. Wiederholen Sie diese Überprüfungen für alle Konfigurationsdomänen die Sie verwenden.

Name der Konfigurationsdomäne	localhost	Nagios Konfigurationsdatei	OK
Verbindungstyp	Fileaccess	Nagios Prozess Datei	OK
Verbindungsprüfung	OK	Nagios Command Datei	OK
		Nagios Binary Datei	OK

#### Überprüfung der Konfigurationsdateien und der Demon Konfiguration

Konfigurationsname	Benutzt in der Datendomäne	Eingebunden in der Demon Konfiguration (nagios.cfg)
Hosts	OK (4)	OK (cfg_dir=/opt/nagios/inst1/etc/nagiosql/hosts)
Services	OK (21)	OK (cfg_dir=/opt/nagios/inst1/etc/nagiosql/services)
Hostgroups	OK (4)	OK (cfg_file=/opt/nagios/inst1/etc/nagiosql/hostgroups.cfg)
Servicegroups	Nicht benutzt	OK (Definition fehlt, wird aber aktuell nicht benutzt)
Hosttemplates	OK (5)	OK (cfg_file=/opt/nagios/inst1/etc/nagiosql/hosttemplates.cfg)
servicetemplates	OK (2)	OK (cfg_file=/opt/nagios/inst1/etc/nagiosql/servicetemplates.cfg)
Contacts	OK (1)	OK (cfg_file=/opt/nagios/inst1/etc/nagiosql/contacts.cfg)
Contactgroups	OK (1)	OK (cfg_file=/opt/nagios/inst1/etc/nagiosql/contactgroups.cfg)
Timeperiods	OK (5)	OK (cfg_file=/opt/nagios/inst1/etc/nagiosql/timeperiods.cfg)
Contacttemplates	OK (1)	OK (cfg_file=/opt/nagios/inst1/etc/nagiosql/contacttemplates.cfg)
Commands	OK (24)	OK (cfg_file=/opt/nagios/inst1/etc/nagiosql/commands.cfg)
Hostdependencies	Nicht benutzt	OK (Definition fehlt, wird aber aktuell nicht benutzt)
Hostescalations	Nicht benutzt	OK (Definition fehlt, wird aber aktuell nicht benutzt)
Hostextinfo	Nicht benutzt	OK (Definition fehlt, wird aber aktuell nicht benutzt)
Servicedependencies	Nicht benutzt	OK (Definition fehlt, wird aber aktuell nicht benutzt)
Serviceescalations	Nicht benutzt	OK (Definition fehlt, wird aber aktuell nicht benutzt)
Serviceextinfo	Nicht benutzt	OK (Definition fehlt, wird aber aktuell nicht benutzt)

Hier wird geprüft ob alle relevanten Nagios Dateien verfügbar sind und die Benutzerrechte (Punkt 1.3) korrekt sind.

Der letzte Teil liest die Nagios Konfigurationsdatei ein (nagios.cfg) und prüft, ob die nötigen Pfadeinstellungen (siehe 3.1) vorgenommen wurden und mit den Pfadeinstellungen in der gewählten Konfigurationsdomäne übereinstimmen.

Generell weisen auf der Supportseite alle roten Einträge auf ein Problem hin, das behoben werden muss. Die gelben Einträge weisen auf Probleme hin, die möglicherweise später auftreten können.