

# Guacamole Install Skript

**ACHTUNG: Das folgende Skript ist ungetestet und NICHT von mir! Es wird erst noch getestet! Nicht auf produktiven Servern ausführen!**

[guacamole-install-script.sh](#)

```
#!/bin/bash
#####

# Task of this script:
# Install Packages Dependencies
# Download Guacamole and MySQL Connector packages
# Install Guacamole Server
# Install Guacamole Client
# Install MySQL Connector
# Configure MariaDB or MySQL
# Setting Tomcat Server
# Generates a Java KeyStore for SSL Support
# Install and Setting Nginx Proxy (SPDY enabled)
# Generates a Self-Signed Certificate for SSL Support
# Cofigure SELinux for Nginx Proxy
# Configure FirewallD or iptables

#####
#####   VARIABLES   #####
#####
SCRIPT=`basename ${BASH_SOURCE[0]}` #Script File Name
GUACA_VER="0.9.10-incubating"
MYSQL_CONNECTOR_VER="5.1.40"
LIBJPEG_VER="1.5.1"
SCRIPT_BUILD="1"
SCRIPT_VERSION="${GUACA_VER} Build ${SCRIPT_BUILD}"
SERVER_HOSTNAME="localhost"
INSTALL_DIR="/usr/local/src/guacamole/${GUACA_VER}/"
LIB_DIR="/var/lib/guacamole/"
PWD=`pwd`
filename="${PWD}/guacamole-${GUACA_VER}.${date +"%d-%y-%b"}"
logfile="${filename}.log"
fwbkpfile="${filename}.firewall.bkp"
MYSQ_CONNECTOR_URL="http://dev.mysql.com/get/Downloads/Connector-J/"
MYSQL_CONNECTOR="mysql-connector-java-${MYSQL_CONNECTOR_VER}"
MYSQL_PORT="3306"
GUACA_PORT="4822"
GUACA_CONF="guacamole.properties"
GUACA_URL="http://sourceforge.net/projects/guacamole/files/current/"
GUACA_SERVER="guacamole-server-${GUACA_VER}" #Source
#GUACA_CLIENT="guacamole-client-${GUACA_VER}" #Source
GUACA_CLIENT="guacamole-${GUACA_VER}" #Binary
```

```
GUACA_JDBC="guacamole-auth-jdbc-${GUACA_VER}" #Extension
LIBJPEG_URL="http://sourceforge.net/projects/libjpeg-
turbo/files/${LIBJPEG_VER}/"
#LIBJPEG_TURBO="libjpeg-turbo-${LIBJPEG_VER}" #Dependency source
LIBJPEG_TURBO="libjpeg-turbo-official-${LIBJPEG_VER}" #Dependency rpm
CENTOS_VER=`rpm -qi --whatprovides /etc/redhat-release | awk '/Version/
{print $3}'`
if [ $CENTOS_VER -ge 7 ]; then MySQL_Packages="mariadb mariadb-server";
Menu_SQL="MariaDB"; else MySQL_Packages="mysql mysql-server";
Menu_SQL="MySQL"; fi #set rpm packages name
MACHINE_ARCH=`uname -m`
if [ $MACHINE_ARCH = "x86_64" ]; then ARCH="64"; elif [ $MACHINE_ARCH =
"i686" ]; then MACHINE_ARCH="i386"; else ARCH=""; fi #set arch
Black=`tput setaf 0`      #${Black}
Red=`tput setaf 1`       #${Red}
Green=`tput setaf 2`     #${Green}
Yellow=`tput setaf 3`   #${Yellow}
Blue=`tput setaf 4`     #${Blue}
Magenta=`tput setaf 5`  #${Magenta}
Cyan=`tput setaf 6`     #${Cyan}
White=`tput setaf 7`    #${White}
Bold=`tput bold`        #${Bold}
Rev=`tput smso`         #${Rev}
Reset=`tput sgr0`       #${Reset}

#Initialize variables to default values.
MYSQL_PASSWD="guacamole"
DB_NAME="guacamole"
DB_USER="guacamole"
DB_PASSWD="guacamole"
JKSTORE_PASSWD="guacamole"
INSTALL_MODE="interactive"
INSTALL_NGINX="no"
GUACASERVER_HOSTNAME="localhost"
GUACAMOLE_URI_PATH="guacamole"

HELP () { #Help function
  echo -e "\n"${Bold}Guacamole Install Script Help.${Reset}"\n
  echo "${Bold}Usage:${Reset}"
  echo "  $SCRIPT [options] -s      install Guacamole Silently"
  echo -e "  $SCRIPT [options] -p [yes|no]  install Proxy feature"\n
  echo "${Bold}Options:${Reset}"
  echo "  -${Rev}a${Reset}, <string> --Sets the root password for
${Menu_SQL}. Default is ${Bold}guacamole${Reset}."
  echo "  -${Rev}b${Reset}, <string> --Sets the Guacamole DB name.
Default is ${Bold}guacamole${Reset}."
  echo "  -${Rev}c${Reset}, <string> --Sets the Guacamole DB username.
Default is ${Bold}guacamole${Reset}."
  echo "  -${Rev}d${Reset}, <string> --Sets the Guacamole DB password.
Default is ${Bold}guacamole${Reset}."
```

```

    echo " -${Rev}e${Reset}, <string> --Sets the Java KeyStore password
    (least 6 characters). Default is guacamole."
    echo " -${Rev}s${Reset},          --Install Guacamole Silently. Default
    names and password are: guacamole."
    echo " -${Rev}p${Reset}, [yes|no]  --Install the Proxy feature
    (Nginx)?."
    echo " -${Rev}i${Reset},          --This option launch the interactive
    menu. Default is yes."
    echo " -${Rev}h${Reset},          --Displays this help message and
    exit."
    echo -e " -${Rev}v${Reset},          --Displays the script version
    information and exit.""\n
    echo "Examples:"
    echo " * Full and no interactive install: $SCRIPT -a
    sqlpasswd -b guacadb -c guacadbuser -d guacadbpasswd -e guacakey -s -p
    yes"
    echo " * Same as above but with default names and passwords:
$SCRIPT -s -p yes"
    echo " * Same as above but not install Nginx: $SCRIPT -s -p
    no"
    echo -e " * Only install Nginx: $SCRIPT -p yes""\n
    exit 1
}

showscriptversion () {
echo -e " Guacamole Install Script Version $SCRIPT_VERSION""\n
exit 2
}

while getopts a:b:c:d:e:p:sihv FLAG; do
    case $FLAG in
        a) #set option "a"
            MYSQL_PASSWD=$OPTARG
            ;;
        b) #set option "b"
            DB_NAME=$OPTARG
            ;;
        c) #set option "c"
            DB_USER=$OPTARG
            ;;
        d) #set option "d"
            DB_PASSWD=$OPTARG
            ;;
        e) #set option "e"
            JKSTORE_PASSWD=$OPTARG
            ;;
        p) #set option "p"
            INSTALL_NGINX=$OPTARG
            if [ $INSTALL_MODE != "silent" ]; then INSTALL_MODE="proxy"; fi
            ;;
        s) #set option "s"

```

```
INSTALL_MODE="silent"
;;
i) #set option "i"
if [ $INSTALL_MODE != "silent" ]; then
INSTALL_MODE="interactive"; fi
;;
h) #show help
HELP
;;
v) #set option "v"
showscriptversion
;;
\?) #unrecognized option - show help
echo -e "\n"Option -${BOLD}$OPTARG${NORM} not allowed."
HELP
;;
esac
done

#####
#####      MENU      #####
#####

clear
echo -e "

                                                    ${Yellow}'. .'
                                                    ${Green}'..:///:-.....'          ${Yellow}-yyys/-
                                                    ${Green}../////+++++++/-
${Yellow}.yhyyyhs/'
                                                    ${Green}'..+++++++
${Yellow}'yhyyyhy-
            ${White}.+y' ${Green}'://+++++'
${Yellow}':yhyyo:'
            ${White}-yNd. ${Green}'/+++++'
${Yellow}.+yo:' ${White}'::
            ${White}oNMh' ${Green}./+++++/: ' ' ' '
${White}'mMh.
            ${White}-MMM:  ${Green}/+++++-. ./+:
${White}yMMs
            ${White}-MMMs  ${Green}./+++++'
${White}.mMMy
            ${White}'NMMMy. ${Green}'-/+++++/:.
${White}:dMMMo
            ${White}+MMMMNy:' ${Green}'../////+++++//:-.'
${White}./hMMMMN'
            ${White}-MMMMMMMy+-.' ${Green}' ' ' ' .-:-:-:-
.. ' ' ' ${White}.:ohMMMMMMY
            ${White}sMMMMMMMMMdh+/: ${Green}--.....-
```

```

${White}:/oyhmNNNNNNNNNNd-
  ${White}+.dNNNNNNNNNNNNNNNNmmmmmmNNNNNNNNNNNNNNNNmy: '
    ${White}./sdNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNmho: '
  ${White}'      .:+shmmNNNNNNNNNNNNNNNNNNNNNNmdyo/- '
  ${White}.o:.      '.-:./+osssssso++/:-.'      '-/'
  ${White}.ymh+-..'      '.'/ydy.
  ${White}/dMMNdyo/-.'''      '''.-:+shmNnh:
  ${White}:yNNNNNNNNmdhhyyyyyyyhdmNNNNNNNNy:
  ${White}':sdNNNNNNNNNNNNNNNNNNNNNNds: '
  ${White}'-/+syhdmNNNNNNNNmdhyo/- '
"
menu () {
echo -e "
                                Installation Menu\n
Guacamole Remote Desktop Gateway GUACA_VER\n" && tput sgr0
echo -n "Blue Enter the root password for Menu_SQL: Yellow"
  read MYSQL_PASSWD
  MYSQL_PASSWD=${MYSQL_PASSWD:-guacamole}
echo -n "Blue Enter the Guacamole DB name: Yellow"
  read DB_NAME
  DB_NAME=${DB_NAME:-guacamole}
echo -n "Blue Enter the Guacamole DB username: Yellow"
  read DB_USER
  DB_USER=${DB_USER:-guacamole}
echo -n "Blue Enter the Guacamole DB password: Yellow"
  read DB_PASSWD
  DB_PASSWD=${DB_PASSWD:-guacamole}
echo -n "Blue Enter the Java KeyStore password (least 6 characters):
Yellow"
  read JKSTORE_PASSWD
  JKSTORE_PASSWD=${JKSTORE_PASSWD:-guacamole}
while true; do
  read -p "Blue Do you wish to Install the Proxy feature (Nginx)?:
Yellow" yn
  case $yn in
    [Yy]* ) INSTALL_NGINX="yes"; nginxmenu; break;;
    [Nn]* ) INSTALL_NGINX="no"; break;;
    * ) echo "Blue Please enter yes or no. Yellow";;
  esac
done
tput sgr0
}

nginxmenu ()
{
echo -n "Blue Enter the Guacamole Server IP address or hostname
(default localhost): Yellow"
  read GUACASERVER_HOSTNAME
  GUACASERVER_HOSTNAME=${GUACASERVER_HOSTNAME:-localhost}
echo -n "Blue Enter the URI path (default guacamole): Yellow"
  read GUACAMOLE_URIPATH
  GUACAMOLE_URIPATH=${GUACAMOLE_URIPATH:-guacamole}
}

```

```
}

progressfilt () {
    local flag=false c count cr='\r' nl='\n'
    while IFS='' read -d '' -rn 1 c
    do
        if $flag
        then
            printf '%c' "$c"
        else
            if [[ $c != $cr && $c != $nl ]]
            then
                count=0
            else
                ((count++))
                if ((count > 1))
                then
                    flag=true
                fi
            fi
        fi
    done
}

reposinstall () {
    echo -e "\nChecking CentOS version...\n...CentOS $CENTOS_VER found\n";
    echo -e "\nChecking CentOS version...\n...CentOS $CENTOS_VER found\n"
    >> $logfile 2>&1
    echo -e "\nStarting...\n...Preparing ingredients\n"; echo -e
    "\nStarting...\n...Preparing ingredients\n" >> $logfile 2>&1
    sleep 1 | echo -e "\nSearching for EPEL Repository...";echo -e
    "\nSearching for EPEL Repository..." >> $logfile 2>&1
    rpm -qa | grep epel-release
    RETVAL=$?
    if [ $RETVAL -eq 0 ]; then
        sleep 1 | echo -e "No need to install EPEL repository!"; echo -e
        "No need to install EPEL repository!" >> $logfile 2>&1
    else
        sleep 1 | echo -e "\nIs necessary to install the EPEL
        repositories\nInstalling..."; echo -e "\nIs necessary to install the
        EPEL repositories\nInstalling..." >> $logfile 2>&1
        rpm -Uvh http://dl.fedoraproject.org/pub/epel/epel-release-latest-
        ${CENTOS_VER}.noarch.rpm
    fi

    sleep 1 | echo -e "\nSearching for nux-dextop Repository...";echo -e
```

```
"\nSearching for nut-dextop Repository..." >> $logfile 2>&1
rpm -qa | grep nux-dextop-release
RETVAL=$?
if [ $RETVAL -eq 0 ]; then
    sleep 1 | echo -e "No need to install nut-dextop repository!"; echo
-e "No need to install nux-dextop repository!" >> $logfile 2>&1
else
    sleep 1 | echo -e "\nIs necessary to install the nux-dextop
repositories\nInstalling..."; echo -e "\nIs necessary to install the
nux-dextop repositories\nInstalling..." >> $logfile 2>&1
    rpm -Uvh
http://li.nux.ro/download/nux/dextop/el${CENTOS_VER}/${MACHINE_ARCH}/nu
x-dextop-release-0-2.el${CENTOS_VER}.nux.noarch.rpm
    sed -i "s/enabled=1/enabled=0/g" /etc/yum.repos.d/nux-dextop.repo
fi
}

yumupdate () {
sleep 1 | echo -e "\nUpdating CentOS...\n"; echo -e "\nUpdating
CentOS...\n" >> $logfile 2>&1
yum update -y
}

guacamoleinstall () {
sleep 1 | echo -e "\nInstalling Dependencies..."; echo -e "\nInstalling
Dependencies..." >> $logfile 2>&1

rpm -qa | grep libjpeg-turbo-official-${LIBJPEG_VER}
RETVAL=$? ; echo -e "rpm -qa | grep libjpeg-turbo-official-
${LIBJPEG_VER} RC is: $RETVAL" >> $logfile 2>&1

if [ $RETVAL -eq 0 ]; then
    sleep 1 | echo -e "...libjpeg-turbo-official-${LIBJPEG_VER} is
installed on the system\n"; echo -e "...libjpeg-turbo-official-
${LIBJPEG_VER} is installed on the system\n" >> $logfile 2>&1
else
    sleep 1 | echo -e "...libjpeg-turbo-official-${LIBJPEG_VER} is not
installed on the system\n"; echo -e "...libjpeg-turbo-official-
${LIBJPEG_VER} is not installed on the system\n" >> $logfile 2>&1
    yum localinstall -y
${LIBJPEG_URL}${LIBJPEG_TURBO}.${MACHINE_ARCH}.rpm
    RETVAL=$? ; echo -e "yum localinstall -y
${LIBJPEG_URL}${LIBJPEG_TURBO}.${MACHINE_ARCH}.rpm RC is: $RETVAL" >>
$logfile 2>&1
    ln -vfs /opt/libjpeg-turbo/include/* /usr/include/ || exit 1
    ln -vfs /opt/libjpeg-turbo/lib?/* /usr/lib${ARCH}/ || exit 1
fi

rpm -qa | grep ffmpeg-devel
RETVAL=$? ; echo -e "rpm -qa | grep ffmpeg-devel RC is: $RETVAL" >>
```

```
$logfile 2>&1
if [ $RETVAL -eq 0 ]; then
    sleep 1 | echo -e "...ffmpeg-devel is installed on the system\n";
echo -e "...ffmpeg-devel is installed on the system\n" >> $logfile
2>&1
else
    sleep 1 | echo -e "...ffmpeg-devel is not installed on the
system\n"; echo -e "...ffmpeg-devel is not installed on the system\n"
>> $logfile 2>&1
    yum --enablerepo=nux-dextop install -y ffmpeg-devel
    RETVAL=$? ; echo -e "yum --enablerepo=nux-dextop install -y ffmpeg-
devel RC is: $RETVAL" >> $logfile 2>&1
fi

yum install -y wget pv dialog gcc cairo-devel libpng-devel uuid-devel
ffmpeg-devel freerdp-devel freerdp-plugins pango-devel libssh2-devel
libtelnet-devel libvncserver-devel pulseaudio-libs-devel openssl-devel
libvorbis-devel libwebp-devel tomcat gnu-free-mono-fonts
${MySQL_Packages}
RETVAL=$? ; echo -e "yum install RC is: $RETVAL" >> $logfile 2>&1

sleep 1 | echo -e "\nCreating Directories...\n" | pv -qL 25; echo -e
"\nCreating Directories...\n" >> $logfile 2>&1
rm -fr ${INSTALL_DIR}
mkdir -v /etc/guacamole >> $logfile 2>&1
mkdir -vp ${INSTALL_DIR}{client,selinux} >> $logfile 2>&1 && cd
${INSTALL_DIR}
mkdir -vp ${LIB_DIR}{extensions,lib} >> $logfile 2>&1
mkdir -v /usr/share/tomcat/.guacamole/ >> $logfile 2>&1

sleep 1 | echo -e "\nDownloading Guacamole packages for
installation...\n" | pv -qL 25; echo -e "\nDownloading Guacamole
packages for installation...\n" >> $logfile 2>&1
wget --progress=bar:force ${GUACA_URL}source/${GUACA_SERVER}.tar.gz
2>&1 | progressfilt
#wget --progress=bar:force ${GUACA_URL}source/${GUACA_CLIENT}.tar.gz
2>&1 | progressfilt
wget --progress=bar:force ${GUACA_URL}binary/${GUACA_CLIENT}.war -O
${INSTALL_DIR}client/guacamole.war 2>&1 | progressfilt
wget --progress=bar:force ${GUACA_URL}extensions/${GUACA_JDBC}.tar.gz
2>&1 | progressfilt
wget --progress=bar:force
${MYSQL_CONNECTOR_URL}${MYSQL_CONNECTOR}.tar.gz 2>&1 | progressfilt

sleep 1 | echo -e "\nDerompressing Guacamole Server Source...\n" | pv -
qL 25; echo -e "\nDerompressing Guacamole Server Source...\n" >>
$logfile 2>&1
pv ${GUACA_SERVER}.tar.gz | tar xzf - && rm -f ${GUACA_SERVER}.tar.gz
mv ${GUACA_SERVER} server
```

```
#sleep 1 | echo -e "\nDerompressing Guacamole Client...\n" | pv -qL 25
#pv ${GUACA_CLIENT}.tar.gz | tar xzf - && rm -f ${GUACA_CLIENT}.tar.gz
#mv ${GUACA_CLIENT} client

sleep 1 | echo -e "\nDecrompressing Guacamole JDBC Extension...\n" | pv
-qL 25; echo -e "\nDecrompressing Guacamole JDBC Extension...\n" >>
$logfile 2>&1
pv ${GUACA_JDBC}.tar.gz | tar xzf - && rm -f ${GUACA_JDBC}.tar.gz
mv ${GUACA_JDBC} extension

sleep 1 | echo -e "\nDecompressing MySQL Connector...\n" | pv -qL 25;
echo -e "\nDecompressing MySQL Connector...\n" >> $logfile 2>&1
pv ${MYSQL_CONNECTOR}.tar.gz | tar xzf - && rm -f
${MYSQL_CONNECTOR}.tar.gz

sleep 1 | echo -e "\nCompiling Gucamole Server...\n" | pv -qL 25; echo
-e "\nCompiling Gucamole Server...\n" >> $logfile 2>&1
cd server
./configure --with-init-dir=/etc/init.d
make
sleep 1 && make install
sleep 1 && ldconfig
cd ..

# sleep 1 | echo -e "\nCompiling Gucamole Client...\n" | pv -qL 25
# cd client
# mvn package
# cp guacamole/doc/example/guacamole.properties /etc/guacamole/
# cp guacamole/doc/example/user-mapping.xml /etc/guacamole/

sleep 1 | echo -e "\nCopying Gucamole Client...\n" | pv -qL 25; echo -e
"\nCopying Gucamole Client...\n" >> $logfile 2>&1
cp -v client/guacamole.war ${LIB_DIR}guacamole.war
#cp -v client/guacamole.war /var/lib/tomcat/webapps/guacamole.war

sleep 1 | echo -e "\nMaking Guacamole configurtion files...\n" | pv -qL
25; echo -e "\nMaking Guacamole configurtion files...\n" >> $logfile
2>&1
echo "# Hostname and port of guacamole proxy
guacd-hostname: ${SERVER_HOSTNAME}
guacd-port:      ${GUACA_PORT}

# MySQL properties
mysql-hostname: ${SERVER_HOSTNAME}
mysql-port:     ${MYSQL_PORT}
mysql-database: ${DB_NAME}
mysql-username: ${DB_USER}
mysql-password: ${DB_PASSWD}
mysql-default-max-connections-per-user: 0
mysql-default-max-group-connections-per-user: 0" >
/etc/guacamole/${GUACA_CONF}
```

```
sleep 1 | echo -e "\nMaking Guacamole symbolic links...\n" | pv -qL 25;
echo -e "\nMaking Guacamole symbolic links...\n" >> $logfile 2>&1
ln -vfs ${LIB_DIR}guacamole.war /var/lib/tomcat/webapps || exit 1
ln -vfs /etc/guacamole/${GUACA_CONF} /usr/share/tomcat/.guacamole/ ||
exit 1
ln -vfs ${LIB_DIR}lib/ /usr/share/tomcat/.guacamole/ || exit 1
ln -vfs ${LIB_DIR}extensions/ /usr/share/tomcat/.guacamole/ || exit 1
ln -vfs /usr/local/lib/freerdp/guac* /usr/lib${ARCH}/freerdp || exit 1

sleep 1 | echo -e "\nCopying Guacamole JDBC Extension to Extensions
Dir...\n" | pv -qL 25; echo -e "\nCopying Guacamole JDBC Extension to
Extensions Dir...\n" >> $logfile 2>&1
cp -v extension/mysql/guacamole-auth-jdbc-mysql-${GUACA_VER}.jar
${LIB_DIR}extensions/ || exit 1

sleep 1 | echo -e "\nCopying MySQL Connector to Lib Dir...\n" | pv -qL
25; echo -e "\nCopying MySQL Connector to Lib Dir...\n" >> $logfile
2>&1
cp -v mysql-connector-java-${MYSQL_CONNECTOR_VER}/mysql-connector-java-
${MYSQL_CONNECTOR_VER}-bin.jar ${LIB_DIR}/lib/ || exit 1

if [ $CENTOS_VER -ge 7 ]; then
    sleep 1 | echo -e "\nSetting MariaDB Service...\n" | pv -qL 25;
echo -e "\nSetting MariaDB Service...\n" >> $logfile 2>&1
    systemctl enable mariadb.service
    systemctl restart mariadb.service
    sleep 1 | echo -e "\nSetting Root Password for MariaDB...\n" | pv -
qL 25; echo -e "\nSetting Root Password for MariaDB...\n" >> $logfile
2>&1
else
    sleep 1 | echo -e "\nSetting MySQL Service...\n" | pv -qL 25; echo
-e "\nSetting MySQL Service...\n" >> $logfile 2>&1
    chkconfig mysqld on
    service mysqld start
    sleep 1 | echo -e "\nSetting Root Password for MySQL...\n" | pv -qL
25; echo -e "\nSetting Root Password for MySQL...\n" >> $logfile 2>&1
fi

mysqladmin -u root password ${MYSQL_PASSWD} || exit 1

sleep 1 | echo -e "\nCreating BD & User for Guacamole...\n" | pv -qL
25; echo -e "\nCreating BD & User for Guacamole...\n" >> $logfile 2>&1
mysql -u root -p${MYSQL_PASSWD} -e "CREATE DATABASE ${DB_NAME};" ||
exit 1
mysql -u root -p${MYSQL_PASSWD} -e "GRANT SELECT,INSERT,UPDATE,DELETE
ON ${DB_NAME}.* TO '${DB_USER}'@'localhost' IDENTIFIED BY
'${DB_PASSWD}';" || exit 1
mysql -u root -p${MYSQL_PASSWD} -e "FLUSH PRIVILEGES;" || exit 1
```

```

sleep 1 | echo -e "\nCreating Guacamole Tables...\n" | pv -qL 25; echo
-e "\nCreating Guacamole Tables...\n" >> $logfile 2>&1
cat extension/mysql/schema/*.sql | mysql -u root -p${MYSQL_PASSWD} -D
${DB_NAME}

sleep 1 | echo -e "\nSetting Tomcat Server...\n" | pv -qL 25; echo -e
"\nSetting Tomcat Server...\n" >> $logfile 2>&1
sed -i '72i URIEncoding="UTF-8"' /etc/tomcat/server.xml
sed -i '92i <Connector port="8443" protocol="HTTP/1.1"
SSLEnabled="true" \
        maxThreads="150" scheme="https" secure="true" \
        clientAuth="false" sslProtocol="TLS" \
        keystoreFile="/var/lib/tomcat/webapps/.keystore" \
        keystorePass="JKSTORE_PASSWD" \
        URIEncoding="UTF-8" />' /etc/tomcat/server.xml
sed -i "s/JKSTORE_PASSWD/${JKSTORE_PASSWD}/g" /etc/tomcat/server.xml

if [ $INSTALL_MODE = "silent" ]; then
    sleep 1 | echo -e "\nGenerating the Java KeyStore\n" | pv -qL 25;
echo -e "\nGenerating the Java KeyStore\n" >> $logfile 2>&1
    noprompt="-noprompt -dname CN=,OU=,O=,L=,S=,C="
else
    sleep 1 | echo -e "\nPlease complete the Wizard for the Java
KeyStore\n" | pv -qL 25; echo -e "\nPlease complete the Wizard for the
Java KeyStore\n" >> $logfile 2>&1
fi
keytool -genkey -alias Guacamole -keyalg RSA -keystore
/var/lib/tomcat/webapps/.keystore -storepass ${JKSTORE_PASSWD} -keypass
${JKSTORE_PASSWD} ${noprompt}

sleep 1 | echo -e "\nSetting Tomcat and Guacamole Service...\n" | pv -
qL 25; echo -e "\nSetting Tomcat and Guacamole Service...\n" >>
$logfile 2>&1

if [ $CENTOS_VER -ge 7 ]; then
    systemctl enable tomcat.service >> $logfile 2>&1
    systemctl start tomcat.service >> $logfile 2>&1
    chkconfig guacd on >> $logfile 2>&1
    systemctl start guacd.service >> $logfile 2>&1
else
    chkconfig tomcat on
    service tomcat start >> $logfile 2>&1
    chkconfig guacd on >> $logfile 2>&1
    service guacd start >> $logfile 2>&1
fi
}

selinuxchanges () {
sleep 1 | echo -e "\nInstalling SELinux Tools...\n" | pv -qL 25; echo -
e "\nInstalling SELinux Tools...\n" >> $logfile 2>&1
yum install -y policycoreutils-{python,devel}

```

```
sleep 1 | echo -e "\nSetting SELinux...\n" | pv -qL 25; echo -e
"\nSetting SELinux...\n" >> $logfile 2>&1

if [ -d ${INSTALL_DIR}selinux ]; then
    sleep 1 | echo -e "\nThe directory ${INSTALL_DIR}selinux already
exist...\n" | pv -qL 25; echo -e "\nThe directory ${INSTALL_DIR}selinux
already exist...\n" >> $logfile 2>&1
else
    sleep 1 | echo -e "\nCreating the directory...
${INSTALL_DIR}selinux\n" | pv -qL 25; echo -e "\nCreating the
directory... ${INSTALL_DIR}selinux ...\n" >> $logfile 2>&1
    mkdir -vp ${INSTALL_DIR}selinux >> $logfile 2>&1 && cd
${INSTALL_DIR}selinux
fi

echo '
module nginx 1.0;

require {
    type httpd_t;
    type http_cache_port_t;
    class tcp_socket name_connect;
}

#===== httpd_t =====

#!!!! This avc can be allowed using one of the these booleans:
# httpd_can_network_relay, httpd_can_network_connect
allow httpd_t http_cache_port_t:tcp_socket name_connect;' > nginx.te
checkmodule -M -m -o nginx.mod nginx.te >> $logfile 2>&1
semodule_package -o nginx.pp -m nginx.mod >> $logfile 2>&1
semodule -i nginx.pp >> $logfile 2>&1
}

nginxinstall ()
{
sleep 1 | echo -e "\nInstalling Nginx repository..."; echo -e
"\nInstalling Nginx repository..." >> $logfile 2>&1
echo '[nginx]
name=nginx repo
baseurl=http://nginx.org/packages/centos/$releasever/$basearch/
gpgcheck=0
enabled=1' > /etc/yum.repos.d/nginx.repo

sleep 1 | echo -e "\nInstalling Nginx..."; echo -e "\nInstalling
Nginx..." >> $logfile 2>&1
yum install -y nginx pv
RETVAL=$? ; echo -e "yum install RC is: $RETVAL" >> $logfile 2>&1

sleep 1 | echo -e "\nMaking Nginx Backup\nmv
```

```
/etc/nginx/conf.d/default.conf /etc/nginx/conf.d/default.conf.ori.bkp"
| pv -qL 25; echo -e "\nMaking Nginx Backup\nmv
/etc/nginx/conf.d/default.conf /etc/nginx/conf.d/default.conf.ori.bkp"
>> $logfile 2>&1
mv /etc/nginx/conf.d/default.conf
/etc/nginx/conf.d/default.conf.ori.bkp
sleep 1 | echo -e "\nMaking Nginx configurations..." | pv -qL 25; echo
-e "\nMaking Nginx configurations..." >> $logfile 2>&1
echo 'server {
    listen 80;
    server_name localhost;

    location /_new-path_/ {
        proxy_pass http://_SERVER_HOSTNAME_:8080/guacamole/;
        proxy_buffering off;
        proxy_http_version 1.1;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection $http_connection;
        proxy_cookie_path /guacamole/ /_new-path_/;
        access_log off;
    }
}' > /etc/nginx/conf.d/guacamole.conf
sed -i "s/_SERVER_HOSTNAME_/${GUACASERVER_HOSTNAME}/g"
/etc/nginx/conf.d/guacamole.conf
sed -i "s/_new-path_/${GUACAMOLE_URIPATH}/g"
/etc/nginx/conf.d/guacamole.conf

echo 'server {
    listen          443 ssl http2;
    server_name     localhost;
    ssl_certificate  guacamole.crt;
    ssl_certificate_key guacamole.key;
    ssl_protocols   TLSv1 TLSv1.1 TLSv1.2;
    ssl_ciphers     HIGH:!aNULL:!MD5;

    location /_new-path_/ {
        proxy_pass http://_SERVER_HOSTNAME_:8080/guacamole/;
        proxy_buffering off;
        proxy_http_version 1.1;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection $http_connection;
        proxy_cookie_path /guacamole/ /_new-path_/;
        access_log off;
    }
}' > /etc/nginx/conf.d/guacamole_ssl.conf
sed -i "s/_SERVER_HOSTNAME_/${GUACASERVER_HOSTNAME}/g"
/etc/nginx/conf.d/guacamole_ssl.conf
sed -i "s/_new-path_/${GUACAMOLE_URIPATH}/g"
/etc/nginx/conf.d/guacamole_ssl.conf
```

```
if [ $INSTALL_MODE = "silent" ]; then
    sleep 1 | echo -e "\nGenerating a Self-Signed SSL Certificate...\n"
    | pv -qL 25; echo -e "\nGenerating a Self-Signed SSL Certificate...\n"
    >> $logfile 2>&1
    subj="-subj /C=XX/ST=/L=City/O=Company/CN="
else
    sleep 1 | echo -e "\nPlease complete the Wizard for the Self-Signed
SSL Certificate...\n" | pv -qL 25; echo -e "\nPlease complete the
Wizard for the Self-Signed SSL Certificate...\n" >> $logfile 2>&1
fi
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout
/etc/nginx/guacamole.key -out /etc/nginx/guacamole.crt ${subj}

sleep 1 | echo -e "\nStarting Nginx Service...\n" | pv -qL 25; echo -e
"\nStarting Nginx Service...\n" >> $logfile 2>&1
if [ $CENTOS_VER -ge 7 ]; then
    systemctl enable nginx.service || exit 1
    systemctl start nginx.service || exit 1
else
    chkconfig nginx on
    service nginx start
fi

sleep 1 | echo -e "${Bold}\nIf you need understand the Nginx
configurations please go to:\n ${Green} http://nginx.org/en/docs/
${Reset} ${Bold} \nIf you need replace the certificate file please read
first:\n ${Green}
http://nginx.org/en/docs/http/configuring_https_servers.html ${Reset}
\n" | pv -qL 25; echo -e "${Bold}\nIf you need understand the Nginx
configurations please go to:\n ${Green} http://nginx.org/en/docs/
${Reset} ${Bold} \nIf you need replace the certificate file please read
first:\n ${Green}
http://nginx.org/en/docs/http/configuring_https_servers.html ${Reset}
\n" >> $logfile 2>&1

selinuxchanges
}

firewalld () {
    echo -e "\nMaking Firewall Backup...\n"
    cp /etc/firewalld/zones/public.xml $fwbkpfile >> $logfile 2>&1
    cp /etc/firewalld/zones/public.xml $fwbkpfile
    if [ $INSTALL_NGINX = "yes" ]; then
        sleep 1 | echo -e "...Opening ports 80 and 443\n" | pv -qL 25;
        echo -e "...Opening ports 80 and 443\n" >> $logfile 2>&1
        echo -e "Add new rule...\nfirewall-cmd --permanent --
zone=public --add-service=http" >> $logfile 2>&1
        firewall-cmd --permanent --zone=public --add-service=http >>
```

```

$logfile 2>&1
    echo -e "Add new rule...\nfirewall-cmd --permanent --
zone=public --add-service=https" >> $logfile 2>&1
    firewall-cmd --permanent --zone=public --add-service=https >>
$logfile 2>&1
    fi
    if [ $INSTALL_MODE = "interactive" ] || [ $INSTALL_MODE = "silent"
]; then
        sleep 1 | echo -e "...Opening ports 8080 and 8443\n" | pv -qL 25;
echo -e "...Opening ports 8080 and 8443\n" >> $logfile 2>&1
        echo -e "Add new rule...\nfirewall-cmd --permanent --
zone=public --add-port=8080/tcp" >> $logfile 2>&1
        firewall-cmd --permanent --zone=public --add-port=8080/tcp >>
$logfile 2>&1
        echo -e "Add new rule...\nfirewall-cmd --permanent --
zone=public --add-port=8443/tcp" >> $logfile 2>&1
        firewall-cmd --permanent --zone=public --add-port=8443/tcp >>
$logfile 2>&1
        echo -e "Reload firewall...\nfirewall-cmd --reload\n" >>
$logfile 2>&1
    fi
    firewall-cmd --reload >> $logfile 2>&1
}

Iptables () {
    echo -e "Making Firewall Backup...\niptables-save >> $fwbkpfile" >>
$logfile 2>&1
    iptables-save >> $fwbkpfile 2>&1
    if [ $INSTALL_NGINX = "yes" ]; then
        sleep 1 | echo -e "...Opening ports 80 and 443\n" | pv -qL 25;
echo -e "...Opening ports 80 and 443\n" >> $logfile 2>&1
        echo -e "Add new rule...\niptables -I INPUT -m tcp -p tcp --
dport 80 -m state --state NEW -j ACCEPT" >> $logfile 2>&1
        iptables -I INPUT -m tcp -p tcp --dport 80 -m state --state NEW
-j ACCEPT >> $logfile 2>&1
        echo -e "Add new rule...\niptables -I INPUT -m tcp -p tcp --
dport 443 -m state --state NEW -j ACCEPT" >> $logfile 2>&1
        iptables -I INPUT -m tcp -p tcp --dport 443 -m state --state
NEW -j ACCEPT >> $logfile 2>&1
    fi
    if [ $INSTALL_MODE = "interactive" ] || [ $INSTALL_MODE = "silent"
]; then
        sleep 1 | echo -e "...Opening ports 8080 and 8443\n" | pv -qL
25; echo -e "...Opening ports 8080 and 8443\n" >> $logfile 2>&1
        echo -e "Add new rule...\niptables -I INPUT -m tcp -p tcp --
dport 8080 -m state --state NEW -j ACCEPT" >> $logfile 2>&1
        iptables -I INPUT -m tcp -p tcp --dport 8080 -m state --state
NEW -j ACCEPT >> $logfile 2>&1
        echo -e "Add new rule...\niptables -I INPUT -m tcp -p tcp --
dport 8443 -m state --state NEW -j ACCEPT" >> $logfile 2>&1
        iptables -I INPUT -m tcp -p tcp --dport 8443 -m state --state

```

```
NEW -j ACCEPT >> $logfile 2>&1
fi
echo -e "Save new rules\nservice iptables save\n" >> $logfile 2>&1
service iptables save >> $logfile 2>&1
}

firewallsetting () {
sleep 1 | echo -e "\nSetting Firewall...\n" | pv -qL 25; echo -e
"\nSetting Firewall..." >> $logfile 2>&1
echo -e "Take Firewall RC...\n" >> $logfile 2>&1
echo -e "rpm -qa | grep firewalld" >> $logfile 2>&1
rpm -qa | grep firewalld >> $logfile 2>&1
RETVALqaf=$?
echo -e "\nservice firewalld status" >> $logfile 2>&1
service firewalld status >> $logfile 2>&1
RETVALsf=$?

if [ $RETVALsf -eq 0 ]; then
    sleep 1 | echo -e "...firewalld is installed and started on the
system\n" | pv -qL 25; echo -e "...firewalld is installed and started
on the system\n" >> $logfile 2>&1
    firewalld
elif [ $RETVALqaf -eq 0 ]; then
    sleep 1 | echo -e "...firewalld is installed but not enabled or
started on the system\n" | pv -qL 25; echo -e "...firewalld is
installed but not enabled or started on the system\n" >> $logfile 2>&1
    firewalld
else
    sleep 1 | echo -e "...firewalld is not installed on the system\n" |
pv -qL 25; echo -e "...firewalld is not installed on the system\n" >>
$logfile 2>&1
    echo -e "Checking Firewall RC..." >> $logfile 2>&1
    rpm -qa | grep iptables-services >> $logfile 2>&1
    RETVALqai=$?
    service iptables status >> $logfile 2>&1
    RETVALsi=$?

    if [ $RETVALsi -eq 0 ]; then
        sleep 1 | echo -e "...iptables service is installed and started
on the system\n" | pv -qL 25; echo -e "...iptables service is installed
and started on the system\n" >> $logfile 2>&1
        Iptables
    elif [ $RETVALqaf -eq 0 ]; then
        sleep 1 | echo -e "...iptables is installed but not enabled or
started on the system\n" | pv -qL 25; echo -e "...iptables is installed
but not enabled or started on the system\n" >> $logfile 2>&1
        Iptables
    else
        sleep 1 | echo -e "...iptables service is not installed on
the system\n" | pv -qL 25; echo -e "...iptables service is not
```

```
installed on the system\n" >> $logfile 2>&1
    sleep 1 | echo -e "Please check and configure you
firewall...\nIn order to Guacamole work properly open the ports tcp
8080 and 8443." | pv -qL 25; echo -e "Please check and configure you
firewall...\nIn order to Guacamole work properly open the ports tcp 80,
443, 8080 and 8443." >> $logfile 2>&1
    fi
fi
}

showmessages () {
sleep 1 | echo -e "\nFinished Successfully\n" | pv -qL 25; echo -e
"\nFinished Successfully\n" >> $logfile 2>&1
sleep 1 | echo -e "\nYou can check the log file ${logfile}\n" | pv -qL
25; echo -e "\nYou can check the log file ${logfile}\n" >> $logfile
2>&1
sleep 1 | echo -e "\nYour firewall backup file ${fwbkpfiler}\n" | pv -qL
25; echo -e "\nYour firewall backup file ${fwbkpfiler}\n" >> $logfile
2>&1
if [ $INSTALL_NGINX = "yes" ]; then
    sleep 1 | echo -e "\nTo manage the Guacamole GW via proxy go to
http://<IP>/${GUACAMOLE_URIPATH}/ or
https://<IP>/${GUACAMOLE_URIPATH}/\n" | pv -qL 25; echo -e "\nTo manage
the Guacamole GW via proxy go to http://<IP>/${GUACAMOLE_URIPATH}/ or
https://<IP>/${GUACAMOLE_URIPATH}/\n" >> $logfile 2>&1
    fi
if [ $INSTALL_MODE = "interactive" ] || [ $INSTALL_MODE = "silent" ];
then
    sleep 1 | echo -e "\nTo manage the Guacamole GW go to
http://<IP>:8080/${GUACAMOLE_URIPATH}/ or
https://<IP>:8443/${GUACAMOLE_URIPATH}/\n" | pv -qL 25; echo -e "\nTo
manage the Guacamole GW go to http://<IP>:8080/${GUACAMOLE_URIPATH}/ or
https://<IP>:8443/${GUACAMOLE_URIPATH}/\n" >> $logfile 2>&1
    sleep 1 | echo -e "\nThe username and password is: guacadmin\n" | pv -
qL 25; echo -e "\nThe username and password is: guacadmin\n" >>
$logfile 2>&1
    fi
sleep 1 | echo -e "\nIf you have any suggestions please write to:
correo@nacimientohernan.com.ar\n" | pv -qL 25; echo -e "\nIf you have
any suggestions please write to: correo@nacimientohernan.com.ar\n" >>
$logfile 2>&1
}

# Start
if [[ $INSTALL_MODE = "interactive" && $INSTALL_MODE != "silent" &&
$INSTALL_MODE != "proxy" ]]; then menu; fi
if [ $INSTALL_MODE = "interactive" ] || [ $INSTALL_MODE = "silent" ] ||
[ $INSTALL_NGINX = "yes" ]; then reposinstall; fi
if [ $INSTALL_MODE = "interactive" ] || [ $INSTALL_MODE = "silent" ];
```

```
then yumupdate; fi
if [ $INSTALL_MODE = "interactive" ] || [ $INSTALL_MODE = "silent" ];
then guacamoleinstall; fi
if [ $INSTALL_NGINX = "yes" ]; then nginxinstall; fi
if [ $INSTALL_MODE = "interactive" ] || [ $INSTALL_MODE = "silent" ] ||
[ $INSTALL_NGINX = "yes" ]; then firewallsetting; fi
if [ $INSTALL_MODE = "interactive" ] || [ $INSTALL_MODE = "silent" ] ||
[ $INSTALL_NGINX = "yes" ]; then showmessages; fi
```

---

**Quelle:** <https://sourceforge.net/projects/guacamoleinstallscript/>

Last update: **2017/09/04 13:44**