

# TeamTILT for Nagios Installation guide Client Nagios 1.1

*(Version 1.0, last change 08.09.2010)*

## Table of Contents

1.Introduction.....	3
2.Installation.....	4
3.Installation details.....	7
COMMUNICATION WITH THE TEAMTILT SERVER.....	7
TEST OF COMMUNICATION SERVICES.....	7
NAGIOS PLUGIN CONFIGURATION.....	11
Step 1.....	11
Step 2.....	11
Step 3.....	11
Step 4.....	11
step 5.....	12

(Version 1.0, last change 08.09.2010)

## 1. Introduction

---

This document guides you through the installation of the TeamTILT client for Nagios. In a few minutes you will be able to relay your alerts to all your contacts around the world.

The TeamTILT client for Nagios is a light Java application that enables the relaying of your Nagios notifications to the TeamTILT platform. The alert can then be sent to multiple media types, SMS, voice call, email or iPhone push notification, without any additional hardware!

*(Version 1.0, last change 08.09.2010)*

## 2. Installation

A detailed description about the different actions performed during the installation process can be found in chapter 4. Please note that this is provided solely as information and is not required for the installation process itself and can be skipped.

- To start, login as root to install the client :

**NOTE :** *This installation guide is based on a Linux distribution that does not use the sudo command. Should your distribution require it, place it right before the described commands hereunder.*

```
su -
```

- Download « TeamTILT for Nagios » console version, 32 or 64 bits depending your system. We suggest to place yourself under /usr/local/

```
cd /usr/local
```

```
wget http://www.alarmtilt.com/clients/TeamTILTClientForNagios_CL_JRE_32.tar.gz
```

```
cd /usr/local
```

```
wget http://www.alarmtilt.com/clients/TeamTILTClientForNagios_CL_JRE_64.tar.gz
```

- Unpack the archive containing the the installer folder :

```
tar xzvf TeamTILTClientForNagios_CL_JRE_32.tar.gz
```

```
tar xzvf TeamTILTClientForNagios_CL_JRE_64.tar.gz
```

- A directory named « TeamTILTForNagios » will be created and the owner of this directory set to Nagios. The client will later be run as Nagios user. If you have no Nagios user defined, the client will be run as root user.
- Change the current directory to the newly created one and start the « runTeamTILT.sh » script :

```
cd TeamTILTForNagios/
```

```
./runTeamTILT.sh
```

- The licensing information « license.txt » can be found in this directory.
- A communication test between the client and the TeamTILT platform is performed.

**NOTE :** *The port 80 must be opened to the internet !*

- In this step, a TeamTILT account will be created (free of any charge). This account is needed for the client authentication and enables you to manage your contacts and alerts from a web interface accessible under [www.teamtilt.net](http://www.teamtilt.net) Simply complete the form, including your email address, first and last name and your company name . The email address and your password are your credentials to login to the web interface.

**NOTE :** *If you already have a TeamTILT account, enter your existing login and your password to reuse it.*

(Version 1.0, last change 08.09.2010)

```
*****
USER REGISTRATION
*****
```

```
Please enter a valid email address      : nagios@teamtilt.net
Insert your password (min 6 car.)       : yourpassword
Confirm password                        : yourpassword
Insert first name                       : TeamTILT
Insert last name                        : User
Insert company name                     : M-PLIFY S.A
```

- The client will check if you have the latest version. Shouldn't this be the case, an update will be available. Simply confirm and download/install the latest version.
- After the account creation, the client is started and a cron task is added to restart the client if a problem occurs.

```
crontab -e -u nagios
# ----- TEAMTILT CLIENT START -----
# Added by nagios at Thu Jul 29 14:53:23 CEST 2010
*/5 * * * *
/usr/local/TeamTILTForNagios/ClientManager/sh_scripts/startTTService.sh
/usr/local/TeamTILTForNagios >>
/usr/local/TeamTILTForNagios/ClientCommunication/logs/TeamTILT_ClientCommunication.log
# ----- TEAMTILT CLIENT END -----
```

- Indicate the correct Nagios installation paths (default paths are provided) A file named « TeamTILT.conf » is created with two new notification commands, a new contact group « TeamTILTClient.group » and a new contact « TeamTILTClient.contact ». Attach this contact group to the monitored hosts or services whose notifications you want to relay to TeamTILT.
- The last step is to choose the hosts or service whose notifications should be relayed to TeamTILT . Choose « y », if you want to include all your detected services and hosts. Choose « n », if you want to manually select them by adding the « TeamTILTClient.group » to the hosts or services

**NOTE :** *This configuration can be done at any other moment after the installation procedure.*

- The client installation is now finished. Go to <http://www.teamtilt.net> and login using the email address (login) and the password provided during earlier.
- In the web interface, a contact group « TeamTILTClient » has been created including the contact defined during the installation procedure. This 'TeamTILTClient' group is the group that will be alerted if a notification comes in from Nagios. You can setup new contacts and add them to this group

(Version 1.0, last change 08.09.2010)

**Note:** *Do not delete this group in your web interface! Should this happen, please recreate a group named 'TeamTILTClient' (case sensitive).*

- Search your contact and configure a telephone number for voice calls and the SMS messages.
- Deactivate a service on your monitored server and test if you will receive an alert on your phone. Under the web interface, click on the « action history » button to see a list of all generated alerts.

*(Version 1.0, last change 08.09.2010)*

### 3. Installation details

---

#### COMMUNICATION WITH THE TEAMTILT SERVER

StepCheckWebServiceConnectivity

Testing the existence of JAVA\_HOME : dossier jre1.6.0\_06\_linux / jre1.6.0\_21

Testing the existence of java file under bin/

Launch : /home/notroot/Download/TeamTILTForNagios/java/jre1.6.0\_06\_linux/bin/java com.mplify.alarmtilt.client.plugin.setup.biz.HttpProxyCheck (

CLASSPATH=/home/notroot/Download/TeamTILTForNagios/ClientManager/dist:

/home/notroot/Download/TeamTILTForNagios/ClientManager/dist/binding.jar:

/home/notroot/Download/TeamTILTForNagios/ClientManager/dist/atgvt-wscli-acc.jar:

/home/notroot/Download/TeamTILTForNagios/ClientManager/dist/atgvt-wscli-dir.jar:

/home/notroot/Download/TeamTILTForNagios/ClientManager/dist/atcli.jar:

/home/notroot/Download/TeamTILTForNagios/ClientManager/dist/atgvt-wscli-proc.jar:

/home/notroot/Download/TeamTILTForNagios/ClientManager/dist/synthetica.jar:

/home/notroot/Download/TeamTILTForNagios/ClientManager/dist/  
syntheticaOrangeMetallic.jar:

/home/notroot/Download/TeamTILTForNagios/ClientManager/dist/atgvt-wscli-subscr.jar:

/home/notroot/Download/TeamTILTForNagios/ClientManager/dist/forms.jar:

/home/notroot/Download/TeamTILTForNagios/ClientManager/dist/atgvt-ws-common.jar:

/home/notroot/Download/TeamTILTForNagios/ClientManager/dist/log4j.jar:

/home/notroot/Download/TeamTILTForNagios/ClientManager/dist/atgvt-wscli.jar

#### TEST OF COMMUNICATION SERVICES

StepServiceInstall

removeService

uninstallATService.sh

parameter : TEAMTILT\_HOME

#create temp file

CRON\_TEMP\_FILE=`mktemp \$TEAMTILT\_HOME/at\_XXXXXXXXX`

# remove lines from crontab

crontab -l | sed '/--- TEAMTILT CLIENT START/,/--- TEAMTILT CLIENT END/d' >  
\$CRON\_TEMP\_FILE

*(Version 1.0, last change 08.09.2010)*

```

cat $CRON_TEMP_FILE | crontab -
# remove temp file
/bin/rm $CRON_TEMP_FILE
stopService
stopATService.sh
parameter : TEAMTILT_HOME
IS_SERVICE_STARTED = $TEAMTILT_HOME/ClientManager/sh_scripts/isServiceStarted.sh
source $IS_SERVICE_STARTED
checkIfServiceIsStarted $MAIN_CLASS
OCCURENCES=$?
if [[ $OCCURENCES == 0 ]]; then
    echo "In stopATService.sh, there is no instance running($OCCURENCES), so there is
instance to be stopped"
else
    echo "In stopATService.sh, there are $OCCURENCES instances running which needs to be
stopped"
    mkdir -m+rwx $TEAMTILT_HOME/ClientCommunication/run
    cd $TEAMTILT_HOME/ClientCommunication/run
    touch $STOP_FILE
    chmod 777 $STOP_FILE
    echo "Successfully created stop file"
fi
installService
installATService.sh
parameter : TEAMTILT_HOME
startATService.sh $TEAMTILT_HOME >> $LOG_FILE
# add line in crontab file which will make TEAMTILTService function as a daemon. It will try
to execute startATService.sh once at 5 mins
COMMAND = "*/5 * * * * startATService.sh $TEAMTILT_HOME >> $LOG_FILE"
CRON_TEMP_FILE=`mktemp $TEAMTILT_HOME/at_XXXXXXXXX`
crontab -l > $CRON_TEMP_FILE
echo "# ----- AT CLIENT START -----" >> $CRON_TEMP_FILE

```

*(Version 1.0, last change 08.09.2010)*

```

echo "# Added by "`whoami`" at `date`" >> $CRON_TEMP_FILE
echo "$COMMAND" >> $CRON_TEMP_FILE
echo "# ----- AT CLIENT END -----" >> $CRON_TEMP_FILE
cat $CRON_TEMP_FILE | crontab -
# remove temp file
/bin/rm $CRON_TEMP_FILE
isServiceInstalled
isServiceInstalled.sh
checkIfServiceIsInstalled()
    CRON_TEMP_FILE=`mktemp $TEAMTILT_HOME/at_XXXXXXXX`
    crontab -l > $CRON_TEMP_FILE
    # find out if there is any record of the ATService
    OCCURENCES=`cat $CRON_TEMP_FILE | fgrep -c "# ----- TEAMTILT CLIENT START
-----"
    # remove temp file
    /bin/rm $CRON_TEMP_FILE
return $OCCURENCES
startService
startATService.sh
parameter : TEAMTILT_HOME
CLIENT_COMMUNICATION_HOME = $TEAMTILT_HOME/ClientCommunication
MPLIFY_JARS_DIR = $CLIENT_COMMUNICATION_HOME/mplify-jars
THIRD_PARTY_JARS_DIR = $CLIENT_COMMUNICATION_HOME/third-party-jars
# define the class that is going to be executed and is the entry point to the
ClientCommunication module
MAIN_CLASS = com.mplify.longhaul.client.Main
# create the list of jars that will be used when executing the main class
CLASSPATH=.
LIBS=`find $AT_HOME/ClientCommunication -name *.jar`
for LIB in $LIBS
do
    CLASSPATH=$CLASSPATH:$LIB

```

*(Version 1.0, last change 08.09.2010)*

```

done
# define the java execution additional parameters
ADDITIONAL_JAVA_PARAMS = "-Dinstance=longhaulclient"
ADDITIONAL_JAVA_PARAMS = "$ADDITIONAL_JAVA_PARAMS -Dsun.net.inetaddr.ttl=10"
ADDITIONAL_JAVA_PARAMS = "$ADDITIONAL_JAVA_PARAMS -ea"
ADDITIONAL_JAVA_PARAMS = "$ADDITIONAL_JAVA_PARAMS -Xss200k"
ADDITIONAL_JAVA_PARAMS = "$ADDITIONAL_JAVA_PARAMS -Xmx100m"
# define the log file
LOG_FILE=$CLIENT_COMMUNICATION_HOME/logs/TeamTILT_ClientCommunication.log
# define the run-time parameters for the main class
PARAMS="PROPERTIES"
PARAMS="$PARAMS $CLIENT_COMMUNICATION_HOME/config/longhaulclient.properties"
PARAMS="$PARAMS WATCHDOG 22234 STOPFILE
$CLIENT_COMMUNICATION_HOME/run/stop.server"
# launch the java class
JAVA_HOME=$AT_HOME/java/jre1.6.0_06_linux/bin/java
$JAVA_HOME $ADDITIONAL_JAVA_PARAMS -cp $CLASSPATH $MAIN_CLASS $PARAMS >>
$LOG_FILE

isServiceStarted
IsServiceStarted.sh
checkIfServiceIsStarted()
    MAIN_CLASS = $1
    # create temp file for the result of ps faux command
    FILE_NAME=`mktemp at_XXXXXXXX`
    ps faux > $FILE_NAME
    # find out if there is an instance of the Main class already running
    OCCURENCES=`cat $FILE_NAME | fgrep -c $MAIN_CLASS`
    # remove temp file
    /bin/rm $FILE_NAME
    return $OCCURENCES
CheckIfNagiosUserExists
checkNagiosUserProps.sh

```

*(Version 1.0, last change 08.09.2010)*

parameter : TEAMTILT\_HOME

This script checks if nagios user is defined in /etc/passwd. If it is, then check if it has rights on the \$INSTALLATION\_FOLDER

## NAGIOS PLUGIN CONFIGURATION

### Step 1

stepGetNagiosInstallationPath

Checking for the launch script 'nagios'

getAllNagiosInstallationsPaths.sh /etc/rc.d/init.d/

getAllNagiosInstallationsPaths.sh /etc/init.d/

Saving the Nagios file path to

ClientManager/data/teamtiltNagios.properties

### Step 2

stepGetNagiosCheckConfigFilesPath

Checking for the 'nagios' binary file

getNagiosCheckConfigFilePaths.sh TEAMTILT\_HOME /usr/local/

getNagiosCheckConfigFilePaths.sh TEAMTILT\_HOME /opt/

/usr/local/ and /opt/

Saving the path to ClientManager/data/teamtiltNagios.properties : nagios LSB executable file path settings

### Step 3

stepGetNagiosCfgFilesPath

Checking for 'cgi.cfg' and inside the string : 'main\_config\_file='

(usually: /usr/local/nagios/etc/nagios.cfg)

/usr/local/nagios/etc, /etc/nagios and /opt/nagios/etc

### Step 4

stepNagiosChooseOutputFilesPath

Ask to the user the localisation of the eventfiles directory

If the user choses the default path :

mkdir /usr/local/nagios/etc/eventfiles/

--> Restart Client Communication module : stopService / removeService / installService / startService

(Version 1.0, last change 08.09.2010)

If the user doesn't chose the default path :

Mkdir choosen directory

Restart Client Communication module : stopService / removeService / installService / startService

step 5

stepNagiosChooseHostAndServices

Reading teamTILTClient.properties file:

teamtiltclient.contactgroup.name

teamtiltclient.contact.name

teamtiltclient.service.command.name

teamtiltclient.host.command.name

Reading nagios.cfg file :

cfgFilesList[] : file = /usr/local/nagios/etc/objects/commands.cfg

cfgFilesList[] : file = /usr/local/nagios/etc/objects/contacts.cfg

cfgFilesList[] : file = /usr/local/nagios/etc/objects/timeperiods.cfg

cfgFilesList[] : file = /usr/local/nagios/etc/objects/templates.cfg

cfgFilesList[] : file = /usr/local/nagios/etc/objects/localhost.cfg

Searching for hosts and services in this file

Displaying these hosts and services to the user

Writing TeamTILT.cfg file : adding of contact data, contact group data, command data.

Add TeamTILT.cfg to nagios.cfg

(Version 1.0, last change 08.09.2010)