



install-vts-helpers.sh – Bash Script

```
#!/bin/bash -e

# Script used to create a yum repository config entry for Cisco VTS

CONTINUE_SCRIPT=true
INTERACTIVE_MODE=false
REPO_DIR="/etc/yum.repos.d/"
RepoUrl="https://devhub.cisco.com/artifactory/vts-yum/2.6.0.vts260-os-newton/"
RepoUrlDef=$RepoUrl
RepoUserName=""
RepoPassword=""

packages="cisco-vts-overcloud-installer cisco-vts-tripleo-heat-templates-extra "

####
# Help
####

function show_help {

echo "Options are:"
echo " --RepoUser=Username for the repo hosting the Cisco VTS RPMS. OPTIONAL"
echo " --RepoPass=Password for the repo hosting the Cisco VTS RPMS. OPTIONAL"
echo " --RepoUrl=Base URL for the repo hosting the Cisco VTS RPMS. OPTIONAL. Defaults
to ${RepoUrl}"
echo " --Proxy=Host name/IP and port for the HTTP/HTTPS proxy. Eg proxy-server:3128.
OPTIONAL."
echo " --ProxyCreds=Proxy Username:Password. Eg user:password OPTIONAL."
echo " -h or --help: Show this message"
}

####
# Repo create
####

function create_repo_file {

rm -f ${REPO_FILE}

cat <<EOT >> ${REPO_FILE}
[${repo}]
name=${repo}
baseurl=${RepoUrl}
username=${RepoUserName}
password=${RepoPassword}
enabled = 0
gpgcheck = 0
metadata_expire = 86400
EOT

}
```

```

function export_proxy {
if [[ -n "$Proxy" ]]; then
    echo "Exporting Proxy settings into environment variables"

    local_ips=`ifconfig | grep -Eo 'inet
(addr:)?([0-9]*\.){3}[0-9]*' | grep -Eo '([0-9]*\.){3}[0-9]*' | grep -v '127.0.0.1'`
    no_proxy="localhost,127.0.0.1"

    for local_ip in ${local_ips[@]}; do
        no_proxy="${no_proxy},${local_ip}"
    done
    echo "no_proxy settings for local IPs: ${no_proxy}"

    if [[ -n "$ProxyCreds" ]]; then
        _proxy="http://${ProxyCreds}@${Proxy}"
    else
        _proxy="http://${Proxy}"
    fi
    echo "http and https proxy settings: ${_proxy}"

    export no_proxy="${no_proxy}"
    export {http,https,HTTP,HTTPS}_proxy="${_proxy}"

fi
}

####
# Install the vts install tools
####

function install_cisco_vts_install_helpers {
    echo "Installing via yum: ${packages}"
    yum install -y ${packages} --enablerepo ${repo}
}

####
# Main
####

if [ $# -lt 1 ]; then
    INTERACTIVE_MODE=true
    echo "This script can be used in non-interactive mode (install-vts-helpers --help to
display options)"
    read -p "To continue in Interactive mode press return; To quit press 'q': " Dummy
    if [ "$Dummy" = "q" ]; then
        exit 0
    fi

    #show_help
    read -p "VTS Repo User [$RepoUserName]: " RepoUserName
    echo -n "VTS Repo Password [$RepoPassword]: "
    read -s RepoPassword
    echo
    read -p "VTS Repo URL [$RepoUrlDef]: " RepoUrl
    RepoUrl="${RepoUrl:-$RepoUrlDef}"
    read -p "Proxy IP:Port[optional]: " Proxy
    read -p "Proxy username[optional]: " ProxyUser
    echo -n "Proxy password[optional]: "
    read -s ProxyPass
    echo
    ProxyCreds=${ProxyUser}:"${ProxyPass}"

    printf "\nChecking for mandatory input parameters ..... \n"
    if [ -z "$RepoUserName" ]; then
        echo "VTS Repo User is mandatory"
        CONTINUE_SCRIPT=false
    fi
    if [ -z "$RepoPassword" ]; then

```

```

        echo "VTS Repo password is mandatory"
        CONTINUE_SCRIPT=false
    fi
fi

if [ "$CONTINUE_SCRIPT" = false ]; then
    echo "Exiting script"
    exit 1
fi

# This block is relevant for non-interactive mode only
if [ "$INTERACTIVE_MODE" = false ]; then
    for i in "$@"
    do
        case $i in
            --RepoUrl=*)
                RepoUrl="${i#*=}"
                ;;
            --RepoUser=*)
                RepoUserName="${i#*=}"
                ;;
            --RepoPass=*)
                RepoPassword="${i#*=}"
                ;;
            --Proxy=*)
                Proxy="${i#*=}"
                ;;
            --ProxyCreds=*)
                ProxyCreds="${i#*=}"
                ;;
            -h|--help)
                show_help
                CONTINUE_SCRIPT=false
                exit
                ;;
            *)
                echo "unknown option"
                show_help # unknown option
                CONTINUE_SCRIPT=false
                exit
                ;;
        esac
    done
fi

repo=`echo ${RepoUrl%/*} | rev | cut -d/ -f1 | rev`
REPO_FILE="${REPO_DIR}${repo}.repo"

echo "Creating ${REPO_FILE}"
create_repo_file
export_proxy
install_cisco_vts_install_helpers
echo "Done"

```

