

Installation des TSM Clients on Debian Linux 64-bit OS

RPM basierte Systeme (Suse, Fedora, Red-Hat usw.)

Status: v.1.0.0

Software Installation on Linux



TECHNISCHE
UNIVERSITÄT
DARMSTADT

HRZ
Hochschulrechenzentrum

Inhaltsverzeichnis

Inhaltsverzeichnis	i
1.....DEB-basierte Systeme (Debian, Ubuntu usw.)	3
2.....Installation des CommandLine Agent (dsmc)	4
2.1. Linux System Aktualisierung	4
2.2. Installation der notwendigen Packages	4
2.3. Download TSM Packages	4
2.4. Konvertierung RPM-Pakete für Debian	4
2.5. Vorbereitung für Debian build-script	4
2.6. Anpassen der Berechtigungen von Dateien für die Skripte	5
2.7. Bearbeitungen der Steuerdatei „control“ für jedes einzelne Paket	5
2.8. Generieren der Debian Pakete	6
2.9. Installation der deb-Pakete (Rheinenfolge beachten)	6
2.10. Anpassung der Pfade für shared libraries	7
2.11. Testing	7
2.12. Weitere Konfigurationen	7
2.12.1. dsm.sys	8

2.12.2. dsm.opt	8
2.12.3. incl-excl	8
2.12.4. Scheduler Konfiguration und automatische Sicherung Konfiguration	9
3.....Installation des TSM GUI (dsmj)	11

1. DEB-basierte Systeme (Debian, Ubuntu usw.)

Debian-artige Linux Systeme werden NICHT offiziell von IBM unterstützt, der TSM Client funktioniert aber in der Regel ohne große Schwierigkeiten.

Die Funktionstüchtigkeit des TSM Clients unter Debian System wurde von der TSM-Gruppe getestet, aber die Nutzung erfolgt dennoch auf eigene Gefahr.

Um die von IBM zur Verfügung gestellten RPM-Pakete installieren zu können, gibt es zwei verschiedene Wege:

- I. Konvertierung mit Hilfe von Programm „**alien**“ in Debian-Pakete und anschließende Installation
- II. Extrahieren der Dateien per Hand und Einspielen der Dateien in die entsprechenden Pfade

Um die manuelle Arbeit möglicherweise zu reduzieren, wird hier nur die erste Option (Konvertierung) diskutiert.

Es gibt ein Skript tsm2deb (<https://github.com/briner/tsm2deb/tree/7.1.0.1>), welches debian Paket erzeugt. Die Nutzung erfolgt auf eigene Gefahr!

2. Installation des CommandLine Agent (dsmc)

Allgemeine: Für Installation des TSM Clients on Linux System ist „root“-Berechtigung notwendig, so stellen Sie zuerst sicher, dass Sie als „root“ eingeloggt sind.

2.1. Linux System Aktualisierung

Falls Ihre Applikationen nicht aktuelle Versionen unterstützt, bitte ignorieren Sie diesen Schritt.

```
aptitude update
aptitude full-upgrade
```

2.2. Installation der notwendigen Packages

64-bit System (ia32-lib ermöglicht 32-bit Applikation auf 64-bit Maschine zu laufen)

```
aptitude install alien libstdc++6 ksh ia32-libs
```

2.3. Download TSM Packages

Die TSM Clients Software können aus einer von folgenden Seiten heruntergeladen werden (wählen Sie bitte die aktuelle Version, z.B. zur Zeit v7r1).

- HRZ, TU Darmstadt
http://www.software.tu-darmstadt.de/tsm_download/v7r1/
- IBM maintenance Version oder Patches
<ftp://public.dhe.ibm.com/storage/tivoli-storage-management/patches/client/v7r1/>
<ftp://public.dhe.ibm.com/storage/tivoli-storage-management/maintenance/client/v7r1/>

Wählen Sie ein Verzeichnis z.B. /home/install_tsm auf Ihr Linux System, wo es mindestens 1024MB frei Kapazität gibt, starten Sie Ihr Download, z.B. die Installationsdatei: *VersionsNummer-TIV-TSMBAC-LinuxX86.tar*, wobei die Versionsnummer 7.1.1.0 ist.

```
cd /home/install_tsm
wget http://www.software.tu-darmstadt.de/tsm_download/v711/Linux_x86_64/7.1.1.0-TIV-
TSMBAC-LinuxX86.tar
tar xvf 7.1.1.0-TIV-TSMBAC-LinuxX86.tar
```

2.4. Konvertierung RPM-Pakete für Debian

Für jeder Konvertierung mit „alien“ sind jeweils zwei Unterverzeichnisse erstellt, wie z.B. `alien -g -c TIVsm-API64.x86_64.rpm` erstellt **TIVsm-API64-7.1.1** and `TIVsm-API64-7.1.1.orig`, wobei das Verzeichnis ohne „.orig“ sollen wir weiter bearbeiten.

```
alien -g -c gskcrypt64-8.0.14.43.linux.x86_64.rpm
alien -g -c gskssl64-8.0.14.43.linux.x86_64.rpm
alien -g -c TIVsm-API64.x86_64.rpm
alien -g -c TIVsm-APIcit.x86_64.rpm
alien -g -c TIVsm-BA.x86_64.rpm
alien -g -c TIVsm-BAcit.x86_64.rpm
```

2.5. Vorbereitung für Debian build-script

Umbenennen Unterverzeichnisname von „debian“ nach „DEBIAN“.

```
mv ./gskssl64-8.0/debian ./gskssl64-8.0/DEBIAN
mv ./gskcrypt64-8.0/debian ./gskcrypt64-8.0/DEBIAN
```

```
mv ./TIVsm-API64-7.1.1/debian ./TIVsm-API64-7.1.1/DEBIAN
mv ./TIVsm-APIcit-7.1.1/debian ./TIVsm-APIcit-7.1.1/DEBIAN
mv ./TIVsm-BA-7.1.1/debian ./TIVsm-BA-7.1.1/DEBIAN
mv ./TIVsm-BAcit-7.1.1/debian ./TIVsm-BAcit-7.1.1/DEBIAN
```

2.6. Anpassen der Berechtigungen von Dateien für die Skripte

```
chmod 755 ./gskssl64-8.0/DEBIAN/{postinst,preinst}
chmod 755 ./TIVsm-API64-7.1.1/DEBIAN/{postinst,preinst}
chmod 755 ./TIVsm-APIcit-7.1.1/DEBIAN/{postinst,preinst}
chmod 755 ./TIVsm-BA-7.1.1/DEBIAN/{postinst,preinst}
chmod 755 ./TIVsm-BAcit-7.1.1/DEBIAN/{postinst,preinst}
chmod 755 ./gskcrypt64-8.0/DEBIAN/{postinst,preinst}
```

2.7. Bearbeitungen der Steuerdatei „control“ für jedes einzelne Paket

Hier werden Ihre Name und Email-Adresse in dem Feld „Maintainer“ eingetragen. Der Architecture-Feld können Sie von „uname -a“ ermitteln, für 64-bit Maschine ist „amd64“ oder „x86_64“.

Die Version-Feld kann von dem entsprechenden rpm-Name ermittelt werden.

```
cat > ./gskcrypt64-8.0/DEBIAN/control <<EOF
Package: gsk-crypt
Version: 8.0
Architecture: amd64
Maintainer: firstname lastname <your_email_add@xxx.tu-darmstadt.de>
Section: main
Priority: extra
Description: gskcrypt64 for TSM @ TUDarmstadt
  Crypto Library for TSM
  TSM stands for Tivoli Storage Manager and it is a solution for backuping is
  machine at TUDarmstadt.
EOF
```

```
cat > ./gskssl64-8.0/DEBIAN/control <<EOF
Package: gsk-ssl
Version: 8.0
Architecture: amd64
Maintainer: firstname lastname <your_email_add@xxx.tu-darmstadt.de>
Section: main
Priority: extra
Description: gskssl64 for TSM @ TUDarmstadt
  SSL Library for TSM
  TSM stands for Tivoli Storage Manager and it is a solution for backuping is
  machine at TUDarmstadt.
EOF
```

```
cat > ./TIVsm-API64-7.1.1/DEBIAN/control <<EOF
Package: tivsm-api
Version: 7.1.1.0
Architecture: amd64
Maintainer: firstname lastname <your_email_add@xxx.tu-darmstadt.de>
Section: main
Priority: extra
Description: api for TSM @ TUDarmstadt
  API Library for TSM
  TSM stands for Tivoli Storage Manager and it is a solution for backuping is
  machine at TUDarmstadt.
EOF
```

```
cat > ./TIVsm-APIcit-7.1.1/DEBIAN/control <<EOF
Package: tivsm-apicit
Version: 7.1.1.0
Architecture: amd64
```

```
Maintainer: firstname lastname <your_email_add@xxx.tu-darmstadt.de>
Section: main
Priority: extra
Description: api for TSM @ TUDarmstadt
API Library for TSM
TSM stands for Tivoli Storage Manager and it is a solution for backuping is
machine at TUDarmstadt.
```

EOF

```
cat >./TIVsm-BA-7.1.1/DEBIAN/control <<EOF
```

```
Package: tivsm-ba
Version: 7.1.1.0
Architecture: amd64
Maintainer: firstname lastname <your_email_add@xxx.tu-darmstadt.de>
Section: main
Priority: extra
Description: api for TSM @ TUDarmstadt
API Library for TSM
TSM stands for Tivoli Storage Manager and it is a solution for backuping is
machine at TUDarmstadt.
```

EOF

```
cat >./TIVsm-BAcit-7.1.1/DEBIAN/control <<EOF
```

```
Source: tivsm-bacit
Version: 7.1.1.0
Section: alien
Priority: extra
Maintainer: firstname lastname <your_email_add@xxx.tu-darmstadt.de>
Package: tivsm-bacit
Architecture: amd64
Depends: ${shlibs:Depends}
Description: CIT binaries for BA
          IBM Tivoli Storage Manager CIT binaries for the BA client
.
(Converted from a rpm package by alien version 8.87.)
```

EOF

2.8. Generieren der Debian Pakete

```
dpkg -b TIVsm-API64-7.1.1
dpkg -b TIVsm-API64cit-7.1.1
dpkg -b TIVsm-BA-7.1.1
dpkg -b TIVsm-BAcit-7.1.1
dpkg -b gskssl64-8.0
dpkg -b gskcrypt64-8.0
```

2.9. Installation der deb-Pakete (Reihenfolge beachten)

Bevor die generierte deb-Packages auf Ihrem System installiert werden, bitte prüfen ob alte Packages schon im System existieren, es ist besser, wenn die alte Packages zuerst deinstalliert werden (dpkg -r <package Name>).

Prüfen installiert Packages:

```
dpkg -l | grep tivsm
dpkg -l | grep gsk
```

Remove installierter Packages:

```
dpkg -r tivsm-api64
dpkg -r tivsm-ba
```

Installieren deb-Packages:

```
dpkg -i gskcrypt64-8.0.deb
dpkg -i gskssl64-8.0.deb
dpkg -i TIVsm-API64-7.1.1.deb
dpkg -i TIVsm-APIcit-7.1.1.deb
dpkg -i TIVsm-BA-7.1.1.deb
dpkg -i TIVsm-BAcit-7.1.1.deb
```

2.10. Anpassung der Pfade für shared libraries

Anpassung der Pfade für shared libraries, indem Sie folgende Konfigurationsdatei editieren und die Inhalte ändern:

```
vi /etc/ld.so.conf.d/tsm.conf
```

fügen Sie die Zeilen

```
/opt/tivoli/tsm/client/api/bin64
/opt/tivoli/tsm/client/ba/bin
```

ein und aktualisieren Sie die Konfiguration mit folgendem Befehl:

```
ldconfig -v
```

2.11. Testing

Bevor Sie das Programm „dsmc“ starten, können Sie die Abhängigkeit der shared libraries prüfen:

```
cd /opt/tivoli/tsm/client/ba/bin
ldd dsmc
```

es werden folgende Informationen gezeigt: (wenn alle Pfade richtig konfiguriert sind, dann wird keine Fehlermeldungen wie „file not found“ gezeigt)

```
linux-vdso.so.1 => (0x00007fffab292000)
libgsk8ssl_64.so => /opt/tivoli/tsm/gsk8_64/lib64/libgsk8ssl_64.so (0x00007f239d3d9000)
libgsk8iccs_64.so => /opt/tivoli/tsm/gsk8_64/lib64/libgsk8iccs_64.so (0x00007f239d27f000)
libxmlutil-7.1.0.0.so => /opt/tivoli/tsm/client/api/bin64/libxmlutil-7.1.0.0.so (0x00007f239d166000)
libcrypt.so.1 => /lib/x86_64-linux-gnu/libcrypt.so.1 (0x00007f239cf2f000)
libpthread.so.0 => /lib/x86_64-linux-gnu/libpthread.so.0 (0x00007f239cd13000)
libdl.so.2 => /lib/x86_64-linux-gnu/libdl.so.2 (0x00007f239cb0e000)
libstdc++.so.6 => /usr/lib/x86_64-linux-gnu/libstdc++.so.6 (0x00007f239c807000)
libgpfs.so => /opt/tivoli/tsm/client/api/bin64/libgpfs.so (0x00007f239c6fa000)
libdmapi.so => /opt/tivoli/tsm/client/api/bin64/libdmapi.so (0x00007f239c5f2000)
librt.so.1 => /lib/x86_64-linux-gnu/librt.so.1 (0x00007f239c3ea000)
libm.so.6 => /lib/x86_64-linux-gnu/libm.so.6 (0x00007f239c168000)
libgcc_s.so.1 => /lib/x86_64-linux-gnu/libgcc_s.so.1 (0x00007f239bf51000)
libc.so.6 => /lib/x86_64-linux-gnu/libc.so.6 (0x00007f239bbc6000)
libgsk8cms_64.so => /opt/tivoli/tsm/gsk8_64/lib64/libgsk8cms_64.so (0x00007f239b7a9000)
libtsmmerces-depdom.so.28 => /opt/tivoli/tsm/client/api/bin64/libtsmmerces-depdom.so.28 (0x00007f239b635000)
libtsmmerces-c.so.28 => /opt/tivoli/tsm/client/api/bin64/libtsmmerces-c.so.28 (0x00007f239b11a000)
/lib64/ld-linux-x86-64.so.2 (0x00007f239d63d000)
libgsk8sys_64.so => /opt/tivoli/tsm/gsk8_64/lib64/libgsk8sys_64.so (0x00007f239b017000)
```

2.12. Weitere Konfigurationen

TSM Client ist unter /opt/tivoli/tsm/client installiert, die Konfigurationsdatei dsm.sys und dsm.opt müssen in der Verzeichnis /opt/tivoli/tsm/client/ba/bin angelegt werden. Falls es um eine Upgrade des TSM Clients geht, können Sie Ihre vorherige Konfigurationsdatei weiter verwenden.

Für eine neue Installation können Sie folgenden Vorlagen benutzen.

Unter „Nodename“ tragen Sie bitte vom HRZ mitgeteilten Nodennamen ein.

2.12.1.dsm.sys

```
*****
* IBM Tivoli Storage Manager                                     *
*                                                                 *
* Sample Client System Options file for UNIX (dsm.sys)          *
*****

* This file contains the minimum options required to get started
* using TSM*
* If your client node communicates with multiple TSM servers, be
* sure to add a stanza, beginning with the SERVERNAME option, for
* each additional server.

*****

SErvername          tsm1
COMMMethod         TCPip
TCPPort            1500
TCPServeraddress   130.83.126.101
NODENAME           (your Nodename)
InclExcl           /opt/tivoli/tsm/client/ba/bin/incl-excl
domain             /local-mount-point-for-a-filesystem
domain             /path1
virtualmountpoint  /path1
schedlogname       /var/log/tsm-(your hostname).log
schedlogret        3
errorlogname       /var/log/tsm-(your hostname).err
errorlogret        3
compression        off
DIRMC              standard
Passwordaccess     generate
```

Unter "domain" tragen Sie die Verzeichnisnamen ein, für die Sie eine Sicherung durchführen möchten. Wenn der Verzeichnisname nicht gleich der mount-point eines Filesystem ist, muss dieses Verzeichnis auch als „virtualmountpoint“ deklariert werden.

2.12.2.dsm.opt

```
*****
* IBM Tivoli Storage Manager                                     *
*                                                                 *
* Sample Client System Options file for UNIX (dsm.opt)          *
*****

SErvername          tsm1
Subdir              yes
```

2.12.3.incl-excl

```
EXCLUDE /.../*
INCLUDE /home/temp/.../* STANDARD
INCLUDE /etc/.../* STANDARD
```

Diese Datei beschreibt welche Dateien eines File Systems gesichert werden oder nicht, gesteuert über eine sogenannten „include-exclude-list“. In dieser Liste sind folgende Parameter und „Wildcard“ verwendet:

Parameter	Beschreibung
INCLUDE	Erwartet eine Datei als Argument, Datei(en) werden gesichert, „wildcards“ kann in Dateinamen verwendet werden.

EXCLUDE		Erwartet eine Datei als Argument, Datei(en) werden <i>nicht</i> gesichert, „wildcards“ kann in Dateinamen verwendet werden.
EXCLUDE.DIR		Erwartet Verzeichnisse oder Unterverzeichnisse als Argument, Verzeichnisse werden nicht gesichert. „wildcards“ kann in Verzeichnisnamen verwendet werden.
Wildcard	?	beliebiges Zeichen außer Verzeichnistrenner e.g. „/“ in Linux
	*	beliebige Zeichenkette außer Verzeichnistrenner
	...	0, 1 oder beliebig viele Verzeichnisse

Mit Hilfe von Parameter INCLUDE, EXCLUDE und EXCLUDE.DIR kann eine Liste von „include-exclude-list“ in der Konfigurationsdatei (dsm.sys oder dsm.opt) angegeben werden.

ACHTUNG: die „include-exclude-list“ wird entgegen üblicher Praxis von unten nach oben ausgewertet!

Nachdem Sie die beiden Konfigurationsdateien (include incl-excl) in dem richtigen Verzeichnis gespeichert haben, müssen Sie einmal am TSM Server anmelden, damit der TSM Client das Passwort speichern kann. Das Passwort des TSM Clients ist normalerweise nachdem das Client bei dem TSM Server erfolgreich angemeldet unter Linux in /etc/adsm/TSM.PWD verschlüsselt gespeichert, man kann das Passwort nicht mehr in klaren Text wiederstellen. Wenn diese Datei verloren geht, würde der automatische Start des TSM Schedulers oder die automatische Sicherung nicht funktionieren.

Starten Sie folgendes Command unter Linux Prompt:

```
dsmc
```

Hier werden Sie nach Ihrer User-ID (e.g. Ihrem Nodename) und dann nach dem Passwort für diesen Node gefragt(Nodename und Passwort finden Sie in Ihren Anmeldeformular). Wenn Login korrekt ist, meldet TSM Client an dem Server mit Prompt:

```
tsm>
```

Sie können mit folgenden Befehl die Verbindungsdaten zu ermitteln und den TSM Client zu verlassen.

```
tsm> q session
```

```
tsm> quit
```

2.12.4.Scheduler Konfiguration und automatische Sicherung Konfiguration

Um die Sicherung von Client oder Scheduler für Client automatisch zu starten, gibt es in Linux drei Wege:

- Start-Stop-script in /etc/init.d
- /etc/inittab
- crontab

Wir empfehlen Sie, Ihre Systemsicherung (increment backup) über crontab zu starten.

Die Vorteile sind: Sie wissen besser wann Ihre Sicherung gestartet werden soll und die Sicherung (increment backup) wird sofort ohne Wartezeit gestartet.

Erstellen Sie im Verzeichnis `/etc/init.d` mit einem Editor die Datei `tsm_client.sh` mit folgendem Inhalt:

```
#!/bin/sh
unset LC_ALL
export LANG=en_US.ISO-8859-1
export LC_CTYPE=en_US.ISO-8859-1
dsmc inc >/var/log/tsm_client_inc.log 2>&1
```

Setzen Sie das x-Bit zum Ausführen der Datei

```
chmod 755 /etc/init.d/tsm_client.sh
```

Tragen Sie diese Script in Ihrem crontab mit Startzeit z.B. jeden Tag um 1:00

```
00 01 * * * /etc/init.d/tsm_client.sh
```

3. Installation des TSM GUI (dsmj)

Um die grafische Benutzeroberfläche des TSM Clients zu benutzen, müssen Sie folgende Pakete installieren:

- JAVA Runtime Environment (JRE) in Linux Seite (Der alt-least Version von JRE ist 1.5):

```
aptitude install sun-java6-bin
```

Es werden zwei Packages installiert, e.g. sun-java6-bin und sun-java6-jre.

Das Paket java6-runtime in einem 64-bit Debian Linux-System ist virtuell Paket und in dem Paket „openjdk-6-jre“ inklusive, in diesem Fall installieren Sie java6 wie folgendes:

```
aptitude install openjdk-6-jre
```

Es werden zwei Packages installiert, e.g. openjdk-6-jre und openjdk-6-jre-headless.

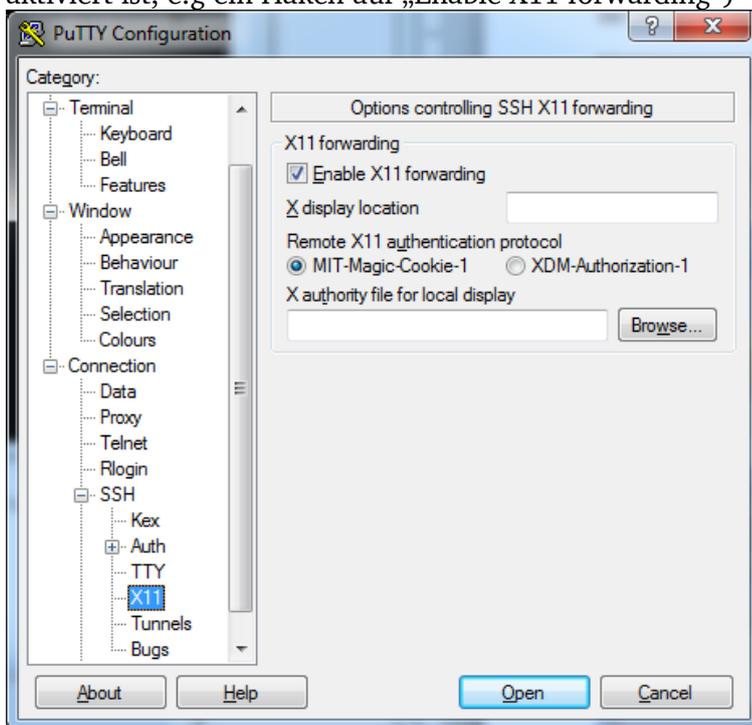
- X-Server auf Windows Workstation

Auf der PC Workstation Seite brauchen Sie ein X-Server, z.B. Xming.

Xming kann mit ssh-Implementierung wie Putty genutzt werden, so dass X11-Sessions verschlüsselt von remote Linux-Server laufen können.

Installieren Sie Xming auf Ihrem PC Workstation und starten Sie Xming.

Versuchen Sie auf Linux Server per Putty einzuloggen (bitte beachten Sie, dass X11-Forwarding aktiviert ist, e.g ein Haken auf „Enable X11 forwarding“)



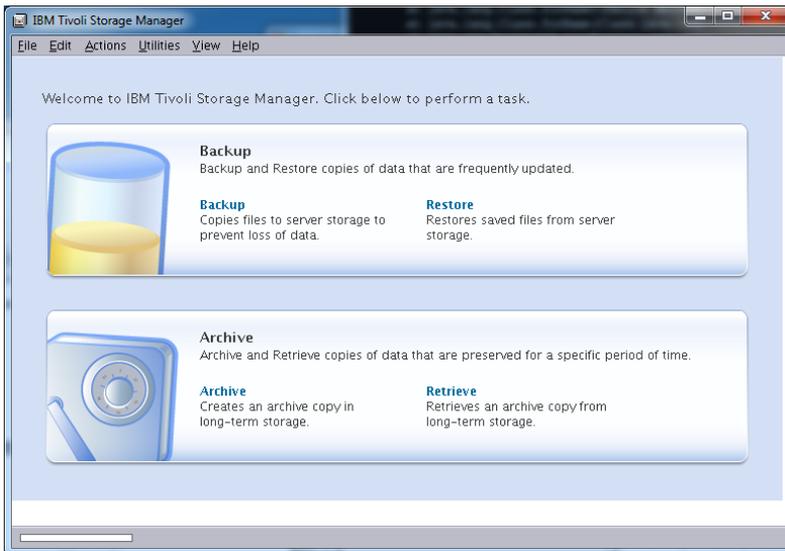
Auf Ihrer Serverseite prüfen Sie ob UNIX Environment DISPLAY automatisch gesetzt ist, sonst liegt das Problem bei der Putty-Konfiguration für die aktuelle Session.

```
echo $DISPLAY  
localhost:10.0
```

dort können Sie das Command „dsmj“ aufrufen:

```
dsmj &
```

Sie bekomme ein Windows auf Ihrem PC-Workstation wie folgendes:

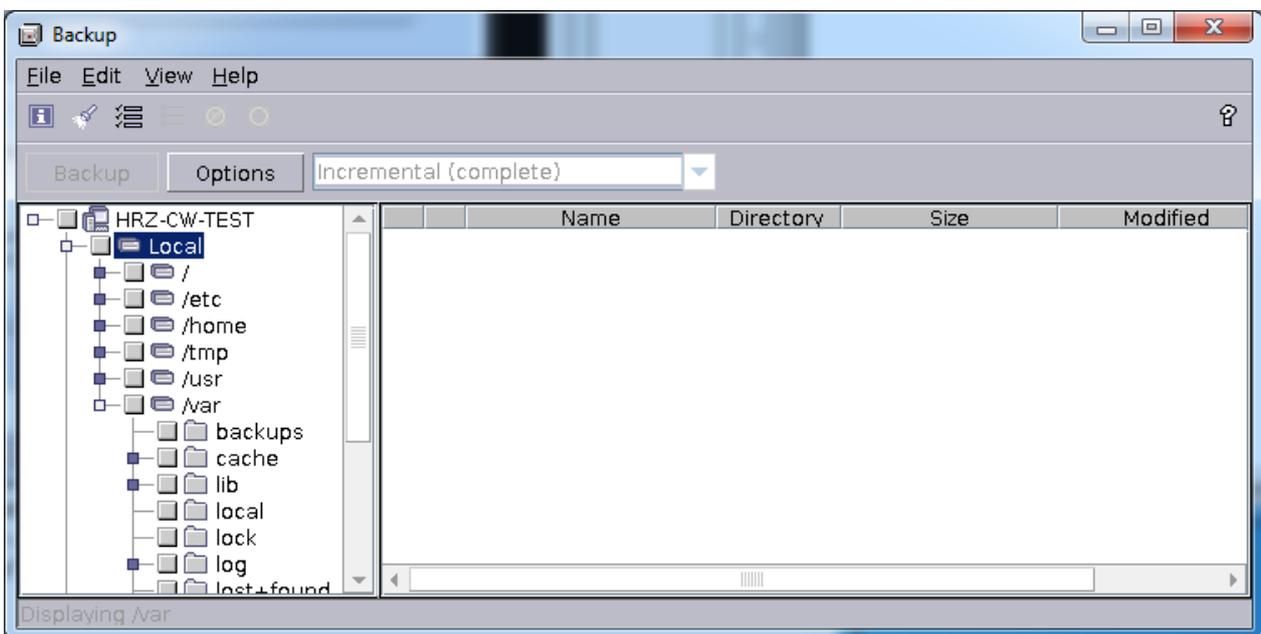


Hier können Sie die Funktionalität von Backup und Archive benutzen, indem Sie auf dem entsprechenden Funktion klicken.

- Backup (entsprechende Funktionen sind „backup“ und „restore“)
- Archive (entsprechende Funktionen sind „archive“ und „retrieve“)

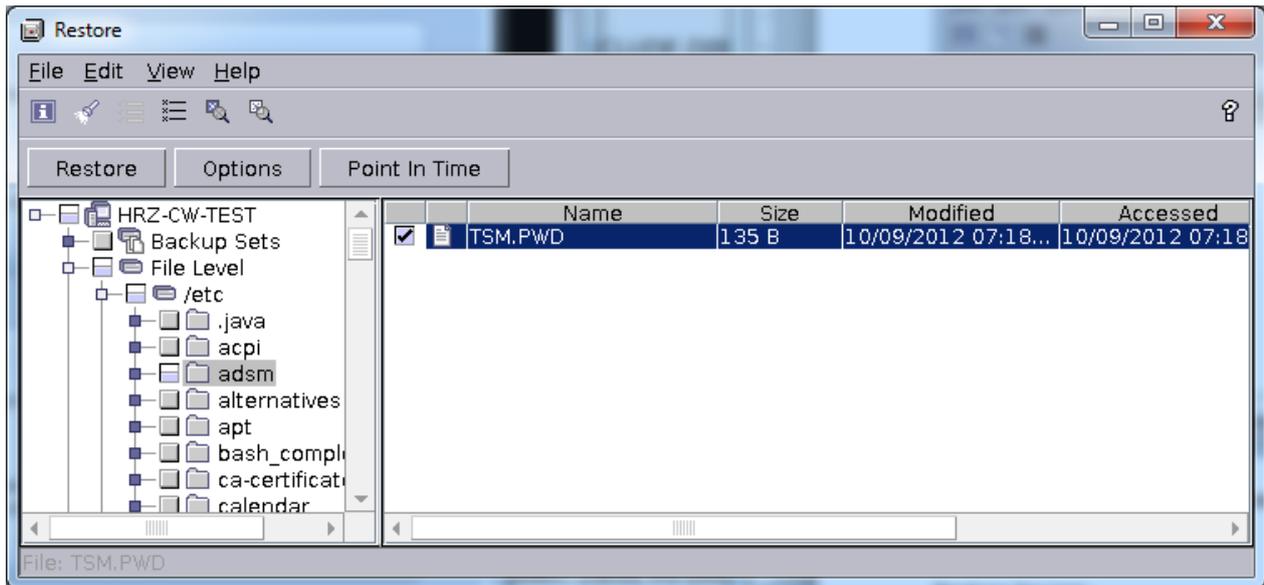
The TSM GUI kann nur eine Funktion gleichzeitig starten.

Backup Session:



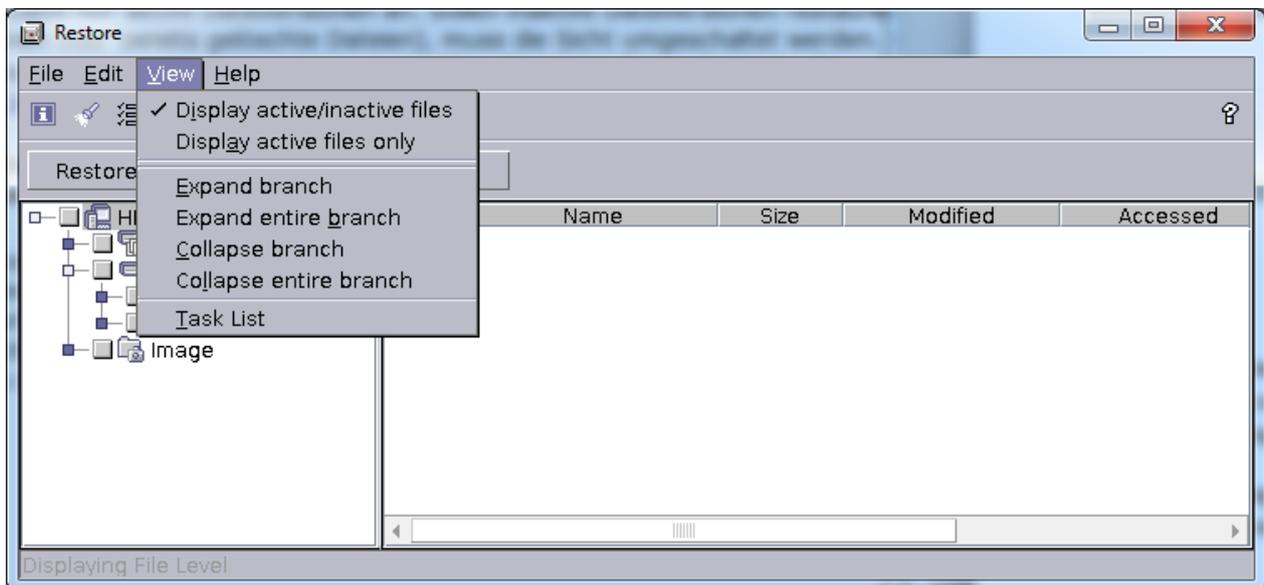
Hier können Sie gewünschte Dateien oder Verzeichnisse markieren indem Sie auf dem entsprechenden Kästchen klicken.

Restore Session:



Hier können Sie beliebige Dateien oder Verzeichnisse markieren, die schon in dem TSM Server gesichert sind, aus dem Backup zurückholen wenn Sie auf dem „Restore“ Kästchen klicken. Dieser Beispiel zeigt Ihnen die Passwort-Datei von TSM-Client zurückzuholen.

Im Allgemeine zeigt der TSm Client nur ative Dateiversion an. Sollten inaktive Dateiversionen aus Backup zurückgeholt werden, muss die Ansicht umgeschlatet werden, indem im Menü „view“ die Zeile „display active/inactive files“ auswählen.



Für weiterführende Information für TSM Backup/Datensicherung besuchen Sie unsere Web-Seite: <http://www.hrz.tu-darmstadt.de/serverbetreuung/tsm/datensicherungmittsm.de.jsp>