
TISBackup Documentation

Release 1.8.2

Tranquil IT

Nov 23, 2020

PRESENTING TISBACKUP

1	Overview of existing solutions	3
1.1	Bacula	3
1.2	r-snapshot	3
2	Indices and tables	25



The objective of TISbackup is to benefit from file backups and centralized alert feedback on “reasonable” data volumes (of the order of a few TB).

TISBackup allows:

- to know if a recent backup exists;
- to keep a history with deduplication at the file level (no duplicate backups);
- to have an immediate view of the contents of a server or a server area for data restoration ;
- to export the last backup to an external media in order to transfer it to a secure location;
- to configure the backup cycle with a simple and readable `.ini` file;
- to work with a module mechanism to extend the type of backups (https, rsync, postgres, mysql,) of virtual machines;

Satisfying these needs stems from the need for a tool to manage a vast pool of machines each hosting a multitude of different software or services (different editors, different hardware platforms and operating environments, etc.). Finally, as the backup procedures of a publisher changed without any warning, the remote backup mechanisms were regularly broken, which caused us some scares with the mechanisms we were using before.

OVERVIEW OF EXISTING SOLUTIONS

Different open source solutions exist but did not meet our specifications.

1.1 Baccula

Bacula is a high-performance solution for full backups on tape and removable media. However, a restore can take a long time and the storage of a history can be voluminous. The backup is saved on a file system that is not readable by a Windows system. An uninitiated “backup manager” will not be able to check the contents of his backup from home.

1.2 r-snapshot

r-snapshot almost corresponds to the specifications but is complex to configure and any necessary modification would have been difficult to develop as an overlay of the existing one:

- the backups are organized by date then by zone which is the opposite of what was desired;
- it is not possible to configure different backup frequencies according to the criticality levels of the servers;
- finally, the deletion of obsolete backups is done in the same process as the backups, which can be very long and can be problematic if there is a problem during the backup.

... and now TISbackup ...

1.2.1 Technical background for TISBackup

The deduplication of this solution is based on the hardlinks of ext3/4 file systems used for storing backup files.

The backup server must run **rsync** in server mode, and the workstations to be backed up must be equipped with **rsync** and **ssh** (usually basic on machines running GNU/Linux, with **cygwin** (or another tool like **cwrsync**) for machines running MS Windows).

tisbackup

tisbackup is a python script that the backup server runs at regular intervals. The configuration file `tisbackup.ini` contains the details of the tasks to be executed.

tisbackup has different options for its execution, available in the **tisbackup --help** command, the main ones being the following:

- **backup**: executes all scheduled backups;
- **cleanup**: examines the backups and deletes those that are older than the defined maximum retention time ;
- **checknagios**: returns the content that can be viewed by nagios ;
- **retryfailed**: redoes the backups that previously failed;
- **exportbackup**: exports the last valid backups to the specified location (remote, external media, ...);
- **register_existing**: scans the backups that have been made and adds the missing ones to the database;

tisbackup.ini

`tisbackup.ini` defines the backups to be executed and supervised. It is written with a simple formalism.

The different types of backups are:

- **rsync**: the backup of a directory by rsync using the rsync protocol;
- **rsync+ssh**: the backup of a directory by rsync with the ssh protocol;
- **mysql+ssh**: saving a mysql database in a gzipped sql file, with the ssh protocol;
- **pgsql+ssh**: the backup of a postgresql database in a gzipped sql file, with the ssh protocol;
- **xen-xva**: the backup of a virtual machine running on an XCP server as an XVA file;
- **xen-meta-data**: the backup of XCP metadata from a virtualization server;
- **switch**: the backup of switches;
- **null**: null backup of a server that does not require a backup but for which it is known to be taken into account (Nagios supervision);

The first part of the `tisbackup.ini` file, starting with the `[Global]` tag, determines:

- the path to the folder where the backups will be stored;
- the maximum retention time of a backup (in days);
- the maximum delay before triggering a nagios critical message (in hours);
- possibly the limit of usable bandwidth;

The rest of the file lists the different backups to be made, with specific parameters for each type of backup:

- name of the directory in the backup;
- backup type;
- server name;
- directory (in case of a directory backup);
- directories to be excluded (idem);

- location of the ssh key to be used (private key on the backup server);
- name of the database (in case of mysql or postgresql database backup);
- ssh port number to use;
- database user and password (in case of mysql or postgresql database backup);

tisbackup.sql

`tisbackup.sql` is the **sqlite** database available on the backup server, in which the backup information of each of the backed up areas is stored. It is used in particular to gather the information necessary for Nagios.

TISbackup GUI

Also developed in python, TISbackup GUI is a graphical interface that allows you to:

- visualize the last backups;
- export a backup to a USB media;
- visualize the backups to be made;



You may now go on to the next step and *install TISBackup on your Debian*.

1.2.2 Installing and configuring TISBackup on Debian

Setting up the GNU/Linux Debian server

In order to install a fresh Debian Linux 10 *Buster* (physical or virtual) without graphical interface, please refer to the [Debian GNU/Linux Installation Guide](#).

Configuring network parameters

The different parameters presented below are not specific to TISBackup; you may adapt them as required for your environment. Modify the following files in order to get a proper named FQDN (Fully Qualified Domain Name) and network addressing strategy. In the following example:

- the FQDN name is `srvbackup.mydomain.lan`;
- the short-name of the TISBackup Server is `srvbackup`;
- the DNS (Domain Name Service) suffix is `mydomain.lan`;
- the IP address is `10.0.0.10/24`;

Configuring the name of the Debian server

Hint: The short name of the future TISBackup server must not be longer than **15 characters** (the limit is due to *sAMAccountName* restriction in Active Directory).

The name of the TISBackup server must be a FQDN, that is to say it has both the server name and the DNS suffix.

- modify the `/etc/hostname` file and write the FQDN of the server;

```
# /etc/hostname of the TISBackup server
srvbackup.mydomain.lan
```

- configure the `/etc/hosts` file, be sure to put both the FQDN and the short name of the server;

```
# /etc/hosts of the server
127.0.0.1    localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
10.0.0.10   srvbackup.mydomain.lan      srvbackup
```

Hint:

- on the line defining the DNS server IP address, be sure to have the IP of the server (not 127.0.0.1), then the FQDN, then the short name;
 - do not change the line with *localhost*;
-

Configuring the IP address of the Debian server

- configure the IP address of the Debian Server in the `/etc/network/interfaces`;

```
# /etc/network/interfaces of the Debian server
auto eth0
iface eth0 inet static
    address 10.0.0.10
    netmask 255.255.255.0
    gateway 10.0.0.254
```

- apply the network configuration by rebooting the machine with a `reboot`;
- if it has not already been done, create the DNS entry for the Server in the Organization's Active Directory;
- after reboot, configure the system language in English in order to have non-localized logs for easier searching of common errors;

```
apt install locales-all
localectl set-locale LANG=en_US.UTF-8
localectl status
```

- check that the machine clock is on time (with NTP installed);

```
dpkg -l | grep ntp
service ntp status
date
```

Hint: If the NTP package is not installed.

```
apt install ntp
systemctl enable ntp
systemctl start ntp
```

- update and upgrade your Debian;

```
apt update
apt upgrade -y
```

- install systemd;

```
apt install systemd
```

- restart the Debian server;

```
reboot
```



The Debian server is now ready. You may now go on to the next step and *install TISBackup on your Debian*.

Installing the TISBackup server on Debian Linux

- install the required dependencies:

```
apt-get install unzip ssh rsync python-paramiko python-pyvmomi python-pexpect
```

- retrieve the git sources from <https://github.com/tranquilit/TISbackup> and place them in the /opt folder on your server:

```
cd /opt/
wget --no-check-certificate https://github.com/tranquilit/TISbackup/archive/master.zip
unzip master.zip
mv TISbackup-master tisbackup
chmod 755 /opt/tisbackup/tisbackup.py
ln -sb /opt/tisbackup/tisbackup.py /usr/local/bin/tisbackup
```

- the **tisbackup** command must return all *tisbackup* actions directly to you. For more information on the actions go to *the section on using TISBackup*;

```
[root@srvbackup.mydomain.lan tisbackup]# tisbackup
ERROR : You must provide one action to perform
Usage: tisbackup -c configfile action

TIS Files Backup system.
```

(continues on next page)

(continued from previous page)

```
action is either :  
backup : launch all backups or a specific one if -s option is used  
cleanup : removed backups older than retention period  
checknagios : check all or a specific backup against max_backup_age parameter  
dumpstat : dump the content of database for the last 20 backups  
retryfailed : try to relaunch the last failed backups  
listdrivers : list available backup types and parameters for config inifile  
exportbackup : copy latest OK backups from local to location defined by --exportdir_  
↳parameter  
register_existing : scan backup directories and add missing backups to database
```

Configuring TISBackup

- create the directory for TISBackup configuration files:

```
mkdir /etc/tis/
```

- in the directory /opt/tisbackup/samples/, you will find the files config.ini.sample and tisbackup-config.ini which you can use as examples. Copy one of these two files into the /etc/tis directory and we will describe in the next section how to customize this files;

```
cp /opt/tisbackup/samples/tisbackup-config.ini.sample /etc/tis/tisbackup-config.ini
```

Launching the backup scheduled task

```
cp /opt/tisbackup/samples/tisbackup.cron /etc/cron.d/tisbackup
```

- modify the /etc/cron.d/tisbackup file to indicate when to launch the task;

Generating the public and private certificates

- as root:

```
ssh-keygen -t rsa -b 2048
```

- press Enter for each one of the steps;



You may now go on to the next step and *configure the backup jobs for your TISBackup*.

Setting up the graphical user interface for the TISBackup server

```
apt-get install python2.7 python-simplejson python-flask python-setuptools sudo
python /usr/lib/python2.7/dist-packages/easy_install.py "huey<=0.4.9"
```

```
cp /opt/tisbackup/samples/tisbackup_gui.ini /etc/tis/
cp /opt/tisbackup/scripts/tisbackup_gui /etc/init.d/tisbackup_gui
cp /opt/tisbackup/scripts/tisbackup_huey /etc/init.d/tisbackup_huey
chmod +x /etc/init.d/tisbackup_gui
chmod +x /etc/init.d/tisbackup_huey
update-rc.d tisbackup_huey defaults
update-rc.d tisbackup_gui defaults
```

You can now access your interface through the url of your TISBackup server on port 8080.

Type: rsync+ssh

Server	Backup	Directory
192.168.149.209	srvdev-slash	/

Type: mysql

Server	Backup	Database
192.168.149.209	srvdev-mysql	*

Copyright (c) 2015 Tranquil IT Systems. All rights reserved. Design by TIS.

Fig. 1: TISBackup Web interface

1.2.3 Configuring the backup jobs

The configuration of the backups is done in an `.ini` file, by default `/etc/tis/tisbackup-config.ini`:

- a global section where general parameters are specified;
- then for each backup a section will be created;

[global] section

Here are the mandatory parameters of the global section.

- the beginning of the global section starts with:

```
[global]
```

- specify directory where to store backups:

```
backup_base_dir = /backup/data/
```

- define the maximum age of the backups (variable used by the cleanup function):

```
backup_retention_time=140
```

- define the maximum time in hours between each backup. When this time is exceeded, then **checknagios** goes critical:

```
maximum_backup_age=30
```

Another non-mandatory parameter allows to define the rsync compression level: `compression_level=7`.

Backup types

Note: to test with a Windows box

Globally, the backups are done through an SSH connection and the steps are:

- creating the **section** in the configuration file;
- installing ssh on the Linux client;
- making an ssh key exchange between the tisbackup server and the client to back up;

Here are the different types of backup possible with **tisbackup**.

Backing up a MySQL database

```
[srvintranet_mysql_mediawiki]
type=mysql+ssh
server_name=srvintranet
private_key=/root/.ssh/id_dsa
db_name=mediawiki
db_user=user
db_passwd=password
```

Mandatory parameters:

- [srvintranet_mysql_mediawiki]: name of the section starts with the name you give to it;
- type: specifies the backup type for the Mysql database dump;
- server_name: defines the server to be backed up by its DNS name or IP address;
- private_key: defines the name of the private key to be used to connect to the client;
- db_name: defines the name of the database to dump;
- db_user: defines the name of a user with the right to dump on the basis of;
- db_passwd: defines the user's password;

Backing up a PostgreSQL database

```
[srvasterisk-pgsql]
type=pgsql+ssh
server_name=srvasterisk
private_key=/root/.ssh/id_rsa
db_name=asterisk
```

Mandatory parameters:

- [srvasterisk-pgsql]: name of the section starts with the name you give to it;
- type: specifies the backup type for the Mysql database dump;
- server_name: defines the server to be backed up by its DNS name or IP address;
- private_key: defines the name of the private key to be used to connect to the client;
- db_name: defines the name of the database to dump;

Backing up a file server

```
[srvfiles-home]
type=rsync+ssh
server_name=srvfiles
remote_dir=/home
private_key=/root/.ssh/id_dsa
exclude_list=".mozilla",".thunderbird",".x2go","*.avi"
bwlimit = 100
```

Mandatory parameters:

- `[srvfiles-home]`: name of the section starts with the name you give to it;
- `type`: specifies the backup type for the Mysql database dump;
- `server_name`: defines the server to be backed up by its DNS name or IP address;
- `remote_dir`: defines the folder on the remote host to backup;
- `private_key`: defines the name of the private key to be used to connect to the client;

Attention: In case of Windows client, specificities are to be expected:

By default we use the root user for backups, for windows we will use the Administrator account (pay attention to the sensitive box).

```
remote_user=Administrator
```

Through **cygwin**, the directory to be backed up will always start with `/cygdrive`, so it must be specified in the `remote_dir` parameter.

```
remote_dir=/cygdrive/c/WINDOWS/
```

Hint: Other non-mandatory parameters can be used. The `listdrivers` option allows you to see them. The two most frequently used parameters are:

- `exclude_list`: defines the files to be excluded from the backup;
 - `bwlimit`: defines the maximum speed of the backup;
-

Backing up a XenCenter virtual machine

On local storage

```
[wsmanage]
type=xen-xva
xcphost=srvxen1
server_name=wsmanage
password_file=/root/xen_passwd
backup_retention_time=2
halt_vm=True
enable_https=False
```

Mandatory parameters:

- `[wsmanage]`: name of the section starts with the name you give to it;
- `type`: specifies the backup type for the Mysql database dump;
- `xcphost`: defines the XCP server where the VM is found by its DNS name or IP;
- `server_name`: defines the server to be backed up by its DNS name or IP address;
- `password_file`: defines a file where are stored the user and the password to be used for exporting the `.xva` file;

- `backup_retention_time`: defines the maximum number of exports for the virtual machine;
- `halt_vm`: **True** = stop the virtual machine then export, **False** = snapshot the virtual machine then export the `xva` without stopping the virtual machine;
- `enable_https`: activate or deactivate https protocol for transfer;

On remote storage

```
[srvads-copy]
type=copy-vm-xcp
server_name=srvxen1
vm_name=srvads
storage_name=iscsi-sr1
password_file=/root/xen_passwd
network_name=test-dcardon
max_copies=3
```

Mandatory parameters:

- `[srvads-copy]`: name of the section starts with the name you give to it;
- `type`: specifies the backup type for the Mysql database dump;
- `server_name`: defines the server to be backed up by its DNS name or IP address;
- `vm_name`: defines the virtual machine to be backed up (its name-label in XCP);
- `storage_name`: defines the storage to where to copy the virtual machine (its name-label in XCP);
- `password_file`: defines a file where are stored the user and the password to be used for exporting the `.xva` file;
- `network_name`: defines the network to which to copy the VM (its name-label in XCP);
- `max_copies`: maximum number of exports for the virtual machine;

XenCenter metadata

```
[srvxen1-metadata] type=xcp-dump-metadata server_name=srvxen1 password_file=/root/xen_passwd
```

Mandatory parameters:

- `[srvxen1-metadata]`: name of the section starts with the name you give to it;
- `type`: specifies the backup type for the Mysql database dump;
- `server_name`: defines the server to be backed up by its DNS name or IP address;
- `password_file`: defines a file where are stored the user and the password to be used for exporting the `.xva` file;

Attention: For maximum security put the password file in the root directory with read-write access only for it.

```
vi /root/xen_passwd
```

example of the content of the file:

```
user
password
```

implementation of restricted rights

```
chmod 600 /root/xen_passwd
```

1.2.4 Using TISBackup

As seen in the *section on installing TISbackup*, once the TISBackup installation is up and running, we have the choice of these actions:

```
backup: launch all backups or a specific one if -s option is used
cleanup: removed backups older than retention period
checknagios: check all or a specific backup against max_backup_age parameter
dumpstat: dump the content of database for the last 20 backups
retryfailed: try to relaunch the last failed backups
listdrivers: list available backup types and parameters for config inifile
exportbackup: copy latest OK backups from local to location defined by --exportdir parameter
register_existing: scan backup directories and add missing backups to database
```

The 3 following options can be used with any tisbackup action.

- the `-c config_file` option allows to specify a backup file, by default `/etc/tis/tisbackup-config.ini` is used:

```
tisbackup backup -c /etc/toto/test-config.ini
```

- the `-s section_name` option allows to launch only the action on the specified section:

```
tisbackup backup -s section_name
```

- the `-d` option allows you to simulate an action in order to see the commands launched by it.

```
tisbackup backup -d
```

- backup** launches a backup action:

```
tisbackup backup
```

- cleanup** removes backups older than the time specified in the `backup_retention_time` parameter of the configuration file:

```
tisbackup cleanup
```

- checknagios** allows the backup information to be uploaded to the nagios monitoring server:

```
tisbackup checknagios
```

- dumpstat** displays all information about the last 20 backups in tabular format:

```
tisbackup dumpstat
```

- retryfailed** restarts only the backup of the failed sections:

```
tisbackup retryfailed
```

- **listdrivers** lists all the possible types of backups and their parameters:

```
tisbackup listdrivers
```

- **exportbackup** copies the last good backup to a directory, you must use the `--exportdir` option to specify or copy the export:

```
tisbackup exportbackup --exportdir example_directory
```

- **register_existing** checks the backup directory and saves information from previous backups to tisbackup in the database;

Exporting backups

With this procedure, you will be able to export your backups on USB Hard Disk Drives for your off-line backup needs.

The partition of your HDD must be **ext4** formatted and labeled *tisbackup*.

```
fdisk /dev/xvdc
Command (m for help): n
Select (default p): p
Partition number (1-4, default 1): 1
"Enter"
"Enter"
Command (m for help): w

mkfs.ext4 /dev/xvdc1
e2label /dev/xvdc1 tisbackup
```

1.2.5 Contacting Tranquil IT

- Tranquil IT: <https://www.tranquil.it/>
- Twitter: https://twitter.com/tranquil_it

1.2.6 Screenshots of TISBackup

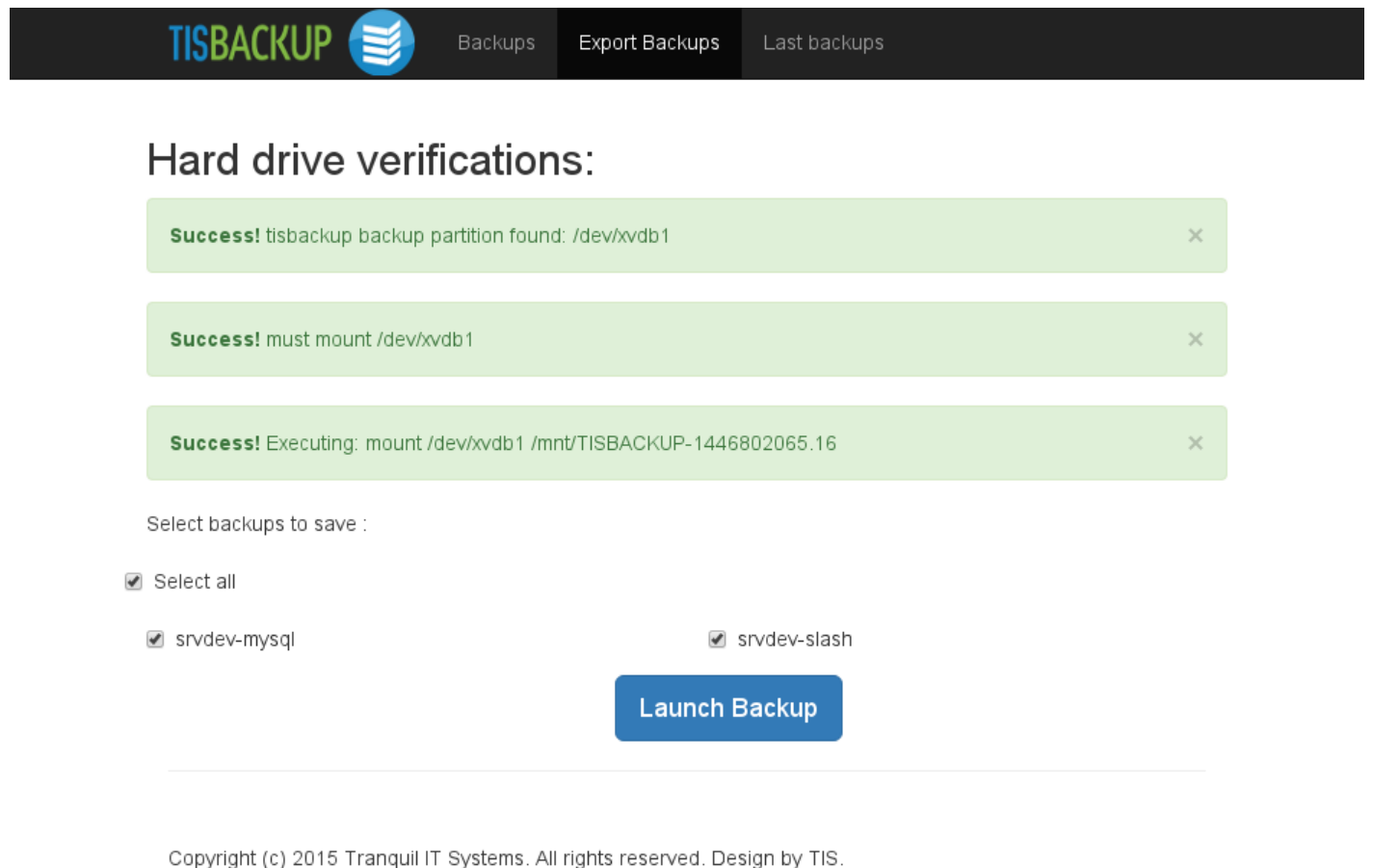



Fig. 2: Exporting a backup to an external USB HDD


Backups Export Backups **Last backups**

Search:

Backup start	Server name	Backup name	Backup duration	Status	Written bytes
2015/11/5 5:07:25 PM	192.168.149.209	srvdev-mysql	0:00:02	OK	229.5 KiB
2015/11/5 5:07:24 PM	192.168.149.209	srvdev-slash	0:00:01	OK	12.2 KiB
2015/11/5 5:07:07 PM	192.168.149.209	srvdev-mysql	0:00:00	ERROR	0 B
2015/11/5 5:07:05 PM	192.168.149.209	srvdev-slash	0:00:02	OK	543.2 KiB
2015/11/5 5:03:43 PM	192.168.149.209	srvdev-slash	0:01:15	OK	1.5 GiB
2015/11/5 5:02:56 PM	192.168.149.209	srvdev-slash	0:00:01	ERROR	0 B
Total Written Bytes			1.6 GB		
Total Duration time			0:01:22		

Showing 1 to 6 of 6 entries
First Previous Next Last

Number of rows per page :


☒ Backup start
☐ Backup end
☒ Server name
☒ Backup name
☒ Backup duration

☒ Written bytes
☐ Written files count
☐ Total files count
☐ Total bytes

☐ Backup location
☐ Description
☐ Log
☐ Type
☒ Status

Copyright (c) 2015 Tranquil IT Systems. All rights reserved. Design by TIS.

Fig. 3: Status of exported backups

BackupsExport BackupsLast backups

Type: rsync+ssh

Server	Backup	Directory
192.168.149.209	srvdev-slash	/

Type: mysql

Server	Backup	Database
192.168.149.209	srvdev-mysql	*

Copyright (c) 2015 Tranquil IT Systems. All rights reserved. Design by TIS.

Fig. 4: TISBackup Web interface

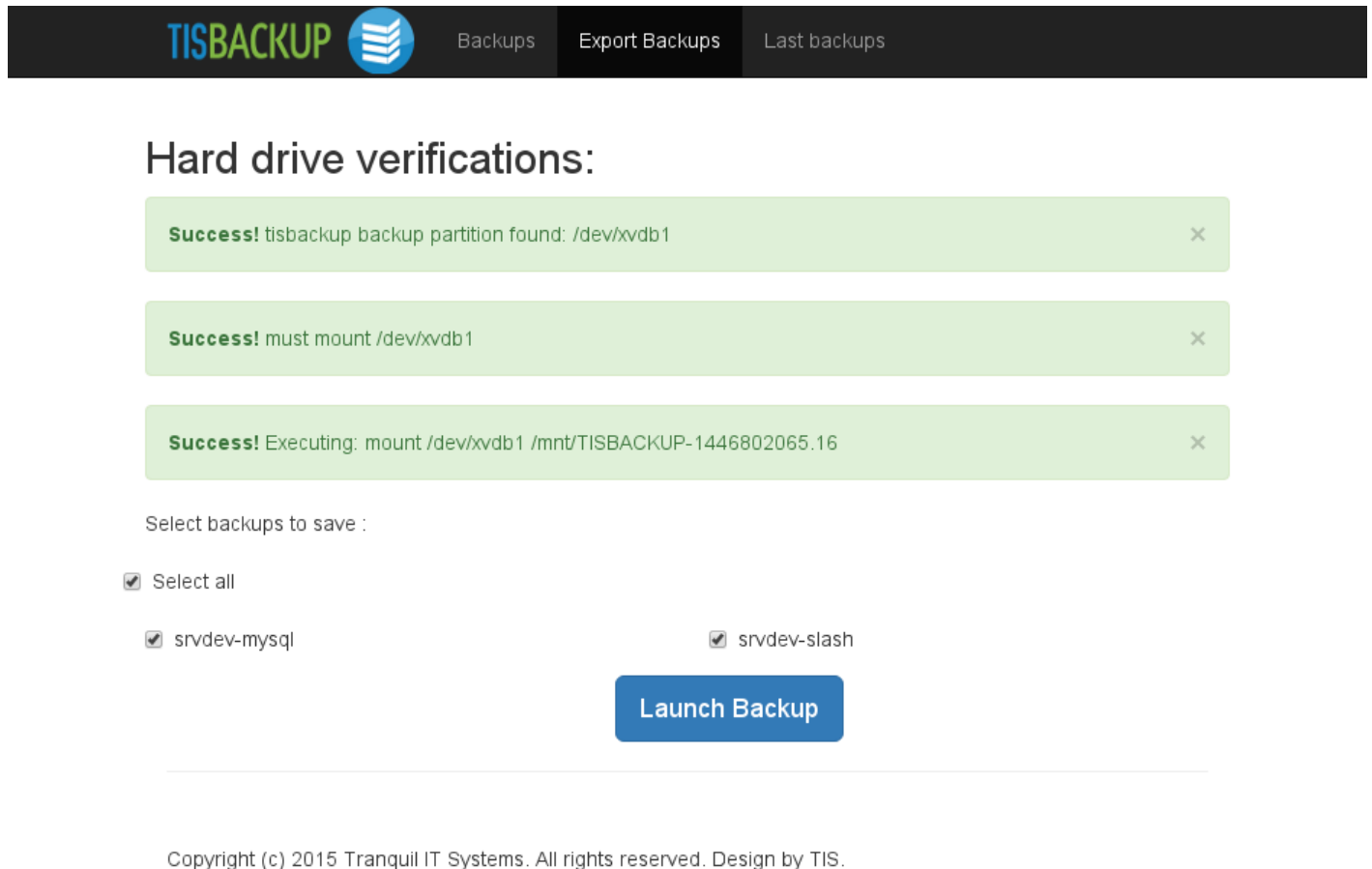



Fig. 5: Exporting a backup to an external USB HDD


Backups
Export Backups
Last backups

Search:

Backup start	Server name	Backup name	Backup duration	Status	Written bytes
2015/11/5 5:07:25 PM	192.168.149.209	srvdev-mysql	0:00:02	OK	229.5 KiB
2015/11/5 5:07:24 PM	192.168.149.209	srvdev-slash	0:00:01	OK	12.2 KiB
2015/11/5 5:07:07 PM	192.168.149.209	srvdev-mysql	0:00:00	ERROR	0 B
2015/11/5 5:07:05 PM	192.168.149.209	srvdev-slash	0:00:02	OK	543.2 KiB
2015/11/5 5:03:43 PM	192.168.149.209	srvdev-slash	0:01:15	OK	1.5 GiB
2015/11/5 5:02:56 PM	192.168.149.209	srvdev-slash	0:00:01	ERROR	0 B
Total Written Bytes			1.6 GB		
Total Duration time			0:01:22		

Showing 1 to 6 of 6 entries
First
Previous
Next
Last

Number of rows per page :

☒ Backup start
☐ Backup end
☒ Server name
☒ Backup name
☒ Backup duration

☒ Written bytes
☐ Written files count
☐ Total files count
☐ Total bytes

☐ Backup location
☐ Description
☐ Log
☐ Type
☒ Status

Copyright (c) 2015 Tranquil IT Systems. All rights reserved. Design by TIS.

Fig. 6: Status of exported backups

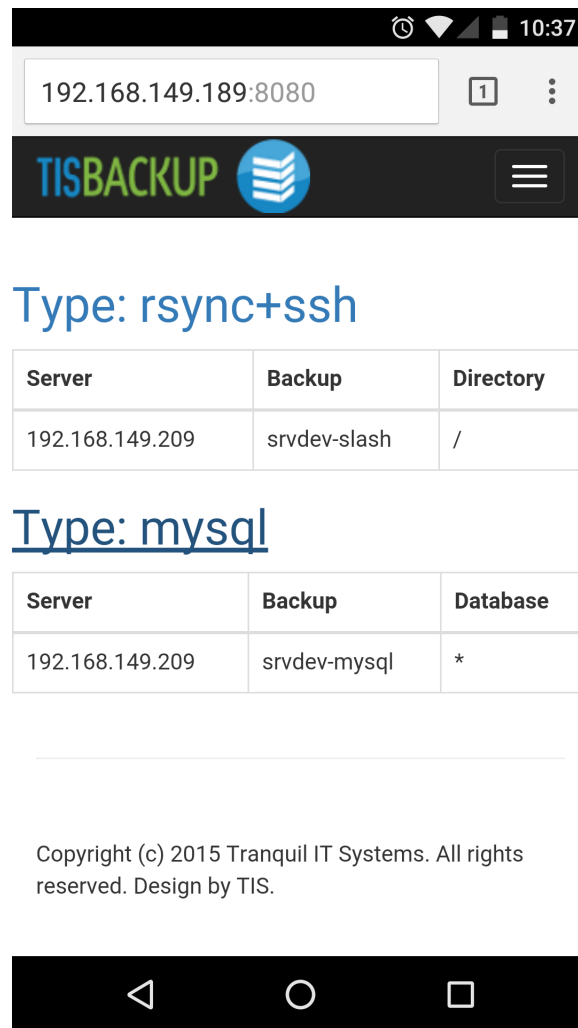


Fig. 7: Overview of current backups

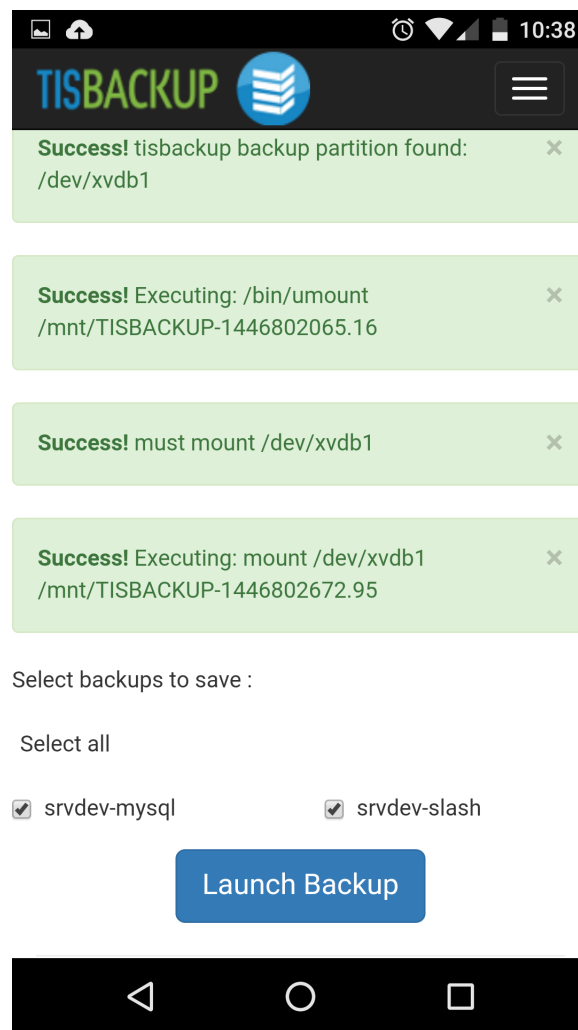


Fig. 8: Overview of successful backups

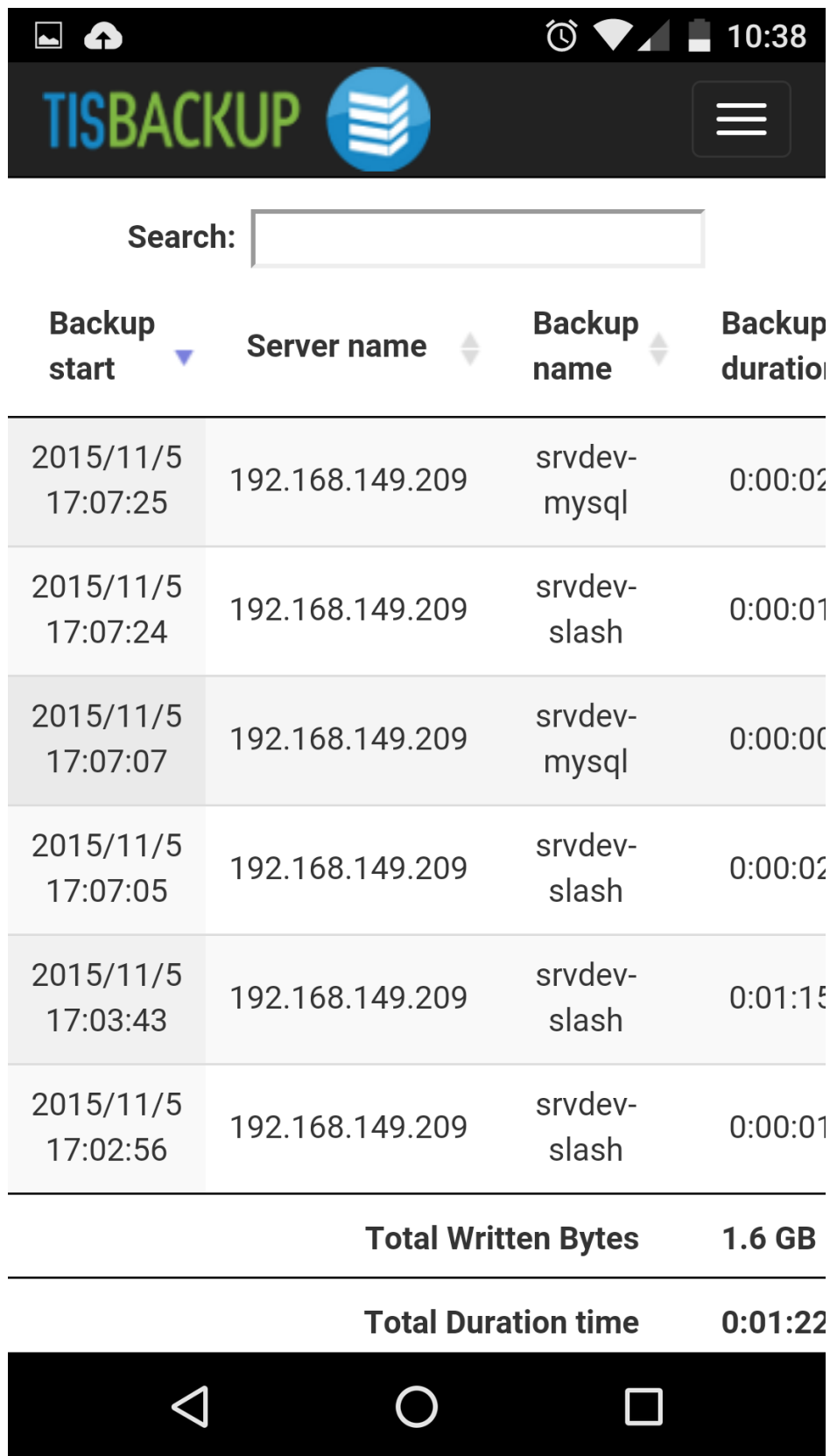
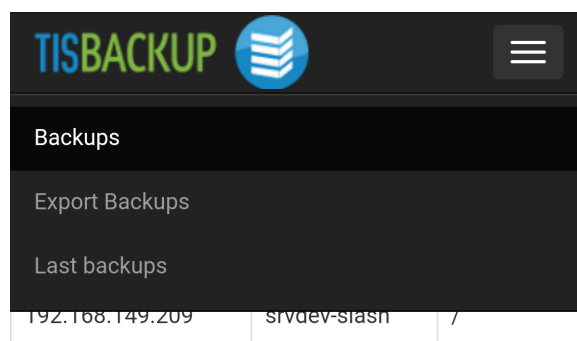


Fig. 9: Searching for past backups



Type: mysql

Fig. 10: TISBackup action menu

INDICES AND TABLES

- genindex
- search