



**TeamDrive Registration Server Virtual
Appliance Installation and
Configuration**

Release 3.0.018.2

Lenz Grimmer, Paul McCullagh, Eckhard Pruehs

November 12, 2014

1	Copyright Notice	1
2	Trademark Notice	3
3	Introduction	5
3.1	Requirements	5
3.2	Hardware Requirements	7
3.3	Main Software components	7
4	Virtual Appliance Installation and Configuration	9
4.1	Download and import the Virtual Appliance	9
4.2	First Boot and Initial Configuration	9
4.3	Changing the Default MySQL Database Passwords	13
5	Configuring and Testing the MySQL Database Connections	15
5.1	Configuring the Registration Server's MySQL configuration	15
5.2	Administration Console MySQL Configuration	16
6	Registration Server Configuration	19
6.1	Importing XML With Initial Configuration Values to the Database	19
6.2	Submitting your Server's Authorization Sequence	21
6.3	Post-install Checks	22
7	Starting and stopping the TeamDrive Registration Server components	25
7.1	Starting services manually	25
7.2	Stopping services manually	25
7.3	Enabling Service Autostart	25
7.4	Logging into the Administration Console	26
8	Troubleshooting	27
8.1	List of relevant log files	27
8.2	Common errors	27
9	Release Notes - Version 3.0.018	31
9.1	Change Log - Version 3.0.018	33
10	Release Notes - Version 3.0.017	35
11	Appendix	39
11.1	Abbreviations	39

COPYRIGHT NOTICE

Copyright © 2014, TeamDrive Systems GmbH. All rights reserved.

TeamDrive Systems GmbH

<https://www.teamdrive.com>

Max-Brauer-Allee 50

22765 Hamburg, Germany

Email: info@teamdrive.com

TRADEMARK NOTICE

Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.

Apache HTTP Server, Apache, and the Apache feather logo are trademarks of The Apache Software Foundation.

MySQL is a registered trademark of Oracle and/or its affiliates.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation.

AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices.

VMware is a trademark or registered trademark of VMware, Inc. in the United States and/or other jurisdictions.

“Amazon Web Services”, “Amazon S3” are trademarks of Amazon.com, Inc. or its affiliates in the United States and/or other countries.

“Red Hat Linux” and “CentOS” are trademarks of Red Hat, Inc. in the U.S. and other countries.

All other names and trademarks used herein are the property of their respective owners.

INTRODUCTION

The TeamDrive Registration Server Virtual Appliance offers a pre-installed and ready-to-run TeamDrive Registration Server suitable for deployment in a virtualized environment like VMWare.

This document will guide you through the deployment and initial installation of the Virtual Appliance and the configuration of the TeamDrive Registration Server.

3.1 Requirements

3.1.1 Required Skills

When installing the TeamDrive Registration Server, we assume that you have basic knowledge of:

- VMware: importing and deploying virtual machines, configuring virtual networking and storage (when installing the TeamDrive Server components in a virtual environment or when using a pre-installed Virtual Appliance)
- Linux system administration:
 - Adding/configuring software packages
 - Editing configurations files with a text editor (e.g. `vi` or `nano`)
 - Starting/stopping services
 - Creating user accounts
 - Assigning file ownerships and privileges
 - Creating and mounting file systems
 - Setting up environment variables
- Apache web server: installation and configuration, adding and enabling modules, modifying configuration files
- MySQL Database: installation and configuration, administration/maintenance, using the MySQL command line client, basic SQL
- Basic knowledge of application server technology (ideal but not required: knowledge of the PrimeBase Application Server)

3.1.2 Network Requirements

The system must have IP connectivity, using a fixed IP address and a resolvable fully qualified domain name. The Registration Server itself needs to be able to properly resolve host names, too.

If the Registration Server is located behind a firewall, please ensure that it is reachable via HTTP (TCP port 80) by the TeamDrive Clients. HTTPS access (TCP port 443) is only required for accessing the web-based Administration Console and can be restricted based on your requirements.

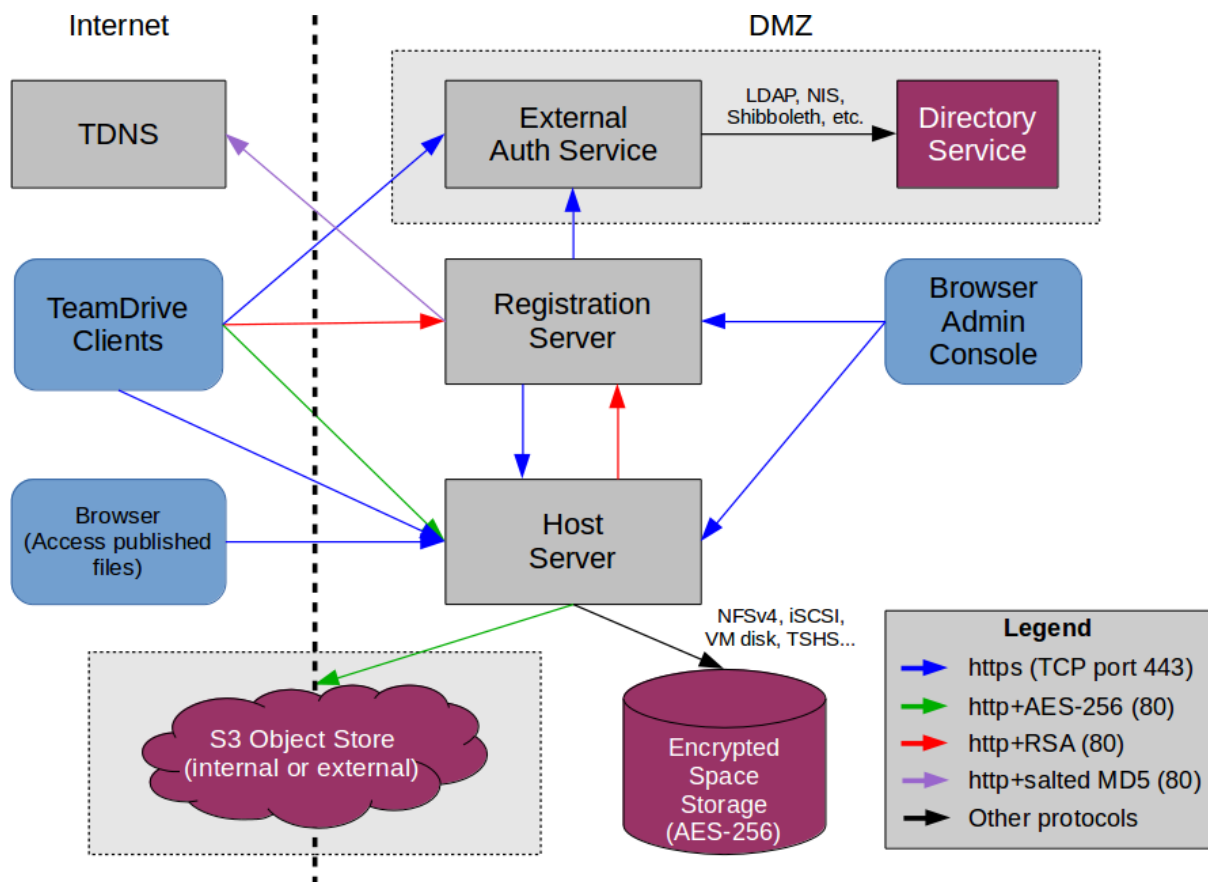


Figure 3.1: TeamDrive Enterprise Server Networking Overview

If the Registration Server has been configured to contact the TeamDrive TDNS service, it needs to be able to establish outgoing HTTP connections (TCP port 80) to <http://tdns.teamdrive.net/>.

For the initial registration and the exchange of cryptographic keys, the Host Server must be able to contact the Registration Server via HTTP (TCP port 80). After the registration and activation, no further connections from the Host Server to the Registration Server will be established.

To perform API calls (e.g. to create new Space Depots or to query for existing Spaces for a particular user), the TeamDrive Registration Server must be able to establish outgoing HTTP or HTTPS connections to the TeamDrive Hosting Service.

If you use External Authentication for Authenticating user accounts, the Registration Server needs to be able to establish outgoing HTTP or HTTPS connections to the host providing the external Authentication Service.

3.2 Hardware Requirements

The TeamDrive Registration Server Virtual Appliance is delivered in the form of a virtual machine image. Its main technical specifications are:

- Supported platforms: VMWare vSphere 4 and VMWare vSphere 5 (VMWare Workstation 7 or Oracle VM VirtualBox can be used for testing purposes)
- Minimum VM Memory: 2 GB
- vCPUs: 2
- HDD: 100GB
- Guest OS: CentOS 6 (64-bit)

3.3 Main Software components

The TeamDrive Registration Server comprises the following components and modules:

- Apache Web Server 2.2
- PrimeBase Apache-Module `mod_pbt`
- PHP 5.3
- MySQL 5.1 (or later) Database Server
- PrimeBase Application Environment version 4.5.48
- Modules for the PrimeBase Application Server

The PrimeBase components are standard software packages that are not TeamDrive-specific.

VIRTUAL APPLIANCE INSTALLATION AND CONFIGURATION

4.1 Download and import the Virtual Appliance

A .zip Archive containing the virtual appliance's disk image and VM configuration can be obtained from the following URL:

<http://s3download.teamdrive.net/HostServer/TD-Registration-Server-CentOS6-64bit-3.0.018.2.zip>

Download and unzip the archive. The archive contains four files, a virtual disk image (.vmdk), two virtual machine description files (.ovf) and a manifest file (.mf), containing the file names and SHA1 checksums.

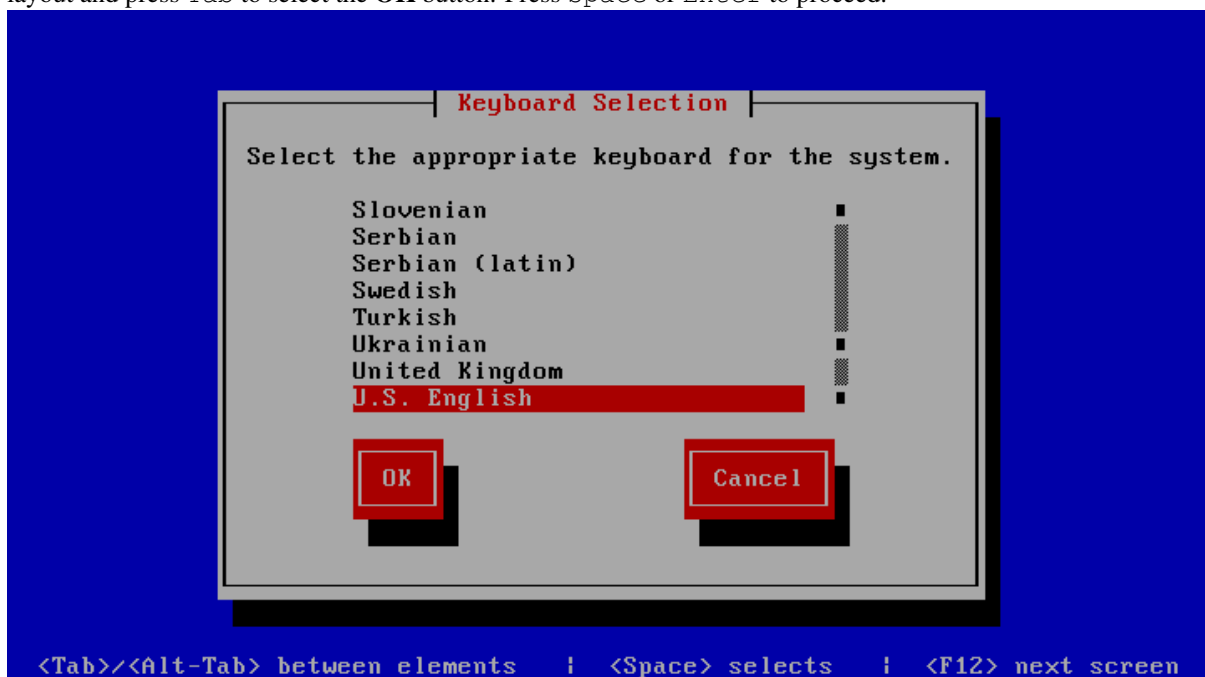
Import the virtual machine image according to the documentation of your virtualization technology and adjust the VM parameters (e.g. number of virtual CPUs, RAM) based on your requirements, if necessary.

Start up the virtual machine and observe the virtual machine's console output.

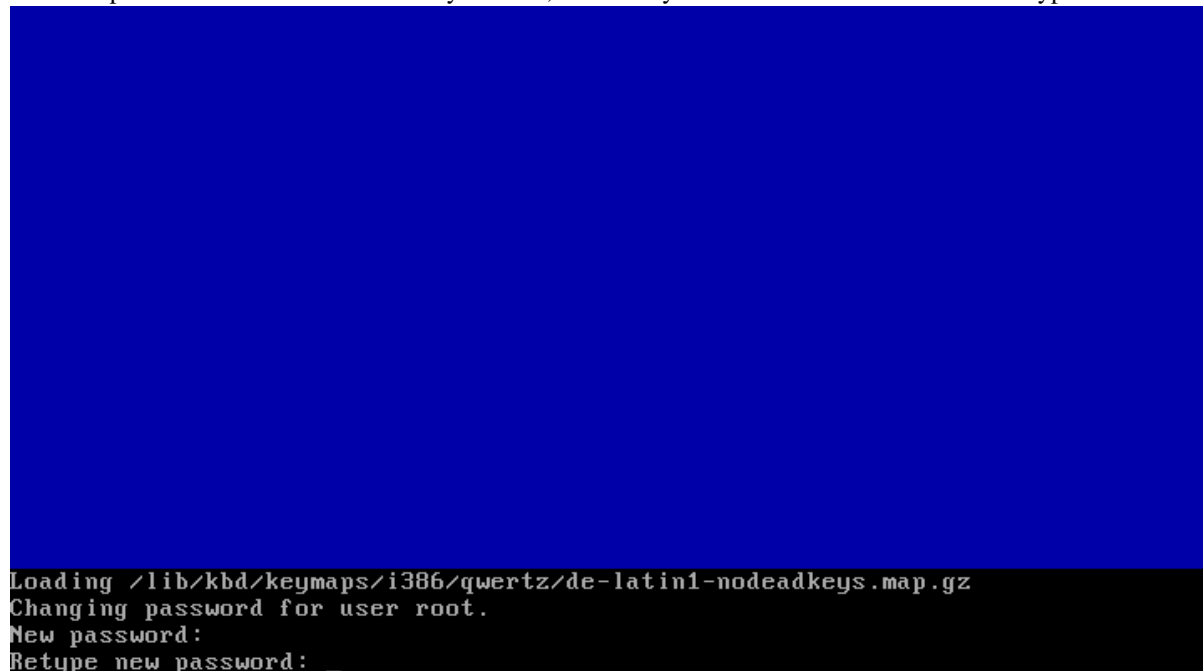
4.2 First Boot and Initial Configuration

When you boot up the VM image for the first time, the Operating System will perform a number of initial configuration steps, to customize the virtual machine for your environment. For more details on these individual steps, please refer to the *Red Hat Enterprise Linux 6 Installation Guide* at https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Installation_Guide/index.html

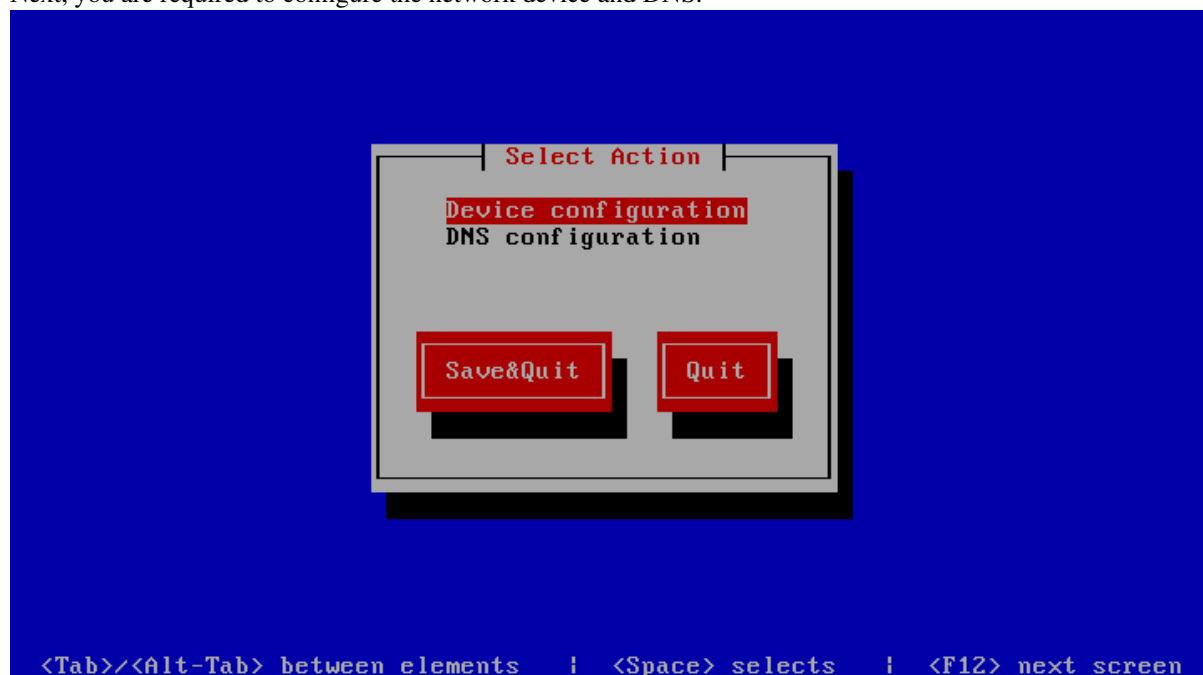
The first screen will prompt you to select your keyboard layout. Use the cursor keys to select the appropriate layout and press Tab to select the **OK** button. Press Space or Enter to proceed.



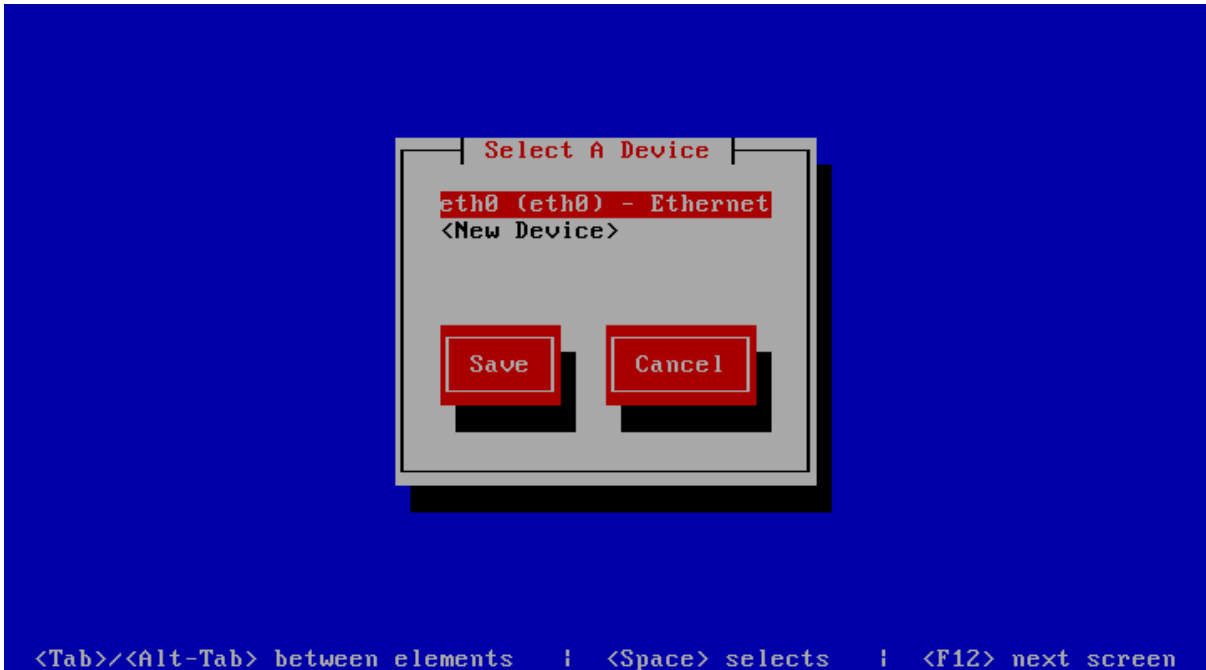
After loading the appropriate keymap, you will now be prompted to enter the password for the *root* user account. Choose a strong password here — the *root* user has full administrative privileges on a Linux system. The password won't be printed on the screen for security reasons, therefore you need to enter it twice to avoid typos.



Next, you are required to configure the network device and DNS.

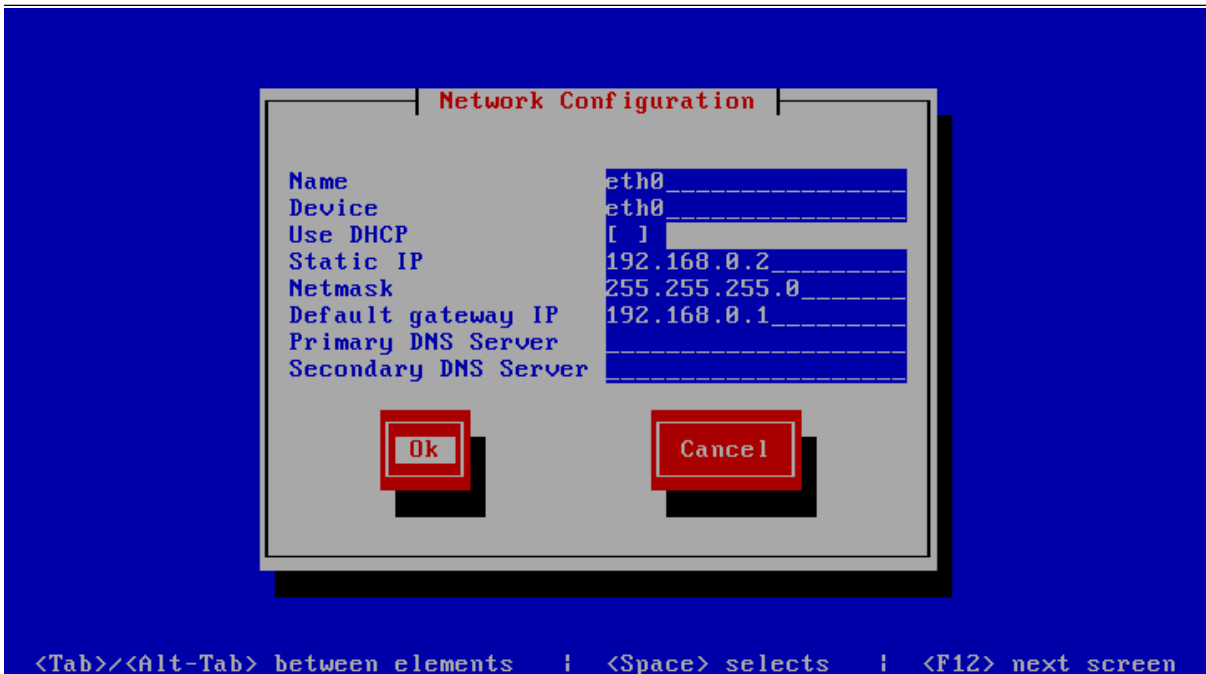


Press `Enter` to enter the device configuration screen. Select the network device which is usually named **eth0**. The hardware description (e.g. **AMD PCnet32** or **Ethernet**) depends on the virtual network card that your virtual machine has been configured with. Press `Enter` to get to the device configuration details.

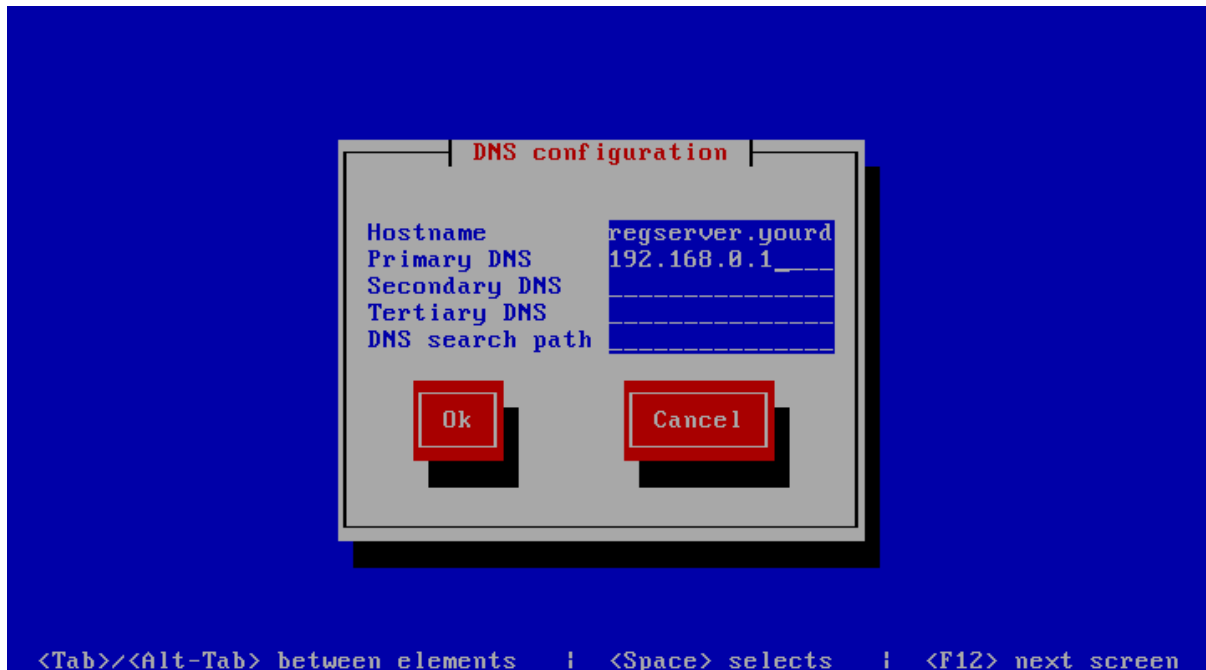


The network configuration screen allows you to specify the IP address and other network-related settings. Adjust these to your local requirements. Use the **Tab** key to move between elements. Select the **Ok** button and press **Space** or **Enter** to apply your changes. In the device selection screen, select the **Save** button to save the device configuration.

Note: Note that you should not use dynamic IP addresses (DHCP) for the TeamDrive Registration Server. The TeamDrive Service depends on static IP addresses and requires a resolvable fully qualified domain name.



Next, select the **DNS configuration** screen from the network configuration and adjust the DNS settings to match your environment. Enter your Registration Server's host name (e.g. `regserver.yourdomain.com`) in the **Hostname** field and add your DNS server's IP address(es). Select the **Ok** button to apply the DNS changes.

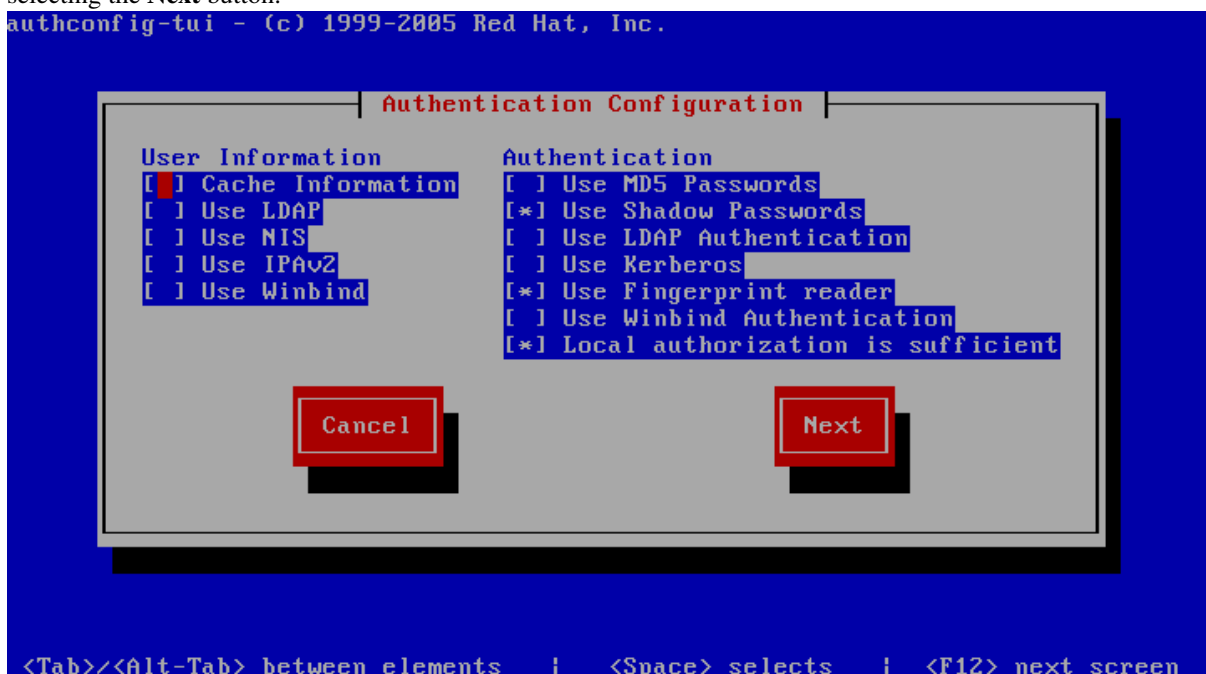


Select the **Save&Quit** button to proceed with the installation.

Note: A cloned CentOS image in a VMWare environment might exhibit problems updating the network interface. If you are observing issues when configuring the network interface, please follow these instructions: <http://alexcline.net/2011/11/15/reconfiguring-network-interfaces-in-centosrhel-systems-cloned-with-vmcenter/>

The next screen allows you to modify the local user authentication configuration. Unless you have specific requirements for your local environment, it's safe to keep the proposed defaults and proceed with the installation by selecting the **Next** button.

authconfig-tui - (c) 1999-2005 Red Hat, Inc.



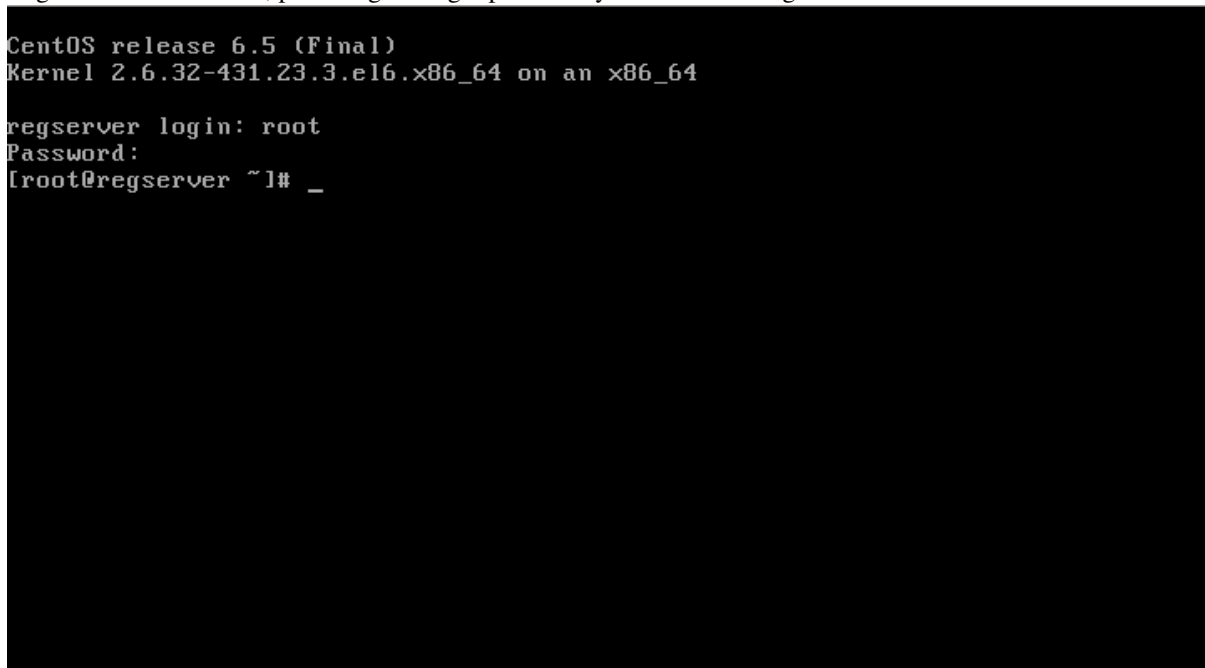
The following screen will ask you to select which services should be started at system bootup time. Adjust these based on your requirements and local policies. Also make sure that the MySQL database (service `mysqld`) is enabled. Select the **Ok** button to proceed.

Note: Do not enable the Apache http Server (`httpd`) yet — this should only be done after the Registration Server has been configured correctly.



The system will now proceed with the bootup process until you reach the login prompt.

Log in as the `root` user, providing the login password you defined during the initial installation.



4.3 Changing the Default MySQL Database Passwords

The TeamDrive Registration Server Virtual Appliance uses the following default passwords for the MySQL database. We strongly suggest changing the passwords of the MySQL users `root` and `teamdrive` before connecting this system to a public network.

Account type	Username	Password (default)	New Password
MySQL Database Server	root	teamdrive	
MySQL Database Server	teamdrive	teamdrive	

To change the passwords for the MySQL `root` and `teamdrive` user, please use the following commands. First change the password for the root user:

```
[root@regserver ~]# mysqladmin -u root -pteamdrive password
Warning: Using a password on the command line interface can be insecure.
New password: <new password>
Confirm new password: <new password>
```

Next, log into the MySQL database as the `root` user (using the new password) and change the password for the user `teamdrive`:

```
[root@regserver ~]# mysql -u root -p
Enter password: <new password>

[...]

mysql> SET PASSWORD FOR 'teamdrive'@'localhost' = PASSWORD('<new password>');
Query OK, 0 rows affected (0.00 sec)

mysql> quit
Bye
```

Note: Take note of the new MySQL password for the `teamdrive` user, as you will need to change some configuration files using that password as outlined in the following chapters *Configuring the Registration Server's MySQL configuration* (page 15) and *Administration Console MySQL Configuration* (page 16).

CONFIGURING AND TESTING THE MYSQL DATABASE CONNECTIONS

5.1 Configuring the Registration Server's MySQL configuration

If the username, password or host name to connect to the MySQL database server have been changed from the installation defaults, you need to update the login credentials used by the Registration Server's PrimeBase Application Environment.

To change the MySQL login credentials for the Registration Server's database connections, edit the connection definitions TD2REG_WRITE and TD2REG_SLAVE in file /usr/local/primebase/setup/connect.def — the \su field identifies the user name, while the \sp field contains the MySQL user's password in plain text.

```
td2as:mem:\tCustom DAL\eflibpbvm.so
TD2REG_WRITE:mem:\xoHost=127.0.0.1;Charset=utf8;Reconnect=\tCustom
DAL\eflibpbvm.so\bOpenServer\nOpenServer\suteamdrive\spteamdrive\dbtd2reg
TD2REG_SLAVE:mem:\xoHost=127.0.0.1;Charset=utf8;Reconnect=\tCustom
DAL\eflibpbvm.so\bOpenServer\nOpenServer\suteamdrive\spteamdrive\dbtd2reg
```

Please make sure to change these values for **both** connections in the list. Each connection definition (beginning with td2as:, TD2REG_WRITE: and TD2REG_SLAVE:) must be on a single line (no white space or line breaks).

Note: Please note that this file contains the MySQL login credentials in plain text. Make sure to restrict the access permissions to this file so that only the root user and the Apache http Server (mod_pbt in particular) can open this file. The file ownerships should be set to root : apache, the file permissions should be set to "640".

You should test the connection after updating the MySQL login credentials. Change into the PrimeBase home directory and start the PrimeBase Automation Client application by typing pbac:

```
[root@regserver ~]# cd $PRIMEBASEHOME
[root@regserver primebase]# pbac
PrimeBase Automation Client.
Copyright 2007-2014, PrimeBase Systems GmbH.
Web:    http://www.primebase.net
E-mail: support@primebase.net
```

```
Select a connection by number, and Login:
Or enter 'A' to add, 'D' to delete, or 'E' to edit an entry.
Or enter 'T' to move an entry to the top of the list.
```

```
File: ./setup/connect.def
```

Alias	Protocol	Server
0	(exit without connecting)	
1	td2as	Internal/Runtime

```
2 TD2REG_WRITE      Internal/Runtime    OpenServer
3 TD2REG_SLAVE     Internal/Runtime    OpenServer
```

Do this by selecting the connection entry 2 TD2REG_WRITE from the connection list:

```
Alias                Protocol            Server
-----
0 (exit without connecting)
1 td2as              Internal/Runtime    OpenServer
2 TD2REG_WRITE       Internal/Runtime    OpenServer
3 TD2REG_SLAVE       Internal/Runtime    OpenServer
-----
```

```
Connection...: 2
User.....: teamdrive <Enter>
Password....: ***** <Enter>
1: Connected to "TD2REG_WRITE" as "teamdrive".
```

For a list of commands enter "#help"

```
1: 1> quit
1: Closed.
```

After confirming username and password the PBAC console should have started without any error messages.

To leave the PBAC console type `quit` and press `<Enter>`.

If you're seeing an error message at this stage, please consult the log file `/var/log/pbvm.log` and double check that the MySQL login credentials are correct. Also try to connect to the MySQL database using these values from the `mysql` command line client.

5.2 Administration Console MySQL Configuration

In order to being able to manage the Registration Server, the PHP-based Administration Console needs to be able to connect to the Registration Server's MySQL Database.

To define the username, password and hostname required to connect to the MySQL database server, you need to provide these login credentials in the configuration file `/var/www/html/tdlibs/globals.php`.

Update the connection string in the variable `$dsn2import` accordingly:

```
$dsn2import = 'mysql://teamdrive:teamdrive@127.0.0.1/td2reg';
```

The format is `mysql://<username>:<password>@<hostname>/databasename`. The database name usually does not need to be modified (`td2reg` is the default name).

As an alternative to providing the MySQL login credentials here, you can create a MySQL INI-style configuration file (e.g. `/etc/td-regserver.my.cnf`):

```
[regdb]
database=td2reg
user=teamdrive
password=teamdrive
host=localhost
```

The file must be readable by the user that the Apache http Server is running under, usually `apache`, but should otherwise be protected against unauthorized viewing (e.g. by setting the file ownerships to `apache:apache` and the access privileges to `600`).

Specify the location of this file by uncommenting and entering the full path in the configuration variable `$mysqlConfigFile` in `globals.php`. The values provided in this file take precedence over any login details entered in `$dsn2import`:

```
$mysqlConfigFile = '/etc/td-regserver.my.cnf';
```

REGISTRATION SERVER CONFIGURATION

6.1 Importing XML With Initial Configuration Values to the Database

The initial configuration of the Registration Server is performed based on values defined in the configuration file `RegServerSetup.xml` which must be placed in directory `/usr/local/primebase/setup/`.

The values included in this file are inserted into the Registration Server's MySQL database and in the PrimeBase configuration file `pbvm.env`.

Please refer to the *Registration Server Reference Guide* for more details on the individual options contained in this file.

If you have any questions about this step, please contact your TeamDrive representative or TeamDrive support via e-mail at support@teamdrive.net.

Most of these values in `RegServerSetup.xml` can be left "as is" — you can modify and fine-tune most of these later via the Admin Console, if required. The most important sections that need to be filled out and updated are `<RegServer>` (global Registration Server settings), `<distributor>` (provider-specific settings, including the provider user account required for logging into the Administration Console) and `<Email>` (SMTP server configuration).

Also, all occurrences of `regserver.yourdomain.com` should be replaced with the domain name of your Registration Server throughout this file.

To avoid "man-in-the-middle attacks", a cryptographic salt value is used to hash API requests. When you set up your own TeamDrive Host Server, this value needs to be provided during the installation and must match the one used on the Registration Server. You define this value in the tag `<APICheckSumSalt>` in the `RegServerSetup.xml` configuration file.

One way to create this random hash salt value is running the following commands:

```
% dd if=/dev/urandom of=/tmp/random bs=1k count=1000
1000+0 records in
1000+0 records out
1024000 bytes (1,0 MB) copied, 0,0944645 s, 10,8 MB/s
% shasum /tmp/random
4edd4f0a2e507b7c03b57cb414cbdeaf58f4a6ad /tmp/random
```

After installation this checksum can also be obtained on the Registration Server Administration Console, from the `APIChecksumSalt` system setting (**Edit Settings -> RegServer**).

Note: Your Registration Server needs to be registered with the global TeamDrive Name Service (TDNS) before it can be installed. You need to submit the following configuration options from your `RegServerSetup.xml` file to the TeamDrive support team before you can proceed:

- Your Provider Code (4 chars), as defined in `<TicketPrefix>`
- The Registration Server's name, as defined in `<RegServerName>`
- The Registration Server URL, as defined in `<RegServerURL>`

In return, you will receive your TDNS ID and a checksum, which you need to define in the settings <TDNSServerID> and <TDNSChecksum> respectively.

After you've updated the XML file to match your environment, change to /usr/local/primebase/setup, start pbac and choose connection 2 (TD2REG_WRITE) to execute the file RegServerSetup.pbt using the following commands:

```
[root@regserver ~]# cd $PRIMEBASEHOME/setup
[root@regserver setup]# pbac
PrimeBase Automation Client.
Copyright 2007-2014, PrimeBase Systems GmbH.
Web:    http://www.primbase.net
E-mail: support@primbase.net
```

```
Select a connection by number, and Login:
Or enter 'A' to add, 'D' to delete, or 'E' to edit an entry.
Or enter 'T' to move an entry to the top of the list.
```

```
File: ./connect.def
```

Alias	Protocol	Server
0 (exit without connecting)		
1 td2as	Internal/Runtime	
2 TD2REG_WRITE	Internal/Runtime	OpenServer
3 TD2REG_SLAVE	Internal/Runtime	OpenServer

```
Connection...: 2
User.....: teamdrive<Enter>
Password....: *****<Enter>
1: Connected to "TD2REG_WRITE" as "teamdrive".
```

```
For a list of commands enter "#help"
```

```
1: 1> execute file "RegServerSetup.pbt";
1: 2> go
1: Execution begins...
```

```
Registration Server Configuration
```

```
Enter one of the following commands, followed by 'go':
```

```
Initial server setup:
setup:init();
```

```
Updating server settings:
setup:modify();
```

```
Compare XML configuration file with server data:
setup:compare();
```

```
Adding a new provider:
setup:newProvider('<file.xml>');
```

```
Update an existing provider:
setup:updateProvider('<file.xml>');
```

```
Delete a provider:
setup:deleteProvider('<Providercode>');
```

```
1: Execution completed successfully.
```


Take note of these values and submit them along with your provider code and RegServerName to support@teamdrive.net.

If required, the authorization sequence can also be obtained later on from the Registration Server's Administration Console (**Edit Settings -> RegServer -> AuthorizationSequence**).

6.3 Post-install Checks

After the initial installation and configuration, you should perform some checks if the registration server works properly.

6.3.1 Sending Email via PBAC

One of the most important tasks for the registration server is sending registration and notification emails to the TeamDrive client users — it's essential that this works. The Registration Server assumes a functional mail configuration that allows sending messages to arbitrary email addresses. It's possible to use a local or remote MTA, the mail configuration is defined in the `<Email>` section of the `RegServerSetup.xml` setup file.

The details of configuring a local MTA are out of the scope of this document, this heavily depends on your local environment. You need to be aware of topics like "Sender Policy Framework" (SPF), to avoid that emails sent by your Registration Server are considered to be Spam by other mail servers.

Note: The PrimeBase Framework is only capable of sending out email using plain SMTP via TCP port 25 to a local or remote MTA. If your mail server requires some form of authentication or transport layer encryption like SSL/TLS, you need to set up a local mail server that relays all outgoing email from the TeamDrive Registration Server to your MTA using the appropriate protocol and credentials. We recommend configuring a local Postfix instance for this. See the Postfix SMTP client documentation at <http://www.postfix.org/smtp.8.html> for details.

You can perform the following steps to perform a mail sending test from within the PrimeBase Application Environment.

Start the `pbac` tool using connection 1 `td2as` and enter the following command line on the prompt. Please change the email addresses to suit your environment:

```
[root@regserver setup]# pbac
PrimeBase Automation Client.
Copyright 2007-2014, PrimeBase Systems GmbH.
Web:    http://www.primebase.net
E-mail: support@primebase.net
```

```
Select a connection by number, and Login:
Or enter 'A' to add, 'D' to delete, or 'E' to edit an entry.
Or enter 'T' to move an entry to the top of the list.
```

```
File: ./connect.def
```

```
      Alias                Protocol                Server
-----
0  (exit without connecting)
1  td2as                   Internal/Runtime      OpenServer
2  TD2REG_WRITE            Internal/Runtime      OpenServer
3  TD2REG_SLAVE            Internal/Runtime      OpenServer
-----
```

```
Connection..: 1
User.....: <Enter>
1: Connected to "td2as" as "".
```

```
For a list of commands enter "#help"
```

```
1: 1> $sendmail ("from_address@example.com", "to_address@example.com",  
"Subject", "Mailtext");  
1: 2> go  
1: Execution begins...  
1: Execution completed successfully.
```

If you did not get any error message and the email arrives at the account identified by the second parameter, the mail service is configured correctly.

In case of errors, check your mail server's log files and see chapter *Troubleshooting* (page 27) for hints about resolving common issues.

STARTING AND STOPPING THE TEAMDRIVE REGISTRATION SERVER COMPONENTS

You can now start the Apache http Server and the PBAC background processes to make the TeamDrive Registration Server available for TeamDrive clients to connect.

The following services need to be up and running for the Registration Server to function:

- `mysqld` — the MySQL database server (local or on a remote server)
- `httpd` — the Apache http Server
- `php-fpm` — the PHP FastCGI Process Manager
- `teamdrive` — the PBAC based background processes

For testing purposes, you can start these services manually, using the following commands. In the production environment, these services should be started automatically, at boot time.

7.1 Starting services manually

You can use the `service` command to start services manually.

```
[root@regserver ~]# service mysqld start
[root@regserver ~]# service php-fpm start
[root@regserver ~]# service teamdrive start
[root@regserver ~]# service httpd start
```

7.2 Stopping services manually

Similarly, you can use `service` to stop the services manually:

```
[root@regserver ~]# service httpd stop
[root@regserver ~]# service teamdrive stop
[root@regserver ~]# service php-fpm stop
[root@regserver ~]# service mysqld stop
```

7.3 Enabling Service Autostart

Once the TeamDrive Registration Server setup is done, the MySQL server, Apache http Server, PHP FastCGI process manager and `pbac` need to be configured to automatically start at system boot.

Use the command `chkconfig` to enable the automatic start for these processes:

```
[root@regserver ~]# chkconfig --levels 235 httpd on
[root@regserver ~]# chkconfig --levels 235 mysqld on
[root@regserver ~]# chkconfig --levels 235 php-fpm on
[root@regserver ~]# chkconfig --levels 235 teamdrive on
```

7.4 Logging into the Administration Console

Once all services are up and running, you will be able to log into the Admin Console by pointing your browser to the following URL:

<https://regserver.yourdomain.com/adminconsole/>

Enter the login name and password you defined in the `<istributor>` section of `RegServerSetup.xml`.

See the TeamDrive Registration Server Reference and Administration Guides for further details on the configuration and customization of the Registration Server and the TeamDrive Clients connecting to your Server.

TROUBLESHOOTING

8.1 List of relevant log files

In order to debug and analyse problems with the Registration Server configuration, there are several log files that you can consult:

- `/var/log/pbt_mod.trace`: The log file of the `mod_pbt` Apache module. The amount of logging information can be defined by changing the value `debug_trace` in configuration file `/etc/httpd/conf.d/pbt.conf`. The following debug levels can be set: 0: OFF, 1: Errors Only, 2: PBT output, 3: everything. Changing this value requires a restart of the Apache httpd server. The file needs to be owned by the Apache user. Logging only occurs if this file exists and is writable by the Apache user.
- `/var/log/pbac_mailer.log`: The default log file written by the `pbac_mailer` process (managed by `pbctl`). The log file location can be configured by changing the file name after the `-l` option in `/usr/local/primebase/pbstab`. Changing this value requires a restart of the `pbac_mailer` process using `service teamdrive restart`.
- `/var/log/pbvm.log`: The log file for the PrimeBase Application Environment. This log file can be useful to investigate issues related to establishing a MySQL connection or sending out email. The amount of logging can be configured by changing the configuration variable 342 (Protocol Log Level) in `/usr/local/primebase/setup/pbvm.env`, which needs to be modified by using the `pbee` command line tool. Note that the log level should be set to at least 2 in order to obtain meaningful debugging messages. After changing this value, you need to restart PBAC-based services using `service teamdrive restart`.
- `/var/log/httpd/`: The Apache httpd Server's log files (e.g. `error_log`) might also contain additional relevant error messages that should be checked.
- `/var/log/td-adminconsole-api.log`: A log file to track API accesses from the Admin Console. The location of this log file can be configured with the Registration Server setting `RegServer/ApiLogFile` via the Admin Console. The file needs to be owned by the Apache user. Logging only occurs if this file exists and is writable by the Apache user.
- `/var/log/td-adminconsole-failedlogins.log`: A log file to keep track of failed login attempts to the Admin Console. The location of this log file can be configured with the Registration Server setting `LoginSecurity/FailedLoginLog` via the Admin Console.

8.2 Common errors

8.2.1 Invitation emails are not being sent

If users don't receive invitation emails, there are several aspects that should be checked:

- On the Admin Console, check the "Manage Auto Tasks" page: did the task "Send Emails" succeed and was it run recently (check the value of "laststarttime"?). On the "Manage Email Queue", do you see emails with status "Failed"?

- Is the `pbac_mailer` up and running? Check with `pbctl status` and use `pbctl start` to start the process. Also ensure that the PBAC process is configured to be started at system bootup time. See chapter *Starting and stopping the TeamDrive Registration Server components* (page 25) for details.
- Does sending of email work in general? Try using `$sendmail` as described in chapter *Sending Email via PBAC* (page 22) and check `/var/log/pbvm.log` and your MTA logs for delivery status notifications.
- Check the `/var/log/pbac_mailer.log` log file for errors.

8.2.2 PBAC: Errors sending email with `$sendmail`

If you get an error message like:

```
Error (501) sending mail: 501 Syntactically invalid HELO argument(s)
```

Try putting your hostname in the file `/etc/hosts`.

In case you get an error like:

```
01/17/2014 06:07:39
1: ERROR: -16045 (-12996) : "$sendmail("from_address@exam ..."@client line 1:
Error (-12996) sending mail: Bad mail ID.
```

Check the log file `/var/log/pbvm.log` for details.

8.2.3 Admin console: Error connecting to the MySQL server

If you get an error like:

```
Error connecting to the MySQL server: MDB2 Error: connect failed
```

Verify that the MySQL connection parameters like username and password are set up correctly. See chapter *Administration Console MySQL Configuration* (page 16) for details.

8.2.4 Admin console: API error code: -30000, message: Access denied

If some operations on the web-based Administration Console (e.g. changing a configuration option) result in an error message `API error code: -30000, message: Access denied`, the IP address of the admin console host is likely not on the white list of IPs that are allowed to perform API calls. Check the content of the Registration Server setting `API_IP_ACCESS` (“Edit Provider Settings” -> “API” -> “API_IP_ACCESS”) and make sure that the external IP address of the server running the Administration Console is included in the list. If necessary, add the missing address in a new line and click **Save**.

8.2.5 Invalid/insufficient connection options (TCP/IP communications error)

If some operations on the web-based admin console (e.g. changing provider settings or any other changes that perform API calls to the Registration Server) result in an error as the following one:

```
The following error occurred in '"OPEN TD2REG_WRITE DBMS USER ..."@network
line 1: Invalid/insufficient connection options (TCP/IP communications
error) : Opening and initializing PBI connection, Alias "td2as"' while
processing your request: -12986 (-12948).
```

The Apache error log on the Registration Server `/var/log/httpd/error_log` shows a similar error:

```
[notice] Mod_pbt Error: pid: 8181, where: "OPEN TD2REG_WRITE DBMS USER
..."@network line 1: Invalid/insufficient connection options (TCP/IP
communications error) : Opening and initializing PBI connection, Alias
"td2as", perr: -12986, serr: -12948
```


Check that the MySQL connection definitions in file `/usr/local/primebase/setup/connect.def` are set up correctly and that the ownerships and permissions of this file allow the Apache http Server to open this file for reading (as the `mod_php` Apache module needs to obtain the MySQL connection information from there).

8.2.6 Email messages sent by the registration server show encoding issues

Invitation emails and other notifications sent out by the Registration Server are encoded as UTF-8. Before they are sent out, they are first inserted into the MySQL database before the `pbac_mailer` task delivers them to the configured MTA. If you notice encoding issues (special chars or umlauts not displayed properly), check the following:

- Double check that your templates are UTF-8 encoded. The default templates shipped with the TeamDrive Registration Server use the correct encoding, but if you're updating from previous versions, the encoding might be off.
- Check the MySQL connection definition file `/usr/local/primebase/setup/connect.def` for the existence of `Charset=utf8` in the `xoHost=` section, e.g.:

```
TD2REG_WRITE:mem:\xoHost=regdb.local;Charset=utf8;Reconnect=\tCustom...
```


RELEASE NOTES - VERSION 3.0.018

TeamDrive Registration Server version 3.0.018 is the next major release following after version 3.0.017.

Version 3.0.018 contains the following features and notable differences compared to version 3.0.017:

- As a security enhancement, TeamDrive user passwords stored on the Registration Server are now hashed using the bcrypt algorithm instead of the previously used salted MD5 method. When logging in with a TeamDrive Client version 3.2.0 (Build: 536) or newer, existing hashed passwords are automatically converted into the new format.
- Changing, invalidating or resetting a user's password now also triggers sending an email to the affected user. For this purpose, the following new mail templates were added: `passwd-changed`, `passwd-invalidated` and `passwd-reset`.
- The Registration Server now supports sharing and synchronizing user profile information across all of the user's devices and with other users, e.g. initials, registration email, profile picture, full name, phone (telephone number), mobile (telephone number). Before, this information was shared with other users on a per-Space basis. Only users that share Spaces are able to exchange profile data with this new method. This feature will be supported by a future TeamDrive Client version.
- The expiry date of licenses is now properly checked via the "Expire Licenses" auto task. Users receive an advance notification 10 and 3 days before the license expires. When the date provided in the **Valid until** field has been reached, the user receives a final notification and his license will be reverted to the default free license. The following email templates were added to facilitate the notification: `license-expirein10days`, `license-expirein3days` and `license-expired-en`. To avoid disruptions/surprises when upgrading from previous Registration Server versions, the update function `setLicenseExpiryDefault()` will set the default value of `ENABLE_LICENSE_EXPIRY` to `False` for providers that already have licenses with an expiry date. When performing a new installation or adding a new provider account, license expiration will be enabled by default.
- Email templates now support the `[[BRAND]]` macro, to replace the term "TeamDrive" with another string if required. This can be defined via the `EMAIL/BRAND_NAME` provider setting. The default is `TeamDrive`.
- Most parts of the TeamDrive Registration Server installation can now be performed via RPM on Red Hat Enterprise Linux 6 and derivative distributions, which significantly improves the installation procedure and the process of applying updates. In particular, the following components are now provided in the form of RPM packages:
 - The PBT-based Registration Server (`td-regserver-3.0.018.2-0.el6.noarch.rpm`, files installed in `/usr/local/primebase/setup/scripts`)
 - The PHP-based Administration Console and support files (`td-regserver-adminconsole-3.0.018.2-0.el6.noarch.rpm`, files installed in `/var/www/html/adminconsole` and `/var/www/html/tdlibs`)
 - The PrimeBase Application Environment (`PrimeBase_TD-4.5.48.<build>-0.el6.x86_64.rpm` installed in `/usr/local/primebase`), including the PrimeBase Apache module `mod_pbt` (installed in `/usr/lib64/httpd/modules/mod_pbt.so`) and some support scripts and configuration files in `/etc/`.

- The installation package now contains a script `mysql_install.sh` that performs the creation of the required `teamdrive` MySQL user account and populating the databases required for the Registration Server.
- The installation package now contains a log rotation script, to support rotation and compression of the Registration Server's log files.
- The installation now uses the default MySQL data directory location (`/var/lib/mysql`) instead of defining a custom one (`/regdb`). The default MySQL configuration settings for `my.cnf` have been reviewed and adjusted.
- The automatic service startup at bootup time is now configured using the distribution's `chkconfig` utility instead of changing the `Boot` options in file `/usr/local/primebase/pbstab`. The `PrimeBase_TD` RPM package provides the required SysV init script `/etc/init.d/teamdrive` to facilitate this.
- The term "Distributor" has been replaced with "Provider" in most occasions.
- The obsolete settings `UseExternalAuthentication` and `UseExternalAuthenticationCall` have been removed. External authentication is now enabled by setting `AUTHSERVICE/USE_AUTH_SERVICE` to `True`.
- In previous versions, the setting `AUTH_VERIFY_PWD_FREQ` did not have any effect (it was added without the actual implementation by accident). Starting with version 3.0.018, a user's Clients will be logged out from the TeamDrive Service after the time defined in this setting. To avoid surprises and a change in behaviour after an upgrade, updating from a previous version of the Registration Server suggests calling the update function `setLoginFreqToZero()`; to change this setting to 0 for any existing Provider.

The PHP-based Administration Console received several new features, numerous usability enhancements and security improvements. Some notable highlights include:

- Tabular output (e.g. a filtered list of users, devices or licenses) can now be exported to CSV files.
- Tabular output now indicates the current sort order and column name with a small arrow icon.
- The columns visible in the table displayed on the **Manage Users** and **Manage Licences** pages are now configurable.
- The summary display of a user's licenses ("Licenses owned" and "Licenses used") on the **Manage Users** page has been simplified.
- The list of Spaces in a user's Depot is now displayed as a sortable table.
- It's now possible to wipe or delete multiple devices of a user at once.
- The Registration Server's Authorization Sequence (required for exchanging invitations with users on other Registration Servers via TDNS) can now be obtained from the Administration Console via **Edit Settings -> RegServer -> AuthorizationSequence**.
- After successful registration, a Host Server's activation key is now displayed on the **Manage Servers** page, to simplify the registration process for new Host Servers.
- It is now possible to remove registered Host Servers via the **Manage Servers** page.
- The Admin Console now supports viewing a selection of server log files directly in the web browser instead of requiring logging in on the server's console. The **View Server Logs** page is only visible for the Registration Server's default provider and any user having the `HAS_VIEW_SERVER_LOGS_RIGHTS` privilege. The list of log files is defined in the (read-only) Reg Server setting `ServerLogFiles` and can only be modified by updating the setting in the database directly. Log files can only be viewed if the user that the Apache http Server is running under (usually `apache`) has the required access privileges to view these files.
- Most of the Admin Console Settings are now stored in table `TD2Setting` of the MySQL database instead of the configuration file `tdlibs/globals.php` and can be configured via the Admin Console instead:
 - `RegServer/ApiLogFile` (default: `/var/log/td-adminconsole-api.log`)
 - `RegServer/ServerTimeZone` (default: `Europe/Berlin`)
 - `LoginSecurity/LoginSessionTimeout` (default: 30)

- LoginSecurity/FailedLoginLog (default: /var/log/td-adminconsole-failedlogins.log)
- LoginSecurity/LoginMaxAttempts (default: 5)
- LoginSecurity/LoginMaxInterval (default: 60)

The only information required in `globals.php` is the MySQL connection string to access the Registration Server's MySQL database. Alternatively, these credentials can be provided from a separate MySQL configuration file. See chapter *Administration Console MySQL Configuration* (page 16) for details.

- Disabling a user does no longer provide the **apply to devices** option, as it's sufficient to disable the user account to block access to the TeamDrive service.
- Changing the Provider setting `AUTHSERVICE/USE_AUTH_SERVICE` to `True` now automatically adds the other required settings like `AUTH_LOGIN_URL` and `VERIFY_AUTH_TOKEN_URL`.
- The provider filter selection list now also prints the company name after the 4-letter code.
- An option was added to assign an existing license to a user when editing the user's details.
- Various settings that used to expect values in bytes only now provide an option to select other units like "MB" or "GB".
- Input fields that expect a date now provide a date picker, to simplify the entering of dates.
- Filter options by date now provide a more intuitive way to define "before", "at" or "after" the entered date.

9.1 Change Log - Version 3.0.018

Table 9.1: Change Log - Version 3.0.018

Build Date	Version	Comment
YYYY-MM-DD	3.0.018.3	
2014-11-12	3.0.018.2	<ul style="list-style-type: none"> • Fixed bug in propagating email address changes to other devices belonging to a user • Fixed bug in deleting a user's privileges when deleting the user's account (REGSERVER-734) • Fixed issue with store forward messages that were not forwarded to a user upon registration (REGSERVER-759) • Admin Console: Fixed encoding issue when adding users with usernames containing UTF-8 characters (REGSERVER-756) • Admin Console: Fixed minor bug in the "Add new provider settings" menu (REGSERVER-747) • RegServerSetup.xml: Fixed missing closing bracket in the <code>APIChecksumSalt</code> tag. • API: fixed <code>addXMLDepot</code> call that returned invalid URLs when the setting <code>SIMULATE_REGSERVER_20</code> was enabled. (REGSERVER-741)
2014-11-05	3.0.018.1	<ul style="list-style-type: none"> • Initial public release

RELEASE NOTES - VERSION 3.0.017

Table 10.1: Release Notes - Version 3.0.017

Build Date	Version	Comment
2014-09-02	30017.13	<ul style="list-style-type: none"> • Admin Console: show extreference in the license Administration screen • Security improvement: fixed OS permissions/ownerships of some configuration files and log files containing plaintext passwords (REGSERVER-599) • Admin Console: Security improvement: Don't display the Console version on the login page (REGSERVER-558) • Virtual Appliance: set <code>ServerTokens</code> to <code>Prod</code> and <code>ServerSignature</code> to <code>Off</code> in <code>httpd.conf</code>, to disable displaying the Apache Server version and OS version in the HTTP headers and on error pages (REGSERVER-608) • Added missing tag <code><APISendEmail></code> in <code>DIST.xml</code> template file • Security improvement: disabled unneeded HTTP methods in <code>pbt.conf</code> (only allow GET, POST, disable PUT, HEAD, OPTIONS, TRACE) (REGSERVER-613) • API: added new API call <code>removedepotfromuser</code> extended <code>setdepotforuser</code>. Fixed bug in <code>setreference</code> and removed deprecated <code>location-Support</code> in <code>getHostForDistributor</code>. Fixed error handling in <code>setinviteduser</code>. Updated API-Version number to "1.0.005". • For monitoring purposes, calling the Reg Server's ping URL with the optional parameter <code>tdns=\$true` `(e.g. <code>http://regserver.yourdomain.com/pbas/td2as/reg/ping.xml?tdns=\$true` `)</code> now also performs a TDNS lookup, to verify that the communication between the Reg Server and TDNS is working properly.</code>
Continued on next page		

Table 10.1 – continued from previous page

Build Date	Version	Comment
2014-07-09	30017.12	<ul style="list-style-type: none"> • Updated to requiring PrimeBase 4.5.48, updated pbstab and documentation accordingly. This version of PrimeBase now installs a shell profile file by default and provides a proper SysV init script that can be used to enable/disable the pbac_mailer background task. • Admin Console: Fixed wrong escaping of HTML characters in the device messages popup (REGSERVER-575) • Admin Console: changed session timeout from 10m to 30m • Admin Console: Added more fields to license editing page • RegServerSetup.pbt now sets APIAllowSettingDistributor to true if another distributor is added (REGSERVER-579) • Added missing globalDepotID to default depots for clients with two accounts on the same server(s). (REGSERVER-583) (this fix also requires an updated Host Server having the fix from HOSTSERVER-326)
2014-06-26	30017.11	<ul style="list-style-type: none"> • Admin Console: “Create Depot” now accepts storage limits in other units than bytes. Unified the UI with regards to selecting a Depot owner and selecting Users to invite (REGSERVER-574)
2014-06-17	30017.10	<ul style="list-style-type: none"> • Admin Console: Added confirmation checkbox for deleting a user’s license when deleting the user (REGSERVER-554) • Admin Console: Improved listing of licenses to no longer show one entry per Device for the same license (REGSERVER-565) • Admin Console: Replaced “parcel” with “key repository”, replaced “Packet” with “Package” in the License creation/editing dialogues (REGSERVER-567) • Admin Console: Added exporting tables as CSV function. • Fixed missing LOG_UPLOADS setting in upload.php log upload script (REGSERVER-559) • Added Proxy support in upgradeDefaultDepot • Major documentation rewrite: added general reference and API documentation, converted all documents to reStructured-Text/Sphinx • RegServerSetup.xml: Fixed incorrect closing tag (</ProviderInfoURL> -> </DownloadURL>)
2014-04-17	30017.9	<ul style="list-style-type: none"> • Removed misleading error output in csvimportregserver.php • Fixed default license key error using the API (REGSERVER-526) • Improved description for StoreRegistrationDeviceIPinSeconds (REGSERVER-532) • Admin Console: bugfix for editUser.php: wrong user got displayed when changing depot limits. • Admin Console: editUser.php didn’t display “extauthid” in all cases (REGSERVER-537) • Admin Console: Display activation code in device-list entry for deactivated tdhosting “users”
Continued on next page		

Table 10.1 – continued from previous page

Build Date	Version	Comment
2014-03-27	30017.8	<ul style="list-style-type: none"> • Admin Console: server/distributor settings can now be empty strings (REGSERVER-476) • Admin Console: displays a warning if LOGIN_IP is not set • REGSERVER-464: RegServerSetup.pbt now prints the Authentication Sequence during initial install • REGSERVER-494: Sending notification to users located on different Reg-Server returned “remote authorization not allowed” • Improved error handling in case of empty hosting_url or hosting_name • REGSERVER-507: Don’t create user accounts in plreg.sql • RegServerSetup.pbt: Improved screen output for readability and clarity • RegServerSetup.xml: Default for <TDNSEnabled> must be \$true to avoid errors for a default setup • OWNERMETA_CSV_IMPORT_ACTIVE should not add OWNERMETA_CSV_UPLOAD_DIR, OWNERMETA_CSV_ERROR_DIR and OWNERMETA_CSV_SUCCESS_DIR, because we support import using the database or a hot folder. Default is using the database and therefore the Dir-Settings are not required. • Packaging: Updated and added DIST.xml to the distribution • Fixed link in bannerAdmin.php • Removed duplicate code in RegServerSetup.pbt
2014-03-14	30017.7	<ul style="list-style-type: none"> • Fixed nasty typo in RegServerSetup.xml
2014-03-14	30017.6	<ul style="list-style-type: none"> • REGSERVER-478: Deleting TD2FreeUserStorage and TD2Parcel in case of deleting a user • reg_init.pbt: Now only use the curl-based code to verify external logins (both via http and https) • External auth: Updated LDAP ext auth example: implement function base64url to encode the token, to avoid “+” and “/” being included in the token string. • REGSERVER-471: Admin Console XSS security fixes related to TD2User • External auth: fixed REGSERVER-443 (Sample login page defaults to “Password lost”, not “Login”), changed error messages to show the same error regardless if user name or password are wrong. • Admin Console: moved failed-logins log file to /var/log/td-adminconsole-failedlogins.log. NOTE: this log file must now be created during installation
2014-02-25	30017.5	<ul style="list-style-type: none"> • Updated pbstab version number from 4546 to 4547 • Added deleteDistributor to RegServerSetup.pbt • Executing HTTPRequest.pbt in RegServerSetup.pbt requires no location • RegServerSetup.pbt: Generate a mysql update script if changes are required to the database structure • Handle the case that the TD2Setting.Format column does not exist, when creating system variables

Continued on next page

Table 10.1 – continued from previous page

Build Date	Version	Comment
2014-02-07	30017.4	<ul style="list-style-type: none"> • REGSERVER-426: Admin Console: changed API log file location to /var/log/td-adminconsole-api.log • Admin Console: added option to edit a depots transfer limit • REGSERVER-428: Removed duplicate entry <UserEmailUnique> from section <RegServer> in RegServerSetup.xml and RegServerSetup.pbt • Admin Console: improved test to check if the setDepot function is available on a host server • Install upload.php into logupload/upload.php instead the document root • Admin: user simply gets a warning when trying to call setdepot on a host server that does not support it • pbt.conf: Reduced mod_pbt log level from 2 (PBT_TRACE) to 1 (ERROR_TRACE) to reduce default log noise in /tmp/pbt_mod.trace • Admin: fixed regex that prevented changing the LogUploadURL setting • REGSERVER-432: API call upgradelicense no longer throws an error if feature is empty • Admin Console: the API log now correctly shows entries that don't have usernames • REGSERVER-436: Setting HAS_DEFAULT_DEPOT to true, creates all missing hosting system parameters
2014-02-04	30017.3	<ul style="list-style-type: none"> • Bug fixes: REGSERVER-424, double <teamdrive> tag removed, fixed invitations when a user was registered with same e-mail on 2 other Reg Servers, Added Download-URL for invitation mail templates
2014-01-30	30017.2	<ul style="list-style-type: none"> • Renamed out.log to api.log • Fixed RegEx for API_IP_ACCESS • Admin Console: Changed default mysql username to teamdrive • Updated pbvm.env to write the log file into /var/log/pbvm.log (REGSERVER-423) • REGSERVER-422: changed the default log file location in pbstab for the pbac_mailer from /tmp/mail.log to /var/log/pbac_mailer.log • Removed setup/pbas.env from the installation package
2014-01-23	30017.1	<ul style="list-style-type: none"> • First build using the scripted build, updated RegServerSetup.pbt and included some Admin Console fixes
2013-10-23	30017	<ul style="list-style-type: none"> • Not final; Bcrypt is still missing

11.1 Abbreviations

PBAC Prime Base Automation Client

PBAS Prime Base Application Server

PBEE Prime Base Environment Editor

PBCON Prime Base Console

PBT Prime Base Talk

SAKH Server Access Key HTTP for TeamDrive 2.0 Clients

TDES Team Drive Enterprise Server

TDNS Team Drive Name Service

TDPS TeamDrive Personal Server

TDRS Team Drive Registration Server

TDSV Same as **SAKH**, but for TeamDrive 3.0 Clients: Team Drive Server