TeamDrive Registration Server Installation and Configuration

Release 4.6.2.0

Paul McCullagh, Eckhard Pruehs

2022

COPYRIGHT NOTICE

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

This manual will guide you through the installation of your own local TeamDrive Registration Server. This document is intended for system administrators who need to install and configure a TeamDrive Registration Server.

This Installation Guide outlines the deployment of a single node installation, where all required components are located on the same OS instance. Please consult the *TeamDrive Registration Server Administration Guide* for recommendations about scalability and/or high availability.

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, enabling them at system bootup time
    - Creating Linux users
    - Assigning file ownerships and privileges
    - Creating and mounting file systems
    - Setting up environment variables

- Apache HTTP 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

- MTA configuration: installing and configuring a local MTA like the Postfix mail server

- Basic knowledge of application server technology

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.
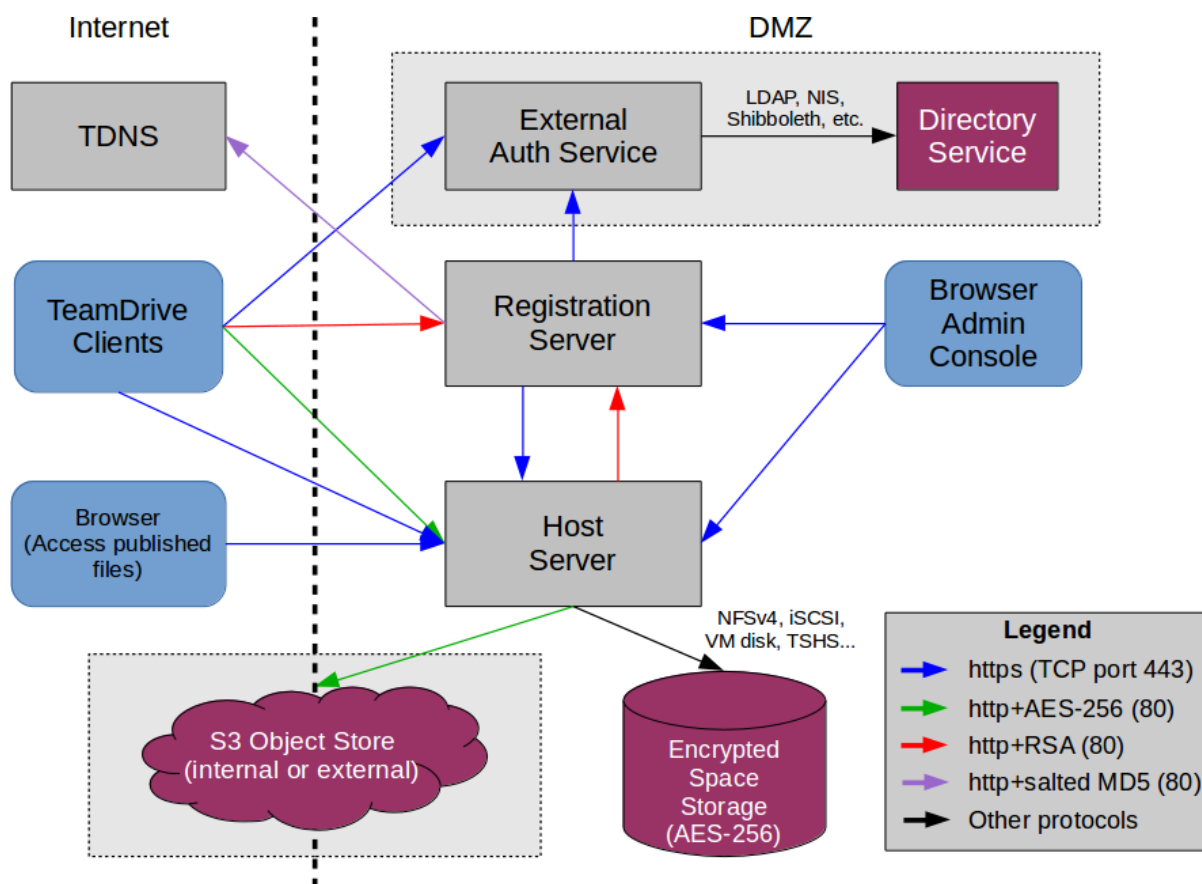
Fig. 3.1: TeamDrive Enterprise Server Networking Overview

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.

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/ and its Master Registration Server (http://reg.teamdrive.net by default), either directly or via an existing HTTP proxy server.

For more details about TDNS, see chapter teamdrive name server (tdns).

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 users, the Registration Server needs to be able to establish outgoing HTTP or HTTPS connections to the host providing the external Authentication Service.

3.1.3 Hardware Requirements

Operating a TeamDrive Registration Server requires an Intel/AMD-based server system, which should have at least a dual-core x86-64 CPU (quad-core or more is recommended), with a minimum of 2 GB of RAM. This could be a physical or a virtual instance.

3.1.4 Operating System

The TeamDrive Registration Server is based on TeamDrive-specific services (the Yvva Runtime Environment) and the "LAMP-Stack" (Linux/Apache/MySQL/PHP).

We recommend an up to date 64-bit version of **Red Hat Enterprise Linux 7** (RHEL 7) or a derivative distribution like **CentOS 7**, **Oracle Linux 7** or **Scientific Linux 7**. Alternatively, **Amazon Linux** may also be used.

The following Linux operating system components are required:

- Apache HTTP Server version 2.4 (CentOS 7)
- MySQL Community Server 5.6
- PHP 7.2 / 7.3
- A working MTA configuration (e.g. a local Sendmail or Postfix instance that relays outgoing messags to a remote MTA)

We suggest starting with a minimal OS installation, adding the required components using the `yum` package manager afterwards.

The Registration Server installation packages have been developed and tested with this OS environment in mind — the names of packages, configuration files and path names might be different on other Linux distributions. If you have any questions about using other Linux distributions, please contact sales@teamdrive.net.

3.1.5 TeamDrive Server Components

The following TeamDrive-specific components will be installed:

- Yvva Runtime Environment 1.5.9 (or newer)
- TeamDrive Registration Server and the PHP-based Administration Console Version 4.6 (or newer)

OPERATING SYSTEM INSTALLATION AND CONFIGURATION

4.1 Base Operating System Installation

Perform a minimal OS installation of a recent RHEL6/7 or derivative Linux distribution, using your preferred installation method (manual install, Kickstart, etc). The details of how to perform this task are out of the scope of this document.

The system should have IP connectivity, using a fixed IP address and a resolvable fully qualified domain name. For performing the installation, the system needs to be able to establish outgoing TCP connections (mainly to download additional components).

Additionally, a local or remote MTA (e.g. Postfix or Sendmail) needs to be installed and configured so the system is capable of sending email.

Boot up the system and log in as the root user.

4.2 Enable Time Synchronization With NTP

We strongly advise that the clocks of all servers in a TeamDrive installation are synchronized using the Network Time Protocol (NTP). This can be achieved by installing the ntp package and enabling the NTP daemon:

```
[root@regserver install]# yum install ntp
[root@regserver install]# service ntpd start
[root@regserver install]# chkconfig ntpd on
```

Edit and update the configuration file /etc/ntp.conf, if necessary for your local environment.

4.3 Disable SELinux

The TeamDrive Registration Server currently can not be run when SELinux is enabled. Edit the file /etc/selinux/config and set SELINUX=disabled.

Reboot the system or change the SELinux enforcing mode at run time using the following command:

```
[root@regserver install]# setenforce 0
```

4.4 Firewall Configuration

You should configure a local firewall so the server is protected against remote attacks. The only TCP ports that must be reachable from the Internet are 80 (http) and 443 (https). Optionally, port 22 (SSH) can be opened to facilitate remote administration, but access to this port should be restricted to known and trusted IP addresses or networks only.

On a minimal installation, you can install and use the text-based firewall configuration utility to enable access to the following services:

- SSH

- Secure WWW (HTTPS)

- WWW (HTTP)

To configure the firewall, you need to run the following commands:

```
[root@regserver install]# yum install system-config-firewall-tui \
newt-python
[root@regserver install]# system-config-firewall-tui
```

Follow the instructions to configure the firewall. Enable additional protections based on your local requirements or security policies.

You can check the result with iptables -L:

```
[root@regserver ~]# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere             state
→RELATED,ESTABLISHED
ACCEPT     icmp --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:ssh
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:http
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:
→https
REJECT     all  --  anywhere             anywhere             reject-with icmp-host-
→prohibited

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
REJECT     all  --  anywhere             anywhere             reject-with icmp-host-
→prohibited

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

In case of using an external company firewall enable the above ports for the incoming traffic. For outgoing communication please enable:

- Secure WWW (Port 443 for HTTPS)

- WWW (Port 80 for HTTP)

- SMTP (Port 25 for sending mails using a public mail server; in case of using SSL-communication to the mail server, also Ports 465, 587)

- DNS Lookup (Port 53 for DNS communication with a public DNS server)

4.5 CentOS Hardening

Install a rootkit scanner:

```
yum install epel-release
yum --enablerepo=epel -y install rkhunter
```

and update the email in MAILTO in the rkhunter config file:

```
/etc/sysconfig/rkhunter
```

The rootkit scanner will automatically be started once a day and will send a status email to the above email.

Update rkhunter and run the rootkit check:

```
rkhunter --update
rkhunter --propupd
rkhunter --check --sk
```

Install USBGuard to protect your systems against intrusive USB devices:

```
yum install usbguard
```

Change config for `PresentControllerPolicy` in `/etc/usbguard/usbguard-daemon.conf` and set this to:

```
PresentControllerPolicy=block
```

Enable usbguard:

```
systemctl enable --now usbguard
```

We recommend to harden the CentOS system based on the DevSec Hardening Framework:

```
https://github.com/dev-sec/linux-baseline
```

and the Lynis - Security auditing tool for Linux:

```
https://cisofy.com/lynis/
```

Install both tools:

```
curl https://omnitruck.chef.io/install.sh | sudo bash -s -- -P inspec
yum install epel-release
yum install lynis
```

To analyze the system using DevSec execute:

```
inspec exec https://github.com/dev-sec/linux-baseline
```

and for Lynis execute:

```
lynis audit system
```

The DevSec report will show the test results with a green and red staus. Lynis generates a similar test result with a green, yellow and red status and calculates a Hardening index.

For hardening the system we offer a script which will be installed with the TeamDrive Registration Server software. The script will be executed later on in the documentation.

4.6 Installing MySQL Server

The TeamDrive Registration Server requires a MySQL database to store its information. This document assumes that the MySQL instance runs on the same host as the Registration Server itself, connecting to it via the local socket file.

Alternatively, it's possible to use an external MySQL Server. In this case, you need to make sure that this external MySQL instance is reachable via TCP from the Registration Server (usually via TCP port 3306) and that the

`teamdrive` MySQL user is defined correctly (e.g. the MySQL username in the remote database would become `teamdrive@regserver.yourdomain.com` instead of `teamdrive@localhost`).

Most MySQL installations usually do not allow the `root` user to log in from a remote host. In this case the installation script is unable to create the dedicated `teamdrive` user automatically and you need to perform this step manually before performing the installation of the TeamDrive Registration Server databases.

Especially the correct definition of the host part is critical, as MySQL considers `username@regserver` and `username@regserver.yourdomain.com` as two different users.

Note: Since CentOS 7 MySQL is no longer in CentOS's repositories and MariaDB has become the default database system offered. We recommend installing the mysql community server instead. If you are installing on CentOS 7 then perform the following steps by downloading the MySQL 8.0 repository (this is note necessary on CentOS 8, because MySQL is in CentOS's repository):

```
yum localinstall https://dev.mysql.com/get/mysql80-community-release-el7-1.noarch.
↪rpm
yum install mysql-community-server
```

On CentOS 8 install the MySQL Client and Server packages from the default repository:

```
dnf install mysql mysql-server
```

For reliablility and performance reasons, we recommend placing the MySQL data directory `/var/lib/mysql` on a dedicated file system or storage volume.

The default maximum file handle limit in CentOS 7 is 1024 which might be to less for the amount of file handles and database connections for the TeamDrive apache module. The amount of file handles can be calculated: tables (currently 26) x 2 (2 files per table) x apache processes x 2 (for restarting the apache). For less than 100 users it will be 26 x 2 x 20 x 2 = 2080 file handles, for 500 users 26 x 2 x 50 x 2 = 2200 file handles and for more than 1000 users 26 x 150 x 2 = 15600

To be save, we increase the value to 65535 in the following three configuration files.

First: Edit `/etc/sysctl.conf` and add the below line, save and exit:

```
fs.file-max = 65535
```

Second: Increase the hard and soft limits in `/etc/security/limits.conf`. Add the below lines before the `#End`, save and exit:

```
* soft nproc 65535
* hard nproc 65535
* soft nofile 65535
* hard nofile 65535
```

After the change execute:

```
sysctl -p
```

Third: For MySQL, edit `/usr/lib/systemd/system/mysqld.service` and add the 2 lines below at the end:

```
LimitNOFILE=65535
LimitNPROC=65535
```

After the change execute:

```
systemctl daemon-reload
```

Please start the MySQL server now and tell systemd to start the service automatically at boot:

```
[root@regserver ~ ]# systemctl start mysqld.service
[root@regserver ~ ]# systemctl enable mysqld.service
```

Run the secure installation script and follow the recommendations. The latest version of mysql generates a temporary password which is written to the log. Use the password to run `mysql_secure_installation` and set a new password for the MySQL `root` user and take note of it:

```
[root@regserver ~ ]# grep 'temporary password' /var/log/mysqld.log
2021-07-22T17:47:13.315642Z 6 [Note] [MY-010454] [Server] A temporary password is
→generated for root@localhost: 4>x...k4s
[root@regserver ~ ]# mysql_secure_installation

Securing the MySQL server deployment.

Enter password for user root:

The existing password for the user account root has expired. Please set
a new password.

...
```

Answer all question with Y for:

- Remove anonymous users?
- Disallow root login remotely?
- Remove test database and access to it?
- Reload privilege tables now?

MySQL is now up and running. It will be populated with the required databases and tables during the Registration Server installation process.

4.7 Apache Setup and Configuration

The TeamDrive Clients use the HTTP protocol to communicate with the Registration Server. The Registration Server's Administraion Console is based on the PHP scripting language; both are served by the Apache HTTP server.

Install the Apache HTTP Server and the `mod_ssl` Apache module by running the following command:

```
yum install httpd mod_ssl
```

For security reasons, we also advise to disable the so-called "Server Signature" - a feature that adds a line containing the server version and virtual host name to server-generated pages (e.g. internal error documents, directory listings, etc). Change the configuration in `/etc/httpd/conf/httpd.conf` as follows:

```
ServerSignature Off
```

By default, the server version and operating system is also displayed in the `Server` response header field, e.g. `Server: Apache/2.4.6 (CentOS)`. To suppress this output, we suggest to update the `ServerTokens` option as follows:

```
ServerTokens Prod
```

The TeamDrive Registration Server only requires a few Apache modules to be enabled. To reduce the memory footprint, please deactivate unnecessary modules in the apache configuration.

4.8 Enable "Prefork" Mode (CentOS 8)

The `mod_yvva` module requires that apache run in prefork mode. Note that Apache will crash when running in a different mode.

To set the mode, execute:

```
sed -e '/LoadModule mpm_event_module/ s/^#*/#/' -i /etc/httpd/conf.modules.d/00-
↪mpm.conf
sed -e '/#LoadModule mpm_prefork_module/ s/^#*//' -i /etc/httpd/conf.modules.d/00-
↪mpm.conf
```

which will comment out the mpm_event_module and uncomment the mpm_prefork_module. The result should look:

```
# Select the MPM module which should be used by uncommenting exactly
# one of the following LoadModule lines.  See the httpd.conf(5) man
# page for more information on changing the MPM.
...
LoadModule mpm_prefork_module modules/mod_mpm_prefork.so
...
#LoadModule mpm_worker_module modules/mod_mpm_worker.so
...
#LoadModule mpm_event_module modules/mod_mpm_event.so
```

4.8.1 Apache 2.4 (CentOS 7 / 8)

In the directory: `/etc/httpd/conf.modules.d` comment out all modules in the following config files. Using the linux stream editor (sed) with the following regular expression will add a '#' comment sign in each line starting with 'LoadModule':

```
sed -e '/LoadModule/ s/^#*/#/' -i /etc/httpd/conf.modules.d/00-dav.conf
sed -e '/LoadModule/ s/^#*/#/' -i /etc/httpd/conf.modules.d/00-lua.conf
sed -e '/LoadModule/ s/^#*/#/' -i /etc/httpd/conf.modules.d/00-proxy.conf
sed -e '/LoadModule/ s/^#*/#/' -i /etc/httpd/conf.modules.d/01-cgi.conf
sed -e '/LoadModule/ s/^#*/#/' -i /etc/httpd/conf.modules.d/10-proxy_h2.conf
```

Disable all modules in `/etc/httpd/conf.modules.d/00-base.conf` and re-enable only the required modules:

```
sed -e '/LoadModule/ s/^#*/#/' -i /etc/httpd/conf.modules.d/00-base.conf
sed -e '/#LoadModule access_compat_module/ s/^#*//' -i /etc/httpd/conf.modules.d/
↪00-base.conf
sed -e '/#LoadModule actions_module/ s/^#*//' -i /etc/httpd/conf.modules.d/00-base.
↪conf
sed -e '/#LoadModule alias_module/ s/^#*//' -i /etc/httpd/conf.modules.d/00-base.
↪conf
sed -e '/#LoadModule authz_core_module/ s/^#*//' -i /etc/httpd/conf.modules.d/00-
↪base.conf
sed -e '/#LoadModule autoindex_module/ s/^#*//' -i /etc/httpd/conf.modules.d/00-
↪base.conf
sed -e '/#LoadModule dir_module/ s/^#*//' -i /etc/httpd/conf.modules.d/00-base.conf
sed -e '/#LoadModule headers_module/ s/^#*//' -i /etc/httpd/conf.modules.d/00-base.
↪conf
sed -e '/#LoadModule log_config_module/ s/^#*//' -i /etc/httpd/conf.modules.d/00-
↪base.conf
sed -e '/#LoadModule mime_module/ s/^#*//' -i /etc/httpd/conf.modules.d/00-base.
↪conf
sed -e '/#LoadModule rewrite_module/ s/^#*//' -i /etc/httpd/conf.modules.d/00-base.
↪conf
sed -e '/#LoadModule setenvif_module/ s/^#*//' -i /etc/httpd/conf.modules.d/00-
↪base.conf
```

```
sed -e '/#LoadModule slotmem_shm_module/ s/^#*//' -i /etc/httpd/conf.modules.d/00-
↪base.conf
sed -e '/#LoadModule socache_shmcb_module/ s/^#*//' -i /etc/httpd/conf.modules.d/
↪00-base.conf
sed -e '/#LoadModule unixd_module/ s/^#*//' -i /etc/httpd/conf.modules.d/00-base.
↪conf
sed -e '/#LoadModule version_module/ s/^#*//' -i /etc/httpd/conf.modules.d/00-base.
↪conf
```

4.9 Configure `mod_ssl`

In order to facilitate access to the Registration Server's API and initial setup screens via SSL, the following needs to be added to the end of the default `<VirtualHost>` section in `/etc/httpd/conf.d/ssl.conf`:

```
Include conf.d/td-regserver.httpd.conf.ssl
</VirtualHost>
```

Note: The Apache HTTP Server package includes a self-signed SSL certificate for testing purposes. If you connect to the server using a web browser, it will likely raise an error about an untrusted/insecure connection. You should consider replacing this certificate with an appropriate one.

Follow the instructions provided by your certificate authority on how to obtain and install an SSL certificate for the Apache HTTP Server.

Verify your SSL configuration using the service from SSL Labs: `https://www.ssllabs.com/ssltest/analyze.html` and make sure that the "Handshake Simulation" is working for current platforms and browser. The following ssl parameters on CentOS 7 for the apache web server will create an A-rating and make sure that the handshake is working for current platforms and browser (for CentOS 8 no changes on the default configurations are necessary):

```
SSLProtocol all -SSLv2 -SSLv3

SSLHonorCipherOrder on

SSLCipherSuite ECDH+AESGCM:DH+AESGCM:ECDH+AES256:DH+AES256:ECDH+AES128:DH+AES:
↪ECDH+3DES:DH+3DES:RSA+AESGCM:RSA+AES:RSA+3DES:!aNULL:!MD5:!DSS
```

4.10 PHP and PEAR framework

The Registration Server's Admin Console requires PHP and the PEAR framework to enable a few additional PHP packages which are not available in RPM format.

CentOS 7/8 will be shipped with a not longer supported PHP version. PHP only supports version 7.x. To install the latest version 7 add the two additional Remi and EPEL repositories and enable PHP 7.4.

For CentOS 7:

```
yum -y install epel-release
yum install http://rpms.remirepo.net/enterprise/remi-release-7.rpm -y
yum update
```

Please activate the PHP 7.4 version in the new repository:

```
yum-config-manager --enable remi-php74
```

If `yum-config-manager` can't be found, install it with:

```
yum install yum-utils
```

Use the following command to install the required PHP components:

```
yum install php php-pear php-mysqlnd.x86_64 php-mbstring
```

For CentOS 8:

```
dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
dnf install https://rpms.remirepo.net/enterprise/remi-release-8.rpm
dnf module enable php:remi-7.4
dnf install php php-cli php-common php-pear php-mysqlnd php-mbstring
```

CentOS 7 and 8:

```
pear channel-update pear.php.net
pear install Log
```

You can use `pear list` to get a list of installed PHP packages.

Finally, we need to change a few PHP-related configuration options. Please edit the `/etc/php.ini` file and change the following values by executing the search and replace calls using `sed`:

```
sed -i 's/expose_php = On/expose_php = Off/g' /etc/php.ini
sed -i 's/max_execution_time = 30/max_execution_time = 900/g' /etc/php.ini
sed -i 's/max_input_time = 60/max_input_time = 900/g' /etc/php.ini
sed -i 's/post_max_size = 8M/post_max_size = 55M/g' /etc/php.ini
sed -i 's/upload_max_filesize = 2M/upload_max_filesize = 50M/g' /etc/php.ini
sed -i 's/memory_limit = 128M/memory_limit = 512M/g' /etc/php.ini
sed -i 's/session.use_strict_mode = 0/session.use_strict_mode=On/g' /etc/php.ini
sed -i 's/session.cookie_httponly =/session.cookie_httponly=On/g' /etc/php.ini
sed -i 's/session.cookie_secure =/session.cookie_secure=On/g' /etc/php.ini
sed -i 's/session.cookie_samesite =/session.cookie_samesite="Strict"/g' /etc/php.
→ini
sed -i 's/;date.timezone =/date.timezone = Europe\/Berlin/g' /etc/php.ini
```

Now create the following directory for storing the PHP session data:

```
install -d -o apache -g apache /var/lib/php/session
```

Optional for CentOS 7:

The Reg-Server Admin-Console offers a new functionality to ping all host servers and listing their SSL certificates and their expiration dates. This functionality requires a curl version >= 7.50. CentOS 7 offers only the curl version 7.29. A newer curl version is available in an alternative repository (this configuration is not necessary for CentOS 8 which already uses the curl version 7.61):

```
rpm -Uvh http://www.city-fan.org/ftp/contrib/yum-repo/city-fan.org-release-2-1.
→rhel7.noarch.rpm
yum --enablerepo=city-fan.org update curl
```

> **Warning:** Do not start the Apache HTTP Server until you have concluded the Registration Server installation and you are ready to proceed with the Registration Server Setup!

REGISTRATION SERVER SOFTWARE INSTALLATION

5.1 Enabling the TeamDrive Registration Server `yum` Repository

The TeamDrive Registration Server components are available in the form of RPM packages, hosted in a dedicated `yum` repository. This makes the installation and applying of future updates very easy — you can simply run `yum update` to keep your Registration Server software up to date.

To enable the repository, you need to download the `td-regserver.repo` file and place it into the directory `/etc/yum.repos.d/`, e.g. by using `wget`:

```
[root@regserver ~]# wget -O /etc/yum.repos.d/td-regserver.repo \
http://repo.teamdrive.net/td-regserver.repo
```

This will enable the "TeamDrive Registration Server Version 4.6" repository, which you can check by running `yum repolist` afterwards:

```
[root@regserver ~]# yum repolist
Loaded plugins: security
repo id                  repo name                                    status
td-regserver-4.6        TeamDrive Registration Server Version 4.6        4
base                    CentOS-7 - Base                              6.367
extras                  CentOS-7 - Extras                               14
updates                 CentOS-7 - Updates                           1.094
repolist: 7.477
```

5.2 Installing the Registration Server package

Note: CentOS 7: A newer curl version is required from an additional repository. Please install the repository and update curl to the latest version (only CentOS 7, CentOS 8 is already shipped with the required curl version):

```
rpm -Uvh http://www.city-fan.org/ftp/contrib/yum-repo/city-fan.org-release-2-1.
↪rhel7.noarch.rpm
yum -y --enablerepo=city-fan.org update curl
```

To install the Registration Server Software, install the following package via `yum` from the "TeamDrive Registration Server" repository:

```
[root@regserver ~]# yum install td-regserver
```

The TeamDrive Registration Server requires the Yvva Runtime Environment. Yvva is a development platform for the production of client-server and web applications and replaces the PrimeBase Application Server that was used in previous versions of the Registration Server (up to and including 3.0.018).

The `td-regserver` package has a dependency on the `yvva` RPM package that provides the Yvva Runtime Environment — the `yum` package manager will automatically take care of installing it.

5.3 Installing the Administration Console

The PHP-based Administration Console can be installed on the same server where the Registration Server has been installed. Alternatively, it can be installed on any other web server that supports Apache and PHP. In this case, you need to ensure that the host running the Admin Console can access the Registration Server's MySQL Database as well as the Registration Server's and Host Server's API URLs.

To install the Administration Console, install the following package via `yum` from the "TeamDrive Registration Server" repository:

```
[root@regserver ~]# yum install td-regserver-adminconsole
```

The installation package ships with an example configuration file `/var/www/html/tdlibs/globals-sample.php`, which needs to be renamed to `globals.php` and configured to match your environment. If the Administration Console is installed on the same host, the `mysql_install.sh` script described in the following chapter will take care of this automatically.

5.4 Installing the Registration Server HTML Documentation (optional)

Beginning with Registration Server version 3.0.018.5, the documentation (in HTML format) can be installed locally, so you can access it directly from the Registration Server (or any other host running an Apache HTTP Server).

To install the HTML Documentation, install the following package via `yum` from the "TeamDrive Registration Server" repository:

```
[root@regserver ~]# yum install td-regserver-doc-html
```

The HTML documents will be installed in the directory `/var/www/html/td-regserver-doc`. From your web browser, open the following URL to access the documentation:

http://regserver.yourdomain.com/td-regserver-doc/

Note: This step is optional. If you leave the documentation installed when the Registration Server goes into production and is accessible from the public Internet, you should ensure to restrict access to this URL to trusted hosts or networks only. This can be achieved by adding the appropriate access control rules to the file `/etc/httpd/conf.d/td-regserver-doc.conf`.

5.5 Installing the Registration Server External Authentication (optional)

If you install the External Authentication on a separate instance, please install these packages again, otherwise proceed with the next step:

```
[root@regserver ~]# yum install httpd mod_ssl php php-pear php-mbstring
[root@regserver ~]# pear channel-update pear.php.net
[root@regserver ~]# pear install Log
```

To install the External Authentication reference implementation, install the following package via `yum` from the
"TeamDrive Registration Server" repository:

```
[root@regserver ~]# yum install php-ldap.x86_64 php-pecl-mcrypt.x86_64 openldap-
↪clients mcrypt.x86_64
[root@regserver ~]# yum install td-regserver-ext-auth
```

The files will be installed in the directory `/var/www/html/authservice`. Before you can use external
authentication you must duplicate the file "ldap_config.php.example" and rename it to "ldap_config.php". Then
edit the parameters is the file as required.

Note: This step is optional. See the chapter ldap_ext_auth_service in the *TeamDrive Registration Server Admin-
istration Guide* for details.

5.6 Installing the Registration Server client log upload (optional)

To install the client log upload script, install the following package via `yum` from the "TeamDrive Registration
Server" repository:

```
[root@regserver ~]# yum install td-regserver-logupload
```

The php upload script will be installed in the directory `/var/www/html/upload`.

Note: This step is optional. See the chapter about the client upload configuration as described in client_log_files.

5.7 Create MySQL Database User and the Databases

The TeamDrive Registration Server requires two MySQL databases `td2reg` and `td2apilog`, which will be
accessed using a dedicated `teamdrive` MySQL user.

The Registration Server installation package ships with a script that performs the required configuration steps:

- Modify the local configuration file `/etc/my.cnf`, start and enable MySQL Server (only when using a
  local MySQL Server)

- Create the required MySQL user `teamdrive`, assign the provided password and the required database
  privileges (requires access to the MySQL `root` user)

- Create and populate the required Registration Server MySQL databases

- Modify the local Registration Server configuration files `/etc/td-regserver.my.cnf` and
  `/var/www/html/tdlibs/globals.php` (if installed).

The following example assumes that the MySQL database is located on the same system where the TeamDrive
Registration Server instance is installed.

If the MySQL Database is hosted on a different system, replace the MySQL host name `localhost` with the host
name or IP address that the MySQL instance is running on.

You need to have the following information available:

- The password of the MySQL `root` user

- The password that you want to assign to the `teamdrive` user

The script is part of the `td-regserver` package and is installed in
`/opt/teamdrive/regserver/mysql/mysql_install.sh`. Call it as the `root` user and follow
the instructions:

```
[root@regserver ~]# /opt/teamdrive/regserver/mysql/mysql_install.sh

TeamDrive Registration Server MySQL Database Install Script
-----------------------------------------------------------

Configuring MySQL database for TeamDrive Registration Server
version 3.6.0

This script will perform the following steps:

 - Modify the local configuration file /etc/my.cnf,
   start and enable MySQL Server
   (only when MySQL Server runs locally)
 - Create the required MySQL user "teamdrive",
   assign the provided password and the required
   database privileges
   (requires access to the MySQL root user)
 - Create and populate the required Registration Server
   MySQL databases
 - Modify the local Registration Server configuration files
   /etc/td-regserver.my.cnf and /var/www/html/tdlibs/globals.php
   (if installed)

Enter MySQL hostname: localhost
Enter MySQL root password for localhost: <root password>
Enter MySQL password to be set for user teamdrive: <teamdrive password>

mysqld (pid  10162) is running...
Stopping mysqld:                                           [  OK  ]
Changing local MySQL Server configuration...
Backing up existing configuration file /etc/my.cnf...
`/etc/my.cnf' -> `/etc/my.cnf-2015-04-20-11:59.bak'
Removing old InnoDB log files...
`/var/lib/mysql/ib_logfile0' -> `/var/lib/mysql/ib_logfile0-2015-04-20-11:59.bak'
`/var/lib/mysql/ib_logfile1' -> `/var/lib/mysql/ib_logfile1-2015-04-20-11:59.bak'
Starting and enabling MySQL Server...
Starting mysqld:                                           [  OK  ]
Trying to connect to the MySQL server as root...
+---------------+
| MySQL Version |
+---------------+
| 5.1.73        |
+---------------+
Creating teamdrive MySQL user on localhost
Trying to connect to the MySQL server as the teamdrive user...
Creating Registration Server databases...
====================
CREATE DATABASE td2reg
====================
CREATE TABLE TD2User
create table TD2UserBlob
create table TD2FreeUserStorage
create table TD2Device
create table TD2Message
create table TD2MessageSF
create table TD2MessageFD
create table TD2Ticket
create table TD2Email
create table TD2AutoTask
create table TD2Owner
create table TD2OwnerMeta
create table TD2OwnerMetaSetting
create table TD2TicketChanges
```

```
create table TD2LicenceType
create table TD2OwnerLicenceType
create table TD2Product
create table TD2OwnerProduct
create table TD2Depots
create table TD2RegServerList
create table TD2Setting
create table TD2UserPrivileges
create table TD2UserPrivilegesSetting
create table TDAddressRange
create table TD2Parcel
create table Keys
========================
CREATE DATABASE td2apilog
========================
create table TD2APIRequests
Updating /etc/td-regserver.my.cnf...
Backing up existing configuration file ...
`/etc/td-regserver.my.cnf' -> `/ext/td-regserver.my.cnf-2015-04-20-12:01.bak'
Setting up /var/www/html/tdlibs/globals.php...
`/var/www/html/tdlibs/globals.php' -> `/var/www/html/tdlibs/globals.php-2015-04-20-
→12:01.bak'

Finished!
The MySQL configuration for TeamDrive Registration Server
version 3.6.0 is now complete.
```

Among other things, the `mysql_install.sh` script modifies a few run-time parameters in the MySQL server configuration file `/etc/my.cnf` — review these carefully and adapt them to match your system configuration as outlined in the MySQL Reference Manual.

In particular, the value for `innodb_buffer_pool_size` should be adjusted to the amount of main memory (RAM) available in your system; typically this value should be set to about 80% of the total memory. Also, the size of the InnoDB log files defined in `innodb_log_file_size` might be worth reviewing.

> **Warning:** Changing the value of `innodb_log_file_size` after MySQL has already been started will lead to InnoDB error messages when the MySQL server restarts, e.g.:
>
> ```
> InnoDB: Error: log file ./ib_logfile0 is of different size 0 5242880 bytes
> InnoDB: than specified in the .cnf file 0 67108864 bytes!
> ```
>
> In order to avoid these, you need to shut down the MySQL Server cleanly, move away the current InnoDB log files (named `ib_logfile0`, `ib_logfile1` and so on), and restart MySQL, so InnoDB can re-create these logs with the correct size.
>
> See http://www.percona.com/blog/2011/07/09/how-to-change-innodb_log_file_size-safely/ for more details.

As a final test, try logging into the MySQL database from the Registration Server system, using the `teamdrive` user and the password you defined — you should be able to see and access the TeamDrive Registration Server databases:

```
[root@regserver ~]# mysql -u teamdrive -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 7
Server version: 5.1.73 Source distribution

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
```

```
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| td2apilog          |
| td2reg             |
+--------------------+
3 rows in set (0.00 sec)

mysql> QUIT
Bye
```

The MySQL database has now been configured and populated with the required databases and tables.

5.8 CentOS Hardening

We recommend to harden the CentOS system as described in *CentOS Hardening* (page 10).

After installation execute the script:

```
/opt/teamdrive/regserver/mysql/os_hardening.sh
```

to automatically configure the hardening settings. Reboot the system afterwards, because the settings will ony be activate after a reboot.

Check the results with both tools:

```
inspec exec https://github.com/dev-sec/linux-baseline
lynis audit system
```

REGISTRATION SERVER CONFIGURATION

This chapter will guide you through the initial configuration of the TeamDrive Registration Server.

The web-based setup process will perform the following steps:

- Defining the Registration Server Identity (e.g. Server Type, Server Name, Provider Code)
- Registering the Registration Server with the selected TDNS and Master Registration Server (optional, when selecting the default Server Type "Standard")
- Setting up the Default Provider (e.g. username/password, API and login access, contact details)
- Registration Server SMTP configuration (SMTP server, email addresses)
- Verification of the SMTP configuration

Once this initial setup has been concluded, other configuration aspects of the Registration Server can be modified using the Registration Server's Administration Console.

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

6.1 Start the Apache HTTP Server

Start the Apache HTTP Server to proceed with the Registration Server configuration:

```
[root@regserver ~]# service httpd start
```

> **Warning:** At this point, the Registration Server's web server is answering incoming requests from any web client that can connect to its address. For security purposes, you should not make it accessible from the public Internet until you have concluded the initial configuration, e.g. by blocking external accesses using a firewall.

6.2 Start the Web Based Setup Process

From a desktop system that can connect to the Registration Server via HTTPS, start a web browser like Mozilla Firefox, Google Chrome (or any other modern web browser) and start the configuration process by opening the following URL in your browser:
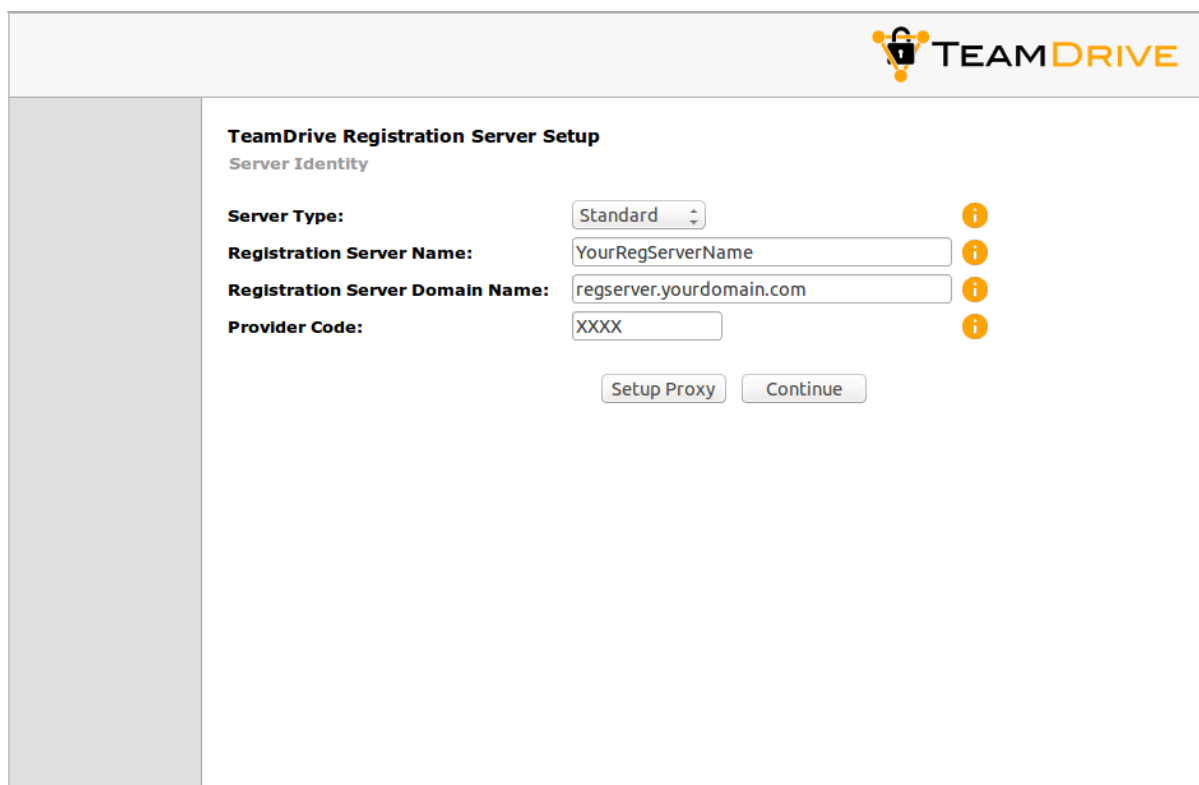
https://regserver.yourdomain.com/setup/

This should open the first Registration Server Setup page. If you get an error message like "500 Internal Server Error", check the log files for any errors. See chapter web installation 500 internal server error for details.

If you have performed a partial setup of the server before, the process will continue with the next unfinished step.

Note: If you haven't replaced the server's self-signed default SSL certificate yet, your web browser most likely will complain about an untrusted/insecure connection. Either replace the SSL certificate with an appropriate one before you proceed, or ignore this message.

6.3 Server Identity

The first step is to define the Registration Server's "identity", in particular what type of server you want to set up, the server's name and your Provider Code.

Fig. 6.1: Registration Server Setup: Configuring the Registration Server Identity

Enter the following information in the appropriate fields:

Server Type Select what type of Registration Server you wish to setup. A **Standard Registration Server** can host one or multiple Providers/tenants and is connected to the TeamDrive Master Registration Server and the TeamDrive Name Service (TDNS). This is the default.

A **Standalone Registration Server** is not connected to a Master Registration Server and/or TDNS.

A **Master Registration Server** is connected to a TeamDrive Name Server (TDNS) and has a number of Standard Registration Servers. If you want to setup a Master Registration Server, you also need to setup your own TDNS instance.

Note: Note that a custom TeamDrive Client is required to connect to a Standalone or Master Registration Server.

Registration Server Name Enter the name of your Registration Server, e.g. RegServerXXXX (where XXXX is your provider code), or RegServerYourCompany. The name may not include spaces and must be unique for the entire TeamDrive Registration Server Network. A TeamDrive Network consists of a Registration Server, connected to a central Master Registration Server and a TDNS (TeamDrive Name Server). Consult

---

your TDNS or Master Registration Server Operator if you have questions about selecting an appropriate name here.

Registration Server Domain Name Enter the domain name of your Registration Server. This is the domain name of the Apache Web-server that will serve data to the TeamDrive clients and must be resolvable via DNS by the TeamDrive Clients.

Note: Dont use an **IP address** instead of a domain name, because using an IP address will cause the following problems: Register a SSL certificate for an IP address can be a problem and the TeamDrive client applications on the Apple platform (MAC and IOS) require a valid and official SSL certificate for the HTTPS communication. Apple only allows HTTPS connections. The second problem is, that the IP of the server cant be changed anymore. There is no possibility in the TeamDrive clients to modify the Registration Server URL. If the server will be not longer reachable by the initial name, the clients will not be able to send or recieve invitations and other informations from the Registration Server.

Provider Code Enter your Provider Code. The Provider Code (aka Distributor Code) is a 4 character code, consisting of letters A-Z and 0-9. The Provider Code must be unique when your Registration Server is connected to TDNS. Contact your TDNS operator (usually TeamDrive Systems), to obtain and/or register your provider code.

6.3.1 HTTP Proxy Setup (optional)

When concluding this step, the setup will submit a "ping" HTTP request to verify that the Registration Server is reachable via the provided host name.

If outgoing HTTP requests initiated by the Registration Server are blocked by a firewall and need to be sent via a proxy server, you can configure it by clicking **Setup Proxy**.
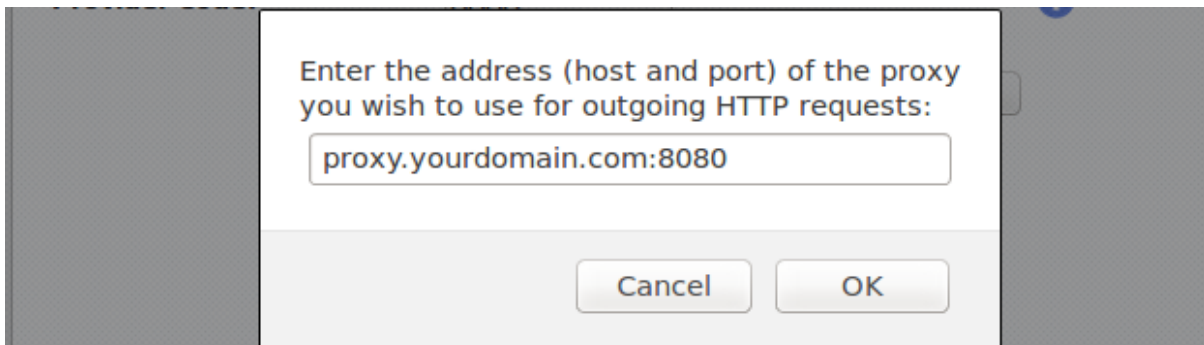
Fig. 6.2: Registration Server Setup: Configuring the HTTP Proxy

In the popup window, enter the proxy's host name and TCP port, if required.

Note: Note that the Registration Server currently does not support proxy auto-config (PAC) files, the Web Proxy Autodiscovery Protocol (WPAD) or proxy servers that require some form of authentication.

Click **OK** to save the proxy settings, or **Cancel** to abort.

Click **Continue** to proceed to the next step.

6.4 Server Registration

This step will register your Registration Server with the TeamDrive Name Service (TDNS) and the Master Registration Server.

For more details about TDNS, see chapter teamdrive name server (tdns).

Note: This step will be skipped entirely, if you are setting up a "Standalone" Registration Server, as it does not have to be registered with TDNS or a Master Registration Server.

If you set up a Master Registration Server, only the TDNS-related information needs to be entered.

Fig. 6.3: Registration Server Setup: Server Registration

Each Registration Server has a unique "Authorization Sequence" that is required so that Clients can submit invitation messages to users managed on other Registration Servers within the TDNS Network.

Take note of the following information and send it as text, not as an image, to the operator of the Master Registration Server and/or TDNS (usually to TeamDrive Systems via support@teamdrive.net).

Registration Server Name The unique Registration Server name you entered in Step server identity.

Registration Server Domain Name Your Registration Server's resolvable public DNS name.

Provider Code Your Provider Code.

Authentication Sequence A randomly generated code that is unique for each Registration Server and is used for the authorization of Clients that want to exchange encrypted messages with that Registration Server.

The Administrator will register your Registration Server, and provide you with the information required to fill in the TDNS fields described below.

Note: Without the information the standard TeamDrive client could not contact your server. You will find more details about the communication between clients and different Registration Server in the chapter teamdrive name server (tdns).

Enter the following information in the appropriate fields:

Master Registration Server Name All Standard Registration Servers must be connected by a Master Registration Server. By default, this is `TeamDriveMaster`.

Master Registration Server Host Enter the host name of the Master Registration Server. Setup will attempt to register your Registration Server with the master server running on this host. By default, this is `reg.teamdrive.net`.

TDNS Host This is the host name of the TDNS (TeamDrive Name Server). By default, this is `tdns.teamdrive.net`.

TDNS Server ID This is an ID allocated by TDNS when the Provider Code is registered on TDNS. You need to obtain if from your TDNS operator.

TDNS Checksum This is an unique code which is generated by TDNS when the Provider Code is registered on TDNS. You need to obtain if from your TDNS operator.

Note: Before proceeding with a Standard Server setup, you must have entered your **TDNS Server ID** and **TDNS Checksum**, which will be provided by your TDNS/Master Registration Server operator (usually by TeamDrive Systems) after you submitted your Registration Server's details as outlined above.

6.4.1 HTTP Proxy Setup (optional)

When concluding this step, the Registration Server will attempt to send HTTP requests to the TDNS and Master Registration Server, to verify they can be reached via the host names you provided and that the TDNS checksum was entered correctly.

If outgoing HTTP requests need to be sent via a proxy server (and you haven't done so in the first step already), you can configure it by clicking **Setup Proxy**.
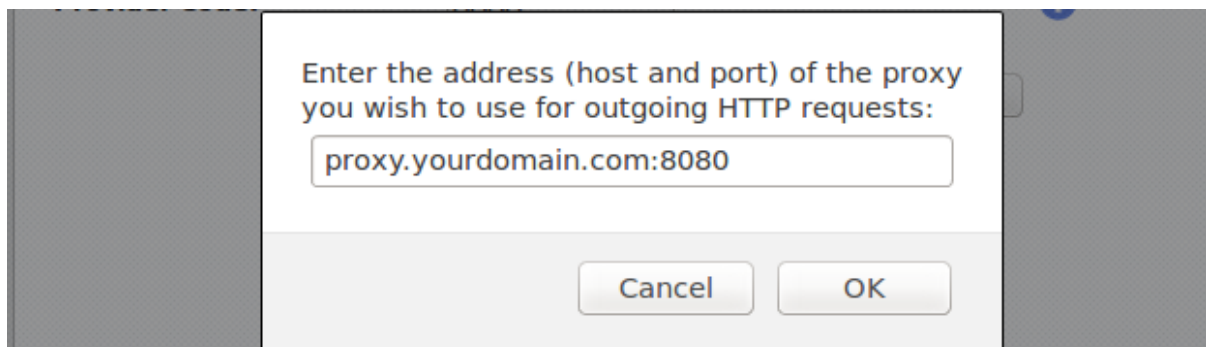
Fig. 6.4: Registration Server Setup: Configuring the HTTP Proxy

In the popup window, enter the proxy's host name and TCP port, if required and click **OK** to save the proxy settings, or **Cancel** to abort.

Click **Continue** to proceed to the next step, or **Back** to return to the previous step.

6.5 Provider Setup

In this step, you create the user associated with your default provider. This user has all privileges required to manage all aspects of the Registration Server as well as all providers hosted on this Registration Server.

Fill in your details as described below.

Username The username of the Registration Server administrator used to login to the Administration Console.

Password Password of the Registration Server administrator used to login to the Administration Console.

Fig. 6.5: Registration Server Setup: Provider Setup

API Access List This is a comma separated list of IP addresses of the hosts that are allowed to access the Registration Server API. You must include the IP address of the host that will be running the Administration Console.

Admin Login Access List This is a comma separated list of IP addresses of the hosts that are allowed to login to the Registration Server Administration Console. This should include the IP address of the browser you are currently using. If the list is empty, access is allowed from any host. This setting is not recommended.

First Name The given name of the Registration Server administrator.

Last Name The surname of the Registration Server administrator.

Email Address Email address of the Registration Server administrator.

Company Name The company name of the Registration Server administrator.

Telephone Telephone number used to contact the Registration Server administrator.

Click **Continue** to proceed to the next step, or **Back** to return to the previous step.

6.6 Email Configuration

The TeamDrive Registration Server needs to be able to send out various notifications (e.g. Space invitations, License modifications) via SMTP.

In this step, you enter the required details about how the Registration Server contacts the MTA and which email addresses should be used for sending out emails. Fill out the fields according to your requirements.

Note: The Yvva Runtime Environment that provides the foundation for the Registration Server is only capable of sending out email using plain SMTP via TCP port 25 to a local or remote MTA.

Fig. 6.6: Registration Server Setup: Email Configuration

If your mail server requires some form of authentication or transport layer encryption like SSL/TLS, you need to set up a local MTA that relays all outgoing email from the TeamDrive Registration Server to your mail server using the appropriate protocol and credentials.

SMTP Server This is the host name (and TCP port) of the SMTP server used to send emails, e.g. `smtp.yourdomain.com:25`. The TCP port number can be omitted, if it's the default port for SMTP (25).

Send Timeout The timeout (in seconds) before an email submission to the SMTP server will be aborted, if there is no reply.

Sender Email Address This is the email address that will appear as sender in email envelope. Sometimes this address is also used as the "From" email address.

Email Sending Host This is the host name of the system that will send the email (aka the HELO host). The value should identify the system sending the email, you should use an externally addressable DNS name for this value (usually the Registration Server's host name).

Administrator Email Email address of the Registration Server administrator. This address will be used to send a test email, before the setup can be completed.

Click **Continue** to proceed to the next step, or **Back** to return to the previous step.

6.7 Email Confirmation

To test that the SMTP setup is functional, the setup process will send an email to the address you provided as the *Administrator Email* in the previous step.

If you don't receive the email within some minutes, check your mail server's log files (e.g. `/var/log/maillog`) and the sender's email account for errors or bounce messages and adjust the SMTP server configuration accordingly.
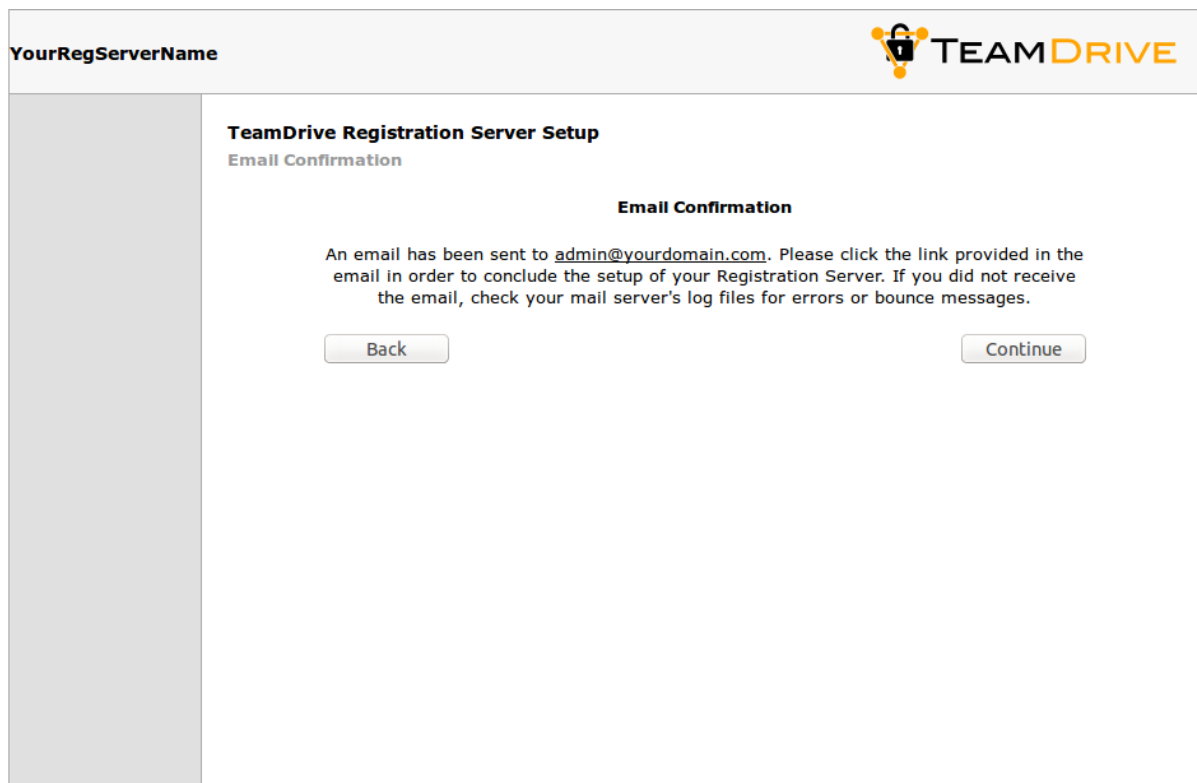
Fig. 6.7: Registration Server Setup: Email Confirmation

If you received the email, the SMTP service for the TeamDrive Registration Server has been configured correctly.

Please click the link provided in the email (or copy and paste it into your web browser's address bar) in order to conclude the setup of your Registration Server.

6.8 Setup Complete

After you have clicked the confirmation link provided in the email, you will see a confirmation page.

At this point, you have completed the initial setup of your Registration Server successfully.
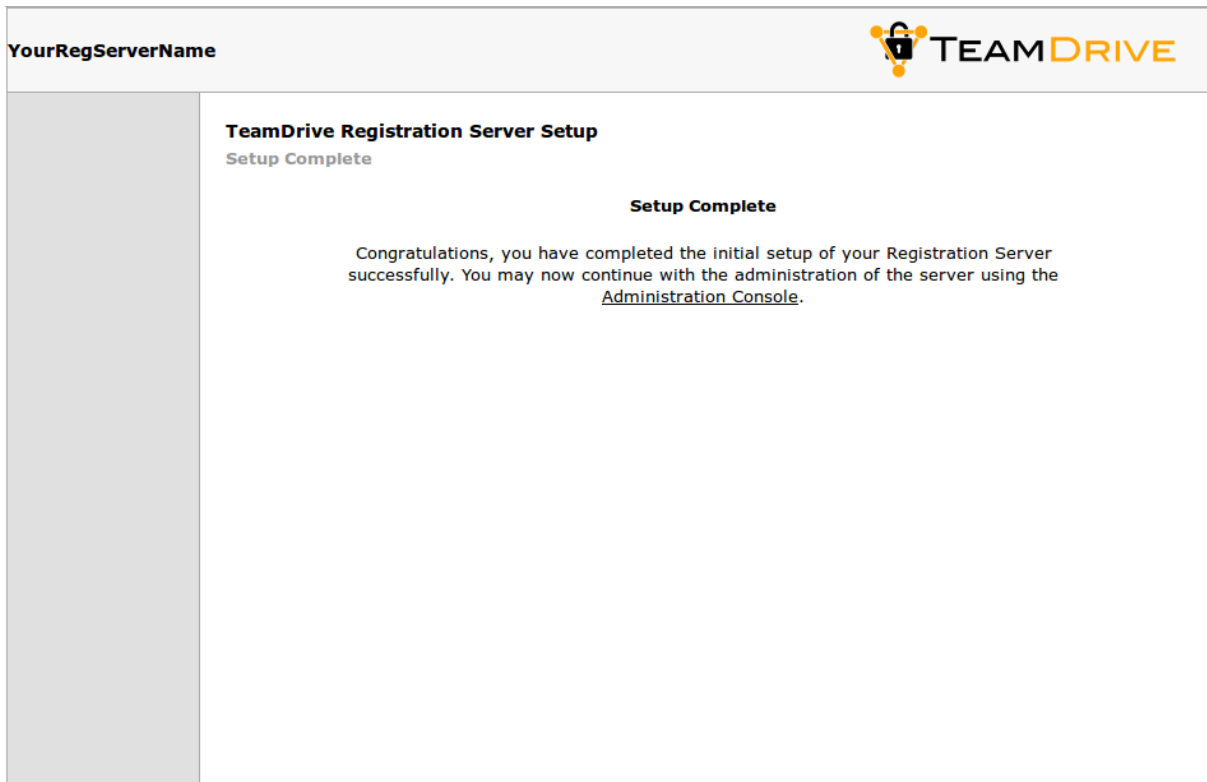
Fig. 6.8: Registration Server: Setup Complete

STARTING AND STOPPING THE TEAMDRIVE REGISTRATION SERVER COMPONENTS

To make the TeamDrive Registration Server available for TeamDrive Clients to connect, the following services need to be up and running:

- `mysqld` — the MySQL database server (local or on a remote server)
- `httpd` — the Apache HTTP Server
- `td-regserver` — the Yvva based background processes
- `postfix` — the Postfix SMTP server (optional, other MTAs like sendmail or qmail or MTAs on remote servers can be used as well)

After the initial installation, most services except for the `td-regserver` service should already be up and running.

To ensure a proper service start and to minimize error messages on the TeamDrive Client side, the following startup sequence of the TeamDrive Registration Server components and services should be observed.

Start the TeamDrive Registration Server services in the following order:

1. Start the Registration Server MySQL databases service
2. Start the SMTP service (or make sure it's available/accessible)
3. Start the `td-regserver` background service
4. Start the Apache HTTP Server

For testing purposes, you can start these services manually, using the `service` command. In a production environment, these services should be started automatically at boot time, by enabling them via the `chkconfig` tool.

7.1 Starting services manually

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

```
[root@regserver ~]# service mysqld start
[root@regserver ~]# service postfix start
[root@regserver ~]# service td-regserver 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 td-regserver stop
[root@regserver ~]# service postfix 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, Postfix (optional) and the `td-regserver` service 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 postfix on
[root@regserver ~]# chkconfig --levels 235 td-regserver on
```

Note: It's important, that the MySQL service starts before the Apache will start. On CentOS 7 edit the file:

```
/lib/systemd/system/httpd.service
```

and add at the end of the line starting with `After=` the entry `mysqld.service`. This will ensure, that the Apache will start after the MySQL service. You can verify the service start dependencies with (after a reboot of the system):

```
[root@regserver ~]# systemd-analyze critical-chain
```

7.4 Logging into the Administration Console

At this point, you can now continue with the administration and configuration of the Registration Server using the Administration Console, which can be reached via the following URL:

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

To log in, enter the login credentials of the Provider you defined in Step provider setup.

Please see the *TeamDrive Registration Server Administration Guide* for a detailed description of the Administration Console and for further details on the configuration and customization of the Registration Server and the TeamDrive Clients connecting to your Server.

Once you have concluded the configuration, start a TeamDrive Client and register a user after entering your Provider Code (or log in using a user that is provided via external authentication or via CSV import).

Consult the TeamDrive Client Documentation for usage details.

TROUBLESHOOTING

8.1 List of relevant configuration files

/etc/httpd/conf.d/td-regserver.httpd.conf: This configuration file loads and enables the TeamDrive Registration Server-specific Apache module `mod_yvva.so`. This Apache module is responsible for providing the web-based Registration Server Installer and the Registration Server API.

/etc/logrotate.d/td-regserver: This file configures how the log files belonging to the TeamDrive Registration Server are being rotated. See the `logrotate(8)` manual page for details.

/etc/td-regserver.conf: This file defines how the `td-regserver` background service is started using the `yvvad` daemon.

/etc/td-regserver.my.cnf: This configuration file defines the MySQL credentials used to access the `regdb` MySQL database. It is read by the Apache module `mod_yvva`, the PHP-based Administration Console as well as the `yvvad` daemon that runs the `td-hostserver` background tasks and the `yvva` command line client.

/etc/yvva.conf: This configuration file contains configuration settings specific to the Yvva Runtime Environment that are shared by all Yvva components, namely the `mod_yyva` Apache module, the `yvvad` daemon and the `yvva` command line shell.

/var/www/html/tdlibs/globals.php: This configuration file defines the MySQL login credentials required for the TeamDrive Registration Server Administration Console.

8.2 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/td-regserver.log`: The log file of the `mod_yvva` Apache module that performs the actual Registration Server functionality (e.g. Client/Server communication and API calls) and the web-based initial setup process. The amount of logging information can be defined by changing the value `YvvaSet log-level` in configuration file `/etc/httpd/conf.d/td-regserver.httpd.conf`. The following debug levels (with increasing verbosity) can be set: `error`, `warning`, `notice`, `trace` or `debug`. The default is `error`. Changing this value requires a restart of the Apache HTTP Server.

  This log file is also used by the `td-regserver` background service (managed by `yvvad`). The amount of logging information can be defined by changing the value `log-level` in configuration file `/etc/td-regserver.conf`. The following debug levels (with increasing verbosity) can be set: `error`, `warning`, `notice`, `trace` or `debug`. The default is `error`. Changing this value requires a restart of the `td-regserver` service using `service td-regserver restart`. This log file needs to be owned by the Apache user. Logging only occurs if the log file exists and is writable by the Apache user.

- `/var/log/httpd/`: The Apache HTTP 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.log`: A log file to keep track of various events on the Administration Console, e.g.

  - Failed logins

  - Failed two-factor-authentication attempts (only admin console logins, not client two-factor-authentication attempts)

  - Password changes

  - Changes to security-related Provider/Server settings (login timeouts, API access lists, etc.)

  - Modifications of user privileges

  - Failed session validations

8.3 Enable Logging with Syslog

As outlined in list of relevant log files, the TeamDrive Registration Server logs critical errors and other notable events in various log files by default.

Starting with Registration Server version 3.5 and Yvva 1.2, it is now possible to redirect the log output of most server components to a local `syslog` instance as well.

Syslog support is an essential feature for auditing, security and/or compliance reasons, as it allows you to funnel all log messages into a centralized syslog server.

This makes it easier to monitor the logs for critical events or errors and prevents tampering with the log files in case of a security breach. It also helps to maintain control over the disk space utilization on the server, as growing log files can't accidentally fill up the file system.

To enable syslog support, the log file name in the `log-file` setting has to be replaced with the keyword `syslog`. Optionally, a custom process identifier can be supplied, by appending it to the `syslog` keyword, using a colon as the separator, e.g. `log-file=syslog:my_process_identifier`. If not used, the default process identifier will be used, which is the name of the program executable.

To enable syslog support for the Yvva-based `td-regserver` background service, edit the `log-file` setting in file `/etc/td-regserver.conf` as follows:

```
log-file=syslog:td-regserver
```

You need to restart the `td-regserver` background service via `service td-regserver restart` in order to activate this change. If the `log-level` is set to `debug` you will now see log messages appearing in `/var/log/messages`:

```
Jun 23 14:13:43 localhost td-regserver: notice: yvvad startup
Jun 23 14:13:43 localhost td-regserver: notice: Using config file:
/etc/td-regserver.conf
Jun 23 14:13:43 localhost td-regserver: notice: No listen port
Jun 23 14:13:43 localhost td-regserver: notice: yvvad running in repeat 10
(seconds) mode
```

To enable syslog support for the Registration Server Client/Server communication and API, edit the `YvvaSet` `log-file` setting in file `/etc/httpd/conf.d/td-regserver.httpd.conf`:

```
YvvaSet log-file=syslog
```

You need to restart the Apache HTTP Server via `service httpd restart` in order to activate this change. If the `log-level` is set to `debug` you will now see log messages appearing in `/var/log/messages`:

```
Jun 23 14:21:01 localhost mod_yvva: notice: mod_yvva 1.2.1 (May 21 2015
11:00:12) startup OK
```

To enable logging of security related Administration Console events to syslog instead of the log file `/var/log/td-adminconsole.log`, you need to change the Registration Server Setting `Security/EnableSyslog` to `True` via the Administration Console.

Click **Admin** -> **Server Settings** -> **Security** and change the **Value** for `EnableSyslog` to `True`. Click **Save** to apply the change. From this point on, security relevant events triggered via the Administration Console will be logged to `/var/log/secure`:

```
Jun 23 14:25:36 localhost td-adminconsole-log[4165]: 2015-23-06 14:25:36
[info] [/var/www/html/adminconsole/editSettings.php:38]: RegServer setting
'EnableSyslog' changed from '$false' to '$true' by user 'xxxx'
Jun 23 14:29:58 localhost td-adminconsole-log[4168]: 2015-23-06 14:29:58
[info] [/var/www/html/adminconsole/libs/auth.php:48]: Failed login for
user 'xxxx'
Jun 23 14:34:09 localhost td-adminconsole-log[4161]: 2015-23-06 14:34:09
[info] [/var/www/html/adminconsole/changePassword.php:54]: Password for
user 'xxxx' has been changed
```

8.4 Common errors

8.4.1 Web Installation: "500 Internal Server Error"

This error can be triggered by several error conditions. Check the log file `/var/log/td-regserver.log` for details.

Some common errors include:

```
[Error] -12036 (2002): Can't connect to local MySQL server through socket
'/var/lib/mysql/mysql.sock' (25)
[Error] "open TD2REG_WRITE dbms option '[regdb]';" (1)
[Error] "sql.pbt" SQL:openDBMSAndDB(387)
[Error] "startup.yv" (32)
```

The local MySQL Server's socket file can't be opened. This could either be a permission problem, or the MySQL Server is simply not available. Check that MySQL is actually up and running (e.g. by running `service mysqld status`) and restart it, if necessary. If the error persists, check the MySQL error log file (usually `/var/log/mysqld.log`) for hints.

Similarly, an error like the following one indicates that a remote MySQL Server might not be answering (e.g. because of a firewall rule or because it's not running):

```
[Error] -12036 (2003): Can't connect to MySQL server on
'mysql.yourdomain.com' (107)
[Error] "open TD2REG_WRITE dbms option '[regdb]';" (1)
[Error] "sql.pbt" SQL:openDBMSAndDB(387)
[Error] "startup.yv" (32)
```

If you see `Access denied` errors like the following one:

```
[Error] -12036 (1045): Access denied for user 'teamdrive'@'localhost' (using
password: YES)
[Error] "open TD2REG_WRITE dbms option '[regdb]';" (1)
[Error] "sql.pbt" SQL:openDBMSAndDB(387)
[Error] "startup.yv" (32)
```

Either the username or password used to connect to the MySQL Server are wrong. Double check that the MySQL username and password provided in `/etc/td-regserver.my.cnf` are correct, e.g. by trying to connect to the MySQL server using these credentials with the `mysql` command line client.

If you see the following error when connecting to a remote MySQL Server:

```
[Error] -12036 (1130): Host 'regserver.yourdomain.com' is not allowed to
connect to this MySQL server
[Error] "open TD2REG_WRITE dbms option '[regdb]';" (1)
[Error] "sql.pbt" SQL:openDBMSAndDB(387)
[Error] "startup.yv" (32)
```

Check the TeamDrive MySQL user's privileges on the remote MySQL server, e.g. by running `SHOW GRANTS FOR `teamdrive`@`regserver.yourdomain.com`;` and make sure that this user is allowed to connect to the MySQL server from the Registration Server's host.

8.4.2 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 service `td-regserver` up and running? Check with `service td-regserver status` and use `service td-regserver start` to start the process. Also ensure that the service is configured to be started at system bootup time. See chapter startingstoppingcomponents for details.

- Check the `/var/log/td-regserver.log` log file for errors.

- Does sending of email work in general? Try using the `mail` utility and check your MTA logs (e.g. `/var/log/maillog`) for delivery status notifications.

8.4.3 Admin console: Error connecting to the MySQL server

If you get an error like:

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

Verify that the MySQL Server is up and running and that the connection parameters like username and password in file `/etc/td-regserver.my.cnf` are set up correctly. See chapter admin_console_config for details.

8.4.4 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 server hosting the Administration 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 Administraton Console is included in the list. If necessary, add the missing address in a new line and click **Save**.

8.4.5 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 `td-regserver` background service 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.

RELEASE NOTES - VERSION 4.6

9.1 4.6.3 (YYYY-MM-DD)

The following Admin Console changes and improvements have been implemented:

- The list of users on the Edit License page is now displayed in standard table form (REGSERVER-1601).

- When removing a user from an account that is both member and manager, you can now select to remove the user as either a member of a manager (REGSERVER-1646).

- The Add Member and Add Manager dialogs now limited to 1000 users in order to shorten the load time for the dialogs (REGSERVER-1666). Users will be prompted to use the filter function if necessary.

- In the list of license users, the Provider Code of users with a provider different to that of the licenses are highlighted in red (REGSERVER-1600).

- When adding a license to an account, the dialog list now also displays the "holder email" and the license limit and usage (REGSERVER-1634). Filtering is still only done on the first column which includes the license number and the owner (if any).

- You can now set an inbox user without also setting an inbox URL, but the URL is still required for the inbox to work (REGSERVER-1663). Setting am inbox URL without an inbox user is not allowed.

  In addition, the inbox user must have a license with the Agent or Inbox feature. Using a TeamDrive hosted inbox requires an the Inbox license feature.

9.2 4.6.2 (2011-12-16)

This release also includes a number of security improvements, please contact TeamDrive for further details.

- The Admin Console now returns ambiguous error on login, if the username/email or password is incorrect (REGSERVER-1669). This is also the case if the user does not have the permission to login to the Admin Console, or if access is only allowed from specific IP addresses.

  In the case of the Lost Password function, when a temporary password is requested the server will always return with the message that a temporary password has been sent, not matter what the input.

  Users will not be warned that the Lost Password functions is not supported when logging in as a provider.

  All errors during login are logged to the td-adminconsole.log file. Check this log file, if a user is having a problem during login.

- Added `FailedLookupLimit` and `FailedLookupPeriod` settings which limit the number of failed lookups for security reasons, during login or when inviting users (REGSERVER-1662).

  The settings `LookupRetensionTime`, `CalculatedLookupMaximum`, `RecentLookupMaximum`, and `LastLookupNotification`, allow you to control and monitor the number of allowed failed lookups.

- Added auto-task "Manage Failed Lookup" which calculates the maximum failed lookup rate over the last 48 hours. This task runs every 4 hours and resets the `RecentLookupMaximum` value.

- the "prelogin", "connect" and "lookupemail" API calls have been updated to not provide information as to the existence of a user. However in the case of "prelogin" and "connect", this breaks the TeamDrive client.

  As a result, the changes will only be made mandatory when an update to the TeamDrive client has been made generally available.

- Added support for two-factor authentication for user login on the Admin Console (REGSERVER-1674).

9.3 4.6.1 (2021-09-30)

This is a security update.

- A number of security issues have been fixed, please contact TeamDrive for further details.

- yvva 1.5.11 is required which includes measures to prevent "Log Poisoning" by encoding r and n characters (YVVA-52).

- Added `REDIRECT_SECURITY` provider setting. The SECURITY page explains how to join a space that has certain security requirements (REGSERVER-1665). The "redirect-security" HTML template is returned by default, when this page is requested.

- Admin Console: Fixed dialogs on Manage Domains & Services page.

9.4 4.6.0 (2021-08-31)

The 4.6 release includes several security bug fixes and a number of hardening measures, and is recommended to all users.

Please contact TeamDrive for further details.

Version 4.6 is an in-place upgrade to all previous versions of the server.

- Initial public release of 4.6.

- OS hardening and security update to Apache configuration.

- Set security headers in Apache configuration (REGSERVER-1654).

- Updated to the latest versions of PHP database and network libraries.

- Email verification improved.

- Number of support files/logs is now limited.

- The `TDNSURL` setting has been change from "http" to "https" by default (REGSERVER-1639). On update a once-off update will change any existing HTTP URL to HTTPS for this setting. Administrators must be aware of this change in case there is a disturbance in the communication with TDNS as a result. Note that HTTP access to TDNS has been deprecated and will be disallowed at somme point in the future.

  External authentication services are also required to use HTTPS to contact TDNS.

- The "support-notification" emails will not be sent with "From:" and "Reply-To:" headers set according to the value of the `FROM_EMAIL_OPTIONS` setting (REGSERVER-1633).

  The default value for the `FROM_EMAIL_OPTIONS` setting, has been changed to `replyto-via`. The default was previously `user`, which should no longer be used as email servers reject unknown from email addresses.

  Note that the `SUPPORT_EMAIL` must now be set to a valid email address in order to receive support uploads.

- Added the server setting: `WebPortalAPICalls` which specifies the API calls that can be made by the Web Portal (REGSERVER-1636).

- It is now possible to override the provider Web access setting, `ALLOW_WEB_PORTAL_ACCESS` at the account and user level (REGSERVER-1615).

  For this purpose the options of this setting have been changed to: `permit`, `deny`, `permit-by-default` and `deny-by-default`. The previous setting value `peruser` is equivalent to `deny-by-default`.

  At the account level, web access can be disabled, if it is enabled or permissable at the provider level. In other words if `ALLOW_WEB_PORTAL_ACCESS` is set to `permit`, `permit-by-default` or `deny-by-default`.

  See allow_web_portal_access, for more details.

- Added an new email priority level (REGSERVER-1613): All emails that the user is actively waiting for (in particular, during login) now have top priority, this includes:

  web-activationlink, web-activationsetpassword, web-activationwithnewsletter, web-emailchangedtonew, web-newpassword, confirm-email, new-passwd, reg-activationlink, reg-activationsetpassword, reg-activationwithnewsletter, reg-emailchangedtonew, too-many-failed-logins, two-factor-auth, recovery and authentication-code.

  As before, the lowest priority is assigned to notification emails sent by the TeamDrive client. All other emails, including invitations is given medium priority.

  All emails of a higher priority are sent before the emails of a lower priority. This means the lower priority emails will only be sent once the rate at which high priority emails are send drops below the overall email send rate (see emailsendrate).

- Account managers can now select a license as the "account default license" (REGSERVER-1611). All users added to the account as a member, will be automatically assigned this license, provided the user is currently using a default license (i.e. a license assigned by the provider using the `DEFAULT_LICENSEKEY` setting, or the user's own default license created using the `DEFAULT_FREE_FEATURE` setting).

  When a member using the account default license is removed from the account, the default license is revoked from the user.

  If the account default license is changed, the license is not revoked from user's that have already been assigned the license. However, if a user has been invited to the account and is scheduled to receive the account default license, this license assignment will be cancelled.

- When a user that belongs to an email domain that is registered by another Registration Server, is invited to a space, the server will now redirect the client to the other Registration Server, where the user may be created as a "guest" (REGSERVER-1606).

  In addition, the **inv-newuser-invited** email template has been changed so that, if a user created on invitation uses external authentication, then the user will receive an activation link instead of a set password link (see templates_for_client_actions).

- Admin Console: it is now possible to "ping" the Host Servers from the Server management page (REGSERVER-1551). When this is done the Registration Server will also check the Host Server version, and the expiry date of the SSL Certificate, provider the HTTPS protocol is used to access the Host Server (see `API_USE_SSL_FOR_HOST` provider setting).

- Added new email template, "depot-frozen", and other functionality to notify the user of depot exceeding the storage limit, this includes the template variables `[[LASTACCESS]]` and `[[DISKMAX]]` (HOSTSERVER-795).

- In the Admin Console you can now set the default for snapshot usage on a depot. This function is only available if it is supported by the Host Server which must be version 4.0 or later.

  This setting only affects whether snapshot are enabled or not for new spaces created in the depot. Existing spaces are unaffected by changing this setting.

- Admin Console: It is now possible to specify a list of "inbox listeners" on accounts that have an inbox. Inbox listeners receive an email notification when files are uploaded to the inbox (REGSERVER-1590).

- Added support for 2-factor authentication (2FA) based on a OTP (one-time password/PIN), send via email (TDCLIENT-3100). A new email template, "authentication-code", is used to send the OTP. This email contains links to the HTML templates: "login-confirmed" and "login-error".

  To enable 2FA you must set the client setting `enable-email-2fa` to either `true` or `web-only` in the `PRE_LOGIN_SETTINGS`.

  Alternatively, 2FA can be enabled at the account level, for all members of the account. If for some reason 2FA is not required for some individual users of an account then the account setting can be diable at in the edit user page (REGSERVER-1612). In this case it is always possible for the user to re-enable 2FA, in the TeamDrive client.

- The Registration Server also supports 2-factor authentication using the Google Authenticator App (REGSERVER-1598).

  The latest TeamDrive client allows you to enable email OTP or Google Authenticator based 2-factor authentication for a user.

  In the Admin Console you can to disable 2-factor authentication that has been enabled by the user.

- Admin Console: devices can now be filtered by "Client Type" (REGSERVER-1586).

- The server will now return an error when trying to register an Outlook Add-in, and the user already has `MAXIMUM_OUTLOOK_PLUGINS` (default is 1) registered, if the client specifies the `<uniquedevice>` tag (REGSERVER-1566). Without the tag, the server will delete an existing Add-in device, in order to make place for the new device, as before.

- Added support for Microsoft Teams (REGSERVER-1571):

  Two new templates have been added, "ref-file" and "ref-decompose". The first is an HTML template, and the second is a JSON template (which can be edited like other HTML templates).

  The "ref-file" template is returned in response to a "file reference" URL, which has the following form:

  https://<reg-server-domain>/yvva/ref/teamdrive/<file-global-id>?<search-args>

  The following search args are optional: size, space and file.

  The second is returned in response to a "decompose" HTML POST, which has the following URL:

  https://<reg-server-domain>/yvva/ref/decompose.json

  The POST body has Content-Type, "application/json".

  The file reference URL is generated by the TeamDrive client when a reference to a TeamDrive file is embedded in a Microsoft Teams communication (for example a chat).

  The decompose POST is done by the Microsoft Teams server, and is used to decompose the file reference URL. The response JSON is used to generate a "card" which is used to embed the file reference in the communication, in a branded form.

- TD2User.ClientSettings was set to nulls allowed, but in some databases this column may be NOT NULL, so NULL values will no longer be stored in the column (REGSERVER-1622).

- `API_USE_SSL_FOR_HOST` is now set to `True` by default.

RELEASE NOTES - VERSION 4.5

10.1 4.5.5 (2020-01-27)

- Fixed the collation sequence on the TD2APIRequests.User column (REGSERVER-1592).

- Admin Console: Only Host Servers that are owned by an account must be excluded from the list when creating a depot (REGSERVER-1589).

- Added `REDIRECT_FUSE` provider setting. The FUSE page should provide information about downloading and installing FUSE, which is used by the TeamDrive client to create a virtual drive for spaces (REGSERVER-1587).

- Added the "redirect-fuse" HTML template which is returned by default when the FUSE redirect is requested by the client, if `REDIRECT_FUSE` has not been set to a specific URL. In general, if an HTML template exists for a redirect, then it will be returned if the corresponding setting is empty. The search arguments on the URL are available as template variables.

- Added the `[[GETURL:<url>]]` template function which is substituted for the contents of the specified URL, for example: `[[GETURL:https://text.teamdrive.com/embedded-text.txt]]`. Template variable substitution is also performed on the retrieved text.

- Added new template conditional functionality. You can now compare a template variable to a specific value, using `[[IF:<name>=<value>]]`, `[[IFNOT:<name>=<value>]]` and `[[ELSEIF:<name>=<value>]]`, for example:

```
[[IF:PLATFORM=win]]
Platform: Windows!<br>
[[ELSEIF:PLATFORM=mac]]
Platform: MacOS!<br>
[[ELSEIF:PLATFORM]]
Unknown Platform: [[PLATFORM]]!<br>
[[ELSE:PLATFORM]]
No platform specified!<br>
[[ENDIF:PLATFORM]]
```

As before, if `=value` is not specified, then `IF` checks that the variable is not empty, and `IFNOT`, is true if the variable is empty.

- Admin Console: fixed a bug that caused confusing messages when devices were deleted (REGSERVER-1582).

- In the Admin Console it is now possible to switch a user to and from external authentication, as long as the super PIN is not enabled. Ensure that the user has a backup of their space keys, or has access to a TeamDrive client installation before making this change (REGSERVER-1556).

  It is also possible to enable and disable the super PIN for a user account, and to enabled and disable the user's Key Repository. Enable and disabling encryption on a user device is also possible. Note that TeamDrive client version 4.6.12 or later is required to support this functionality.

- Providers can now be removed when associated host servers are no longer accessable (REGSERVER-1555). When removing the provider, the depots are marked to be removed from the host server. A new auto-task: "Delete Depots on Host" will remove the marked depots from the host server in the background.

  If an error occurs when removing a depot, the error will be ignored if the host server of the depot has already been removed, or was never registered.

  In addition it is now possible to remove a host server in the Admin Console, even when the server still has existing depots. When removing a host server you can decide if you want to also delete the depots on the host server. If not, the reference to the depot will simply be removed from the Registration Server database.

- Fixed "Table 'td2reg.TD2AccountMember' doesn't exist" error when upgrading from version 3.5.5 (REGSERVER-1579).

- The "activateuser" call now activates the user and all devices that have not been activated (REGSERVER-1574). See activateuser_ref for details of all changes to the call.

- Admin Console: changing a user's email address, now requires confirmation from the user, who must click on an activation link send by email to the new email address (REGSERVER-1561). In addition, a notification is sent to the old email address that a change of email is in progress. If the email change is not confirmed within 2 hours the change is cancelled.

- The Email Queue is now prioritized. Notification emails send by the TeamDrive client are considered low priority, and will only be sent after all other emails have been sent (REGSERVER-1570). This is to ensure that regular emails are sent despite limits to the email send rate.

- Renamed setting SendGridIPList to EmailHookIPList. Added EmailHookURL.

- Updated default HTML templates to look better on small screens.

- Added `DEFAULT_AUTH_SERVICE_NAME` provider setting. If the provider is using an external authentication service that has not been upgraded, and therefore does not return it's external authentication service name (see default_auth_service_name).

- Certain errors when sending emails not result in the email "bounced" flag being set (REGSERVER-1567). This includes, the following error codes, in combination with the text strings in the error messages:

```
550, "invalid dns"
550, "mailbox unavailable"
550, "user unknown"
```

  If this the "bounced" flag is set, then emails will no longer be sent to the user.

  In the Admin Console a button is provided next to the "bounced" flag's checkbox to display the Email Log for the user. This includes any email error events that may have occurred during the email send process.

  A further button, "Send Test Email" is provided, which sends an email to the user with a link in which the user can confirm the validity of their email address. For this purpose the email template **confirm-email** and the HTML template **email-confirmed** have been added.

  When the user clicks on the link, the "bounced" flags is removed from the user's account, and all emails that failed to be sent are reset, and the Registration Server will attempt to send these emails again.

- The portal registration page was incorrectly placing the email address in the username field after an error occurred (REGSERVER-1585).

10.2 4.5.4 (2020-10-20)

- The setting `RedirectorProtocol`, now applies to all URL's returned by the Registration Server. This includes the portal pages, and the provider "REDIRECT" settings, and global "RedirectURL" settings (REGSERVER-1575).

Even if a setting such as `REDIRECT_FAQ` is set to a URL like: `http://my.server.org/faq.html`, if `RedirectorProtocol` is set to "https", then then a request for `REDIRECT_FAQ` will return `https://my.server.org/faq.html`.

- The "tdnslookup" API call new returns the Registration Server URL whenever it is returned by TDNS (REGSERVER-1565). In addition, the request tags `<email>`, and `<lookupboth>` have been added. See tdnslookup_ref for details.

- Removed the deprecated paths from the URL's used by the Registration Server and Host Server. This affects the values of the following global settings: `RegServerURL`, `MasterServerURL`, `LoadBalancerURL`, `PingURL` and `RegServerAPIURL`, and possibly also some of the provider settings which contain URL's (REGSERVER-1550).

  The URL's were modified by changing the path components: "pbas/p1_as", "pbas/td2as" and "pbas/td2api" to "yvva".

  In addition, the ".htm" extension for API access is deprecated, and ".xml" should be used.

  As now specified in API_Basics, the URL to access the the Registration Server's API is as follows:

  ```
  https://<domain>/yvva/api/api.xml?checksum=<md5>
  ```

- Fixed error in the "createdepot" API which caused it to incorrectly return the error: "Cannot create depot, not permitted by license" (REGSERVER-1549).

- Minor email template improvements (REGSERVER-1544).

- Added redirect URL to the Web Portal (REGSERVER-1546). Using the ".json" extension on the request will result in a JSON result, which contains the re-direct URL, for example:

  ```
  {
    "distributor":"PAL3",
    "language":"en",
    "language-arg":"en-GB,en-US;q=0.9,en;q=0.8",
    "page":"webportal",
    "resulttype":"ok",
    "url":"http://localhost:33000?dist=PAL3"
  }
  ```

  If an error occurs then the "resulttype" field is set to "exception":

  ```
  {
    "distributor":"EXT1",
    "errorcode":-30147,
    "errormessage":"No URL provided for requested page",
    "language":"en",
    "language-arg":"en-GB,en-US;q=0.9,en;q=0.8",
    "page":"webportal",
    "resulttype":"exception",
    "secondarycode":0,
    "test":"false"
  }
  ```

  If the request includes "test=true", then the Registration Server will attempt access the URL, and report an error if it fails:

  ```
  {
    "distributor":"PAL3",
    "errorcode":-12171,
    "errormessage":"Failed to connect to localhost port 33000: Connection refused
  ↪",
    "language":"en",
    "language-arg":"en-GB,en-US;q=0.9,en;q=0.8",
    "page":"webportal",
    "resulttype":"exception",
  ```

```
  "secondarycode":7,
  "test":"true",
  "url":"http://localhost:33000?dist=PAL3"
}
```

- All API functions that send emails now also accept a fields list, which is used to set custom template variables, for example:

```
<fields>
  <os>iOS</os>
  <contact-person>Joe Smith</contact-person>
  <description>This is a test...</description>
</fields>
```

This input will set the template variables as follows:

- `[[OS]]` = "iOS"
- `[[CONTACT-PERSON]]` = "Joe Smith"
- `[[DESCRIPTION]]` = "This is a test..."

Template variables set using the `<fields>` tag may may not overwrite default values used by the Registration Server. A warning will be logged if you attempt to do this.

- Added new auto-task: "Synchronise TDNS" (REGSERVER-1554). This task verifies the TDNS entry for all users. If the entry exists, but is owned by another Registration Server or Provider, then it sets a flag on the user, which is indicated by the test "No TDNS Entry" in the Admin Console.

  If TDNS cannot be updated when a user is created, this task performs the update later. These user's will be marked as "TDNS Update Pending" in the Admin Console. Note that only TDNS updates can be delayed, if adding the TDNS entry fails, then user creation will fail.

10.3 4.5.3 (2020-07-22)

- The Host Server of an account can now be specified to be owned by the account (REGSERVER-1532). In this case, managers of the account are automatically granted CREATE-DEPOT, EDIT-DEPOT-COST and DELETE-DEPOT rights to depots using the Host Server.

  Note that if a Host Server is not owned by the account, then it is no longer possible for a manager to set the size of the default depot.

  Specifying a Host Server for an account, without granting ownership means that the Host Server is just used to create the default depots of the users of that account.

- In the Admin Console, the Depot list now includes depots owned and in-use by account members. As a result, an account manager may not have access to all the depots listed because account managers only have access to the following depots:

  - Depots owned by the account
  - Depots that belong to the Host Server that is owned by the account

  Note that you also only have access to the history of a depot, if you have full access to the Host Server of the depot.

- Resetting the number of incorrect login attempts on the Admin Console did not apply to the Admin Console itself (REGSERVER-1533).

- The Admin Console now supports external authentication (REGSERVER-1530).

  In addition, you can login to the Admin Console using your email address as as user or provider (REGSERVER-1537). If an email is in use by both a provider and a user, then you will be required to select the required user from a drop-down list.

- It is now possible to create a user that uses external authentication on the Admin Console. In order to do this, the user's email must be associated with a external authentication service, or setting `AUTH_LOGIN_URL` and `VERIFY_AUTH_TOKEN_URL` must be set for the provider.

  Note that `USE_AUTH_SERVICE` must also be set to `True` in all cases.

  User's created on the Admin Console that use external authentication will have no "External Authentication ID". This value will be set the first time the user actually logs in (either using the Admin Console or the TeamDrive client or a Web Portal).

- If a user is using an named external authentication service, then this will be indicated in the Admin Console. Alternatively the user may now be explicitly marked as using an external authentication service or not. This is the case with all users created on the Admin Console.

  As before, for all other user's the standard authentication is the default, even if `USE_AUTH_SERVICE` is set to `True` and `AUTH_LOGIN_URL` and `VERIFY_AUTH_TOKEN_URL` values are provided. To ensure users of the provider use external authentication, `PRE_LOGIN_SETTINGS` must include (at least) the `enable-login=false` and `enable-web-login=true`, client settings.

  However, if a user is explicitly marked as either using an external authentication service or not it is not necessary to set the `PRE_LOGIN_SETTINGS` for the user. The Registration Server will automatically set the pre-login settings as required.

  In this case, the settings will override any settings that have been set using `PRE_LOGIN_SETTINGS` at the provider level. So, for example, it is possible to have one user use standard login while all other users of a provider as using external authentication.

- On the Admin Console you can change the authentication method of a user from external authentication to standard (Registration Server-based) authentication and back, provider the user has no devices, and no space keys in the key repository (REGSERVER-1529).

  If a user is changed to standard authentication then the user will also be de-activated, and an email sent to the user which provides a link to a page where the user can set a password for their account.

- Moved `ALLOW_WEB_PORTAL_ACCESS` setting to the `WEBPORTAL` settings group and moved `REG_NAME_COMPLEXITY` to the `LOGIN` settings group.

- Added "inbox-confirm-upload" and "inbox-upload-notification" email templates user by the inbox agent to notify users after files are uploaded to an inbox (REGSERVER-1538).

  The setting `MaxInboxEmailPerDay` has been added to limit the number of emails sent by an inbox user. This setting is used instead of the `MaxInboxEmailPerDay` setting used by regular users (see maxinboxmailperday for details).

10.4 4.5.2 (2020-06-25)

- Accounts now include a Super PIN Repository, which stores the Super PINs of all members of the account. To enable the Super PIN Repository the manager must create a "master password" which is then associated with the repository.

  When enabled, users will be prompted to login in order to upload their Recovery Data to the repository. After this point, if a user looses their password, a manager can send the user a Recovery Code, via email, by entering the master password.

  The user can then login using the Recovery Code as a once-off password. The Recovery Code is only valid for a limited time.

  Managers can also request that users of the account enable the Super PIN functionality. Users will then be prompted to login in order to enable the Super PIN.

  Note that the Super PIN functionality will be enabled automatically when using the Web Portal, or when local encryption is enabled (which requires allow-local-encryption=true).

  These functions require TeamDrive client version 4.6.9 or later.

- Added `EnableSuperPINRepository` Registration Server setting. If `False` (the default) the option to enable the Super PIN Repository, and the function to require account users enable the Super PIN are not available in the Admin Console.

- Set the new setting `ACCOUNT_RESTRICTIONS` to `super-pin-repo-pro-license-limit=5` to restrict the use of the Super PIN Repository feature to accounts with 5 or more professional licenses (REGSERVER-1490).

  This means that accounts with less professional license will not be able to enable the Super PIN Repository. By default, this Super PIN Repository is not restricted.

- Added support for SendGrid notifications (sendgrid.com). In order to receive notifications you must set the Registration Server setting `SendGridIPList` to a list of IP addresses that are the source of the notifications (REGSERVER-1517).

  The Registration Server will forward notifications to other Registration Servers if necessary (based in a TDNS lookup of the email address). This is done by the "Forward SendGrid Events" Auto Task. The IP address of forwarding Registration Servers must also by included in the `SendGridIPList` list.

  The "email bounced" flag will be set for user's with email address on which an error notification occurs. Emails are no longer sent to these addresses, and are marked in the email send queue as such. The errors on an email address can be cleared by removing the "email bounced" flag for the user in the Admin Console. When this is done, the Registration Server will attempt to send (retry) all outstanding emails to the user.

- The "Send Emails" auto task will now no longer attempt to send an email to a user who has the "email bounced" flag set. Instead the email will be marked with status "Bounced". Emails will also not be sent to email address that have a error registered in the Email Error log (which is written by SendGrid notifications).

  Resetting the email status will remove both the "email bounced" flag as well as the errors in the Email Error log, to ensure that the Registration Server really attempts to send the email.

- An account can now be set for a domain (REGSERVER-1522). If the domain is active then any user created with an email using the domain will automatically be added as a member of the account, and use the default license as specified by the account.

  This also applies to users that are automatically created due to an invitation. Note that such users will not be removed by the "Remove Auto Created Users" auto task (even if not activated), since they are members of an account.

  An error occurs if a new user is created for an account, with an email address that is reserved for another account. However, it is possible to move a user with a reserved email address to another account.

- Users that are created due to an invitation will only be displayed as guests of an account if they belong to the same provider (REGSERVER-1528).

- On login using a registered external authentication service, the Registration Server incorrectly required the `VERIFY_AUTH_TOKEN_URL` setting to be set, instead of using the Verify URL specified for the service (REGSERVER-1531).

10.4.1 Administration Console

- Combined the HTML and Email templates pages into one page called "Manage Templates".

- When sections are opened on the Edit Account page, they remain open after page reload.

- Resetting the status of an email in the email queue will cause all errors recorded for the email to be deleted, and the user's "email bounced" flag will also be set. This is to ensure that the Registration Server really tries to resend the email.

  If you wish to retry sending all emails to a particular email address, then go to the user of the email, and reset the "email bounced" flag.

10.5 4.5.1 (2020-05-12)

The most significant additions to the Registration Server in this version are the "Super PIN" functionality, and support for a new "on-boarding" process.

The Super PIN functionality is required to support client-side "local encryption". The Super PIN is activated for a user account when client-side local encryption is enabled by the TeamDrive client.

Local encryption adds an addition layer of security by protecting important data stored on then client. When using a Web Portal, local encryption is automatically enabled.

When the Super PIN is activated, the user may no longer set their password using a temporary password. If the user forgets their password they must either enter their Super PIN, or a Recovery Code, which is obtained using a "Recovery URL" (stored as a QR Code).

The user will be promoted by the TeamDrive client to export and store this information when the Super PIN functionality is enabled.

It is now possible for a provider to reserve domains and register external authentication services. Reserved domains must be activated by TeamDrive before they are used. When activated, domain reservation, prevents users of other providers from using email addresses with the reserved domains. In addition, external authentication services can be registered and then associated a reserved domain.

This information used by the TeamDrive client to locate the correction Registration Server during login and registration (required client 4.6.9), and the domain-based external authentication selection service.

Domain and service information is stored on TDNS (the TeamDrive Name Service), and can be managed using the Registration Admin Console.

The new on-boarding process involves the automatic creation of user accounts when a user is invited to a space (see the new `INVITATION_CREATES_USER` provider setting). The user is registered using the email specified in the invitation, and does not have a username (which means that `USER_IDENTIFICATION_METHOD` must be set to `email` or `default`, see user_identification_method). An email (the **reg-activationsetpassword** email template) is sent to the user with a link which allows the user to set a password for their user account. Activation is optional (see `ACTIVATE_ON_INVITATION`).

Note: if the user is not activated within a certain number of days, specified by `AUTO_CREATED_USER_TIMEOUT`, then the user will be automatically deleted. By default this is 60 days (see auto_created_user_timeout for details).

Users added by invitation, are listed as "guests" members of an account, if they are invited by a member of an account (REGSERVER-1504). Guest users can then be easily added to the account as member or manager.

After setting a password, the user may be provided with links to a Web Portal, or with a link to download the TeamDrive client. This can be done by configuring the relevent HTML templates, in particular **set-password-ok**. After login, using email and password the user will have access to the space to which they were invited.

On-boarding in this manner is the default when a user is created in the Admin Console. In other words, after creating a user in the Admin Console the user is sent an email with a link that allows the user to set their password, and after doing this proceed to a Web Portal or to download the TeamDrive client.

Users that are on-boarded automatically using this functionality can be granted a special license as specified by the `NEW_USER_LICENSE_FEATURES` provider setting.

In addition to these changes, it is now possible for a manager to invite an existing users to an account. Support is provided for this in the Registration Server API and the Admin Console. Invited users can be assigned a license which will be applied when the user accepts the invitation. Licenses assigned during invitation are counted to the usage of those license.

10.5.1 Registration Server Functionality

- Added support for registration of users without a username in the TeamDrice client.

- The `[[SUPERPIN]]` conditional template variable is now used in the **new-passwd** and **web-newpassword** templates to return an appropriate message to users that attempt to change their password using an old TeamDrive client, after the Super PIN has been activated (REGSERVER-1447).

  Conditional blocks in templates may now have the form `[[IF:<cond-var>]]` ... `[[ELSE:<cond-var>]]` ... `[[ENDIF:<cond-var>]]` (the `ELSE` markup tag is optional).

- Added `SUPERPIN_LOGIN_WITHOUT_ACTIVATION` setting which determines whether an activation email is sent the user when using a Super PIN to login to a new installation (REGSERVER-1451).

- Added `TEMP_PASSWORD_TIMEOUT` provider setting which determines the amount of time a temporary password valid (REGSERVER-1438). Default is 10 minutes.

- Added the `MAXIMUM_DEVICES_PER_USER` provider setting. The default value is zero, which means no limit. If set to another value the new "Deactivate/Activate Devices" auto task will enabled and disable devices as required to ensure that only the specified number of devices are active. The disabled devices are set to the "too many devices" status (REGSERVER-1399).

  The server always disables the least recently used devices. As a result, a device can be reenabled by simply running the TeamDrive client. However, it takes an average of 3 hours before a device is reenabled by the server.

  The device status is now sent to the TeamDrive client which should disable the GUI and stop synchronisation when the status of the devide is disabled.

  In this state the device will receive invitations, but will not send them to the client. This ensures that if the device is enabled, then the invitations will be sent to the client.

- Added new provider settings: `INVITATION_CREATES_USER`, `INVITATION_NEW_USER_PROVIDER`, `NEW_USER_LICENSE_FEATURES` and `ACTIVATE_ON_INVITATION` (see invitation_settings).

  These new settings belong to the `INVITATION` settings group, which was previously called the `REFERRAL` group.

- A new email template, **inv-newuser-invited**, has been added (see templates_for_client_actions). This template is used when a user is automatically registered by the new `INVITATION_CREATES_USER` feature (see above).

- A `[[DISCLAIMER]]` email template field has been added to all email templates to which it may apply (REGSERVER-1290). The disclaimer text can be set in the Admin Console under the account of the user. If no disclaimer text is available, then the `[[DISCLAIMER]]` field is removed, including the extra empty line that results from this.

- When the TeamDrive client requests the "default" depot, the Registration Server will now return any depot that the user has in use, if the user's "cloud depot" and default depot's do not exist.

- Added the **NoDepot** license feature which disables the creation of a default depot for new users (see default_free_feature).

- The server now retains the "default" depot status, when the default depot is removed from usage. As long as the user's default depot is either in-use or is owned by the user, it retains the "default depot" status for the user.

  In addition, the default depot status will be restored to a depot, if it is removed from the user (both in-use and ownership), and then added again, as long as the user is not given a different default depot.

  As long as the user has a default depot, no new default depot will be created.

  In previous version of the server, removing the usage of a the default depot from a user would cause a new default depot to be created (assuming `HAS_DEFAULT_DEPOT` is set to `True`), as soon as a TeamDrive client calls the Registration Server.

- The `ALLOWED_DIST_CODES` is now also applied on user registration. However, this is only done when the client sends the distributor code from the DISTRIBUTOR file (REGSERVER-1402). This required client version 4.6.8 or later.

For login, clients before this version were not sending the distributor code from the DISTRIBUTOR file if users entered a different code in the Provider panel. In this case the server was checking the entered distributor code.

In the case of external authentication all client version send the correct distributor code (the distributor code from the DISTRIBUTOR file).

- Added a checkbox to accept the "Terms of Service" on the portal registration page (template: **portal-register**), and the set password activation page (template: **set-password**) (REGSERVER-1450).

  Added the `REDIRECT_TERMS` provider setting which specifies the "Terms of Service" page. A reference to this page is used in the **portal-register** and **set-password** HTML templates.

- Added depot template: **depot-warning**, **depot-cancelled**, **depot-reduction** and **depot-reduced** which are used by the Host Server to inform the managers and owners when depots usage has exceeded the required limit (see :ref:mail_templates_for_depots).

  The template **depot-traffic** is used to inform managers and owners about critical levels of network traffic usage.

- The `USE_SENDER_EMAIL` setting has been deprecated, and replaced by `FROM_EMAIL_OPTIONS` (see from_email_options). The `FROM_EMAIL_OPTIONS` value is set by default so that the behaviour of the Registration Server in this regard will not change (REGSERVER-1452).

- Added the `EnableDomainSupport` (**TDNS**) setting. When set to `True` this setting enables the support for the reservation of domains and registration of service by a provider.

- Added the `PREVIOUSLY_UNNAMED_SERVICES` (**AUTHSERVICE**) provider setting. This setting must be used when upgrading an existing external authentication service to a "named" service. "Named" services are services registered globally on TDNS (This is done using the Admin Console).

- Changes to provider settings:

    - Renamed settings group: `LOGIN` to `ADMINCONSOLE`

    - Renamed settings group: `ACTIVATION` to `LOGIN`

    - Rename setting `ALLOW_LOGIN_WITHOUT_EMAI` to `LOGIN_WITHOUT_ACTIVATION`

    - The following setting have been moved from `CLIENT` to new `LOGIN` group: `ALLOWED_DIST_CODES`, `PRE_LOGIN_SETTINGS`, `LOGIN_WITHOUT_ACTIVATION`, `ALLOW_NEW_REGISTRATION`, `ALLOW_MAGIC_USERNAMES`, `ALLOW_WEB_PORTAL_ACCESS`, `ALLOWED_LOGIN_ATTEMPTS`, `FAILED_LOGIN_TIMER`, `SUPERPIN_LOGIN_WITHOUT_ACTIVATION`, `TEMP_PASSWORD_TIMEOUT` and `USER_IDENTIFICATION_METHOD`.

- Documentation: updated screenshots in section **Using the Administration Console**

- The email template: **reg-registrationnotify**, which was previously unused, is now sent after a user sets a password using the link sent in the **activationsetpassword** email. The `[[PASSWORD-SET]]` template variable is also set to `true` in this case.

- Added global setting: `EmailSendRate`, which determines the maximum rate at which emails will be sent. The default is "0", which means unlimited. Any other value is the number of emails that may be sent per minute.

- Added the `SPACE_SIZE_LIMIT` provider setting which restricts the size of spaces for users with a restricted or non-professional license (REGSERVER-1502).

- Fixed TD2OwnerMetaHistory does not exist error when updating from Registration Server 4.0.1 (REGSERVER-1520).

10.5.2 Registration Server API

- The "registeruser" API call now supports the option `<nodepot>` which, when set to `true`, prevents the assignment, or creation of a default depot for the user. In addition, a depot may be assigned to a user on

registration using the appropriate tags (REGSERVER-1326).

- The new "inviteusertoaccount" API call can be used to invite a user to an account via email. The call will send the "account-manager-invitation" or "account-member-invitation" email template depending on the type of invitation. The user is provided with links in the email to either accept or reject the invitation (REGSERVER-1289).

  The function to invite users to an account is also available in the Admin Console.

- The "tdnslookup" API call will now work, even when the Registration Server is not connected to TDNS. In this case the call will return information from the local database (REGSERVER-1410).

- Added a `<messagetext>` tag to the "registeruser" API call. This tag specifies a message that can be placed in the email sent by the call use the `[[MESSAGE-TEXT]]` template variable.

  Also added a `<sendcc>` tag (default is `false`). When set to `true` the Registration Server will "CC" the email sent to the user, to the caller (set by the `<changeuser>` tag).

- All email addresses must now have the form: x@x.x, where x is one or more characters. White space, ',' and ';' are not allowed (REGSERVER-1471).

- Added "getsettings" API call. A list of Registration Server and provider settings can be specified, using the `<settings>` tag. This tag is also supported by the "getuserdata" and "getaccountdata" API calls (REGSERVER-1511).

10.5.3 Administration Console

- The Admin Console will indicate if the Super PIN functionality has been enabled, and also allows managers to disable the Super PIN functionality, which will delete the user's Super PIN.

  Only delete the user's Super PIN if the user has lost both their Super PIN and their password. After removal, the user may then set their password using the temporary password functionality, however previously local encrypted client installations will not be accessable, including Web Portal containers.

  In addition, the user will loose access to space keys stored in the Registration Server's key repository.

- It is now possible to specify a banner and a footer for the Web Portal user interface, for all users of an account, see Customize Web Portal under Extended Settings (REGSERVER-1433).

- Host Servers can now be assigned to accounts (REGSERVER-1299). In this case, the account Host Server overrides the Host Server specified by the `HOST_SERVER_NAME` provider setting. In addition,account managers are able to set the following parameters at the account level:

  Default depot: Determine whether members of the account have a default depot. This setting, can override the provider level settings `PROVIDER_DEPOT` and `HAS_DEFAULT_DEPOT`.

  Storage size: This is the storage size of default depots created. This setting override the `HOST_DEPOT_SIZE` provider setting.

  Traffic limit: This sets the traffic limit of default depots created. This setting override the `HOST_TRAFFIC_SIZE` provider setting.

  See hostserver_settings for more details on these settings.

- When creating a user, a checkbox has been added which allows you to prevent the creation or assignment of a default depot.

  Note that a default depot will nevertheless be created with the user's first client registration, unless:

  - the user belongs to an account with a default account depot,

  - the user's account has a host server and the *Default depot* account level setting to prevents the creation of a depot,

  - the user's license has the **NoDepot** feature.

- The "Purchase License/Depot" buttons now open a new browser page or tab (REGSERVER-1281).

- UI improvement: the background of the paging control is now transparent.

- The user's account depot is now included in the user's list of depots (REGSERVER-1419).

- Added domain and external authentication service management. This is only enabled when the setting `EnableDomainSupport` to `True` (by default `False`). This functionality requires TDNS 1.9.11.

- Depot lists now have separate columns for Storage/Traffic limit/used and can be individually sorted (REGSERVER-1472).

- Added a new `InboxUploadForm` account-level setting, which can be used to configure a form that users must fill out before uploading files to an inbox

10.5.4 External Authentication

- The domain-based selection of the external authentication service (`domain` directory) now uses the reserved domain information, and the associated authentication services to direct user's to the correct external authentication service.

  The other external authentication services have been updated to check the configuration using the information stored centrally (TDNS 1.9.11).

  Check the example configuration files for information on the new settings, and changes you need to make to upgrade services.

RELEASE NOTES - VERSION 4.1

11.1 4.1.4 (2020-02-19)

- Changed collation of all emails columns to case-insensitive (REGSERVER-1480).

  All input email are converted to lower-case: in import, and email addresses from external authentication services (REGSERVER-1479).

- Fixed format of output on "Download Client Log Files" page. Long "words" are are now wrapped as required (REGSERVER-1481).

11.2 4.1.3 (2020-01-16)

- Added the `RedirectorProtocol` server setting which can be used to specify the protocol of the "redirect URL" (REGSERVER-1473).

- Fixed a problem that prevented update notifications to the TeamDrive clients from working (REGSERVER-1474).

  Added a new provider setting: `UPDATE_TEST_VERSION` which can be used to set the version to be used testing an update notification (see update_test_version).

- Admin Console: fixed problem with Provider Codes that consist only of digits (REGSERVER-1028). This caused various problems, for example, it was not possible to create an inbox.

- Fixed a problem that caused the "Expire Licenses" auto-task to fail when a license belonging to an inbox user expired. The error generated was "The inbox user must have a license with the inbox feature" (REGSERVER-1466)

- Fixed the "Lock wait timeout exceeded" errors, that occurred due to an UPDATE to the TD2Message that was performing a table scan (REGSERVER-1465).

- Change required for compatible with yvva 1.5.2.

11.3 4.1.2 (2019-09-16)

- Added documentation for web portal settings. The setting `API_WEB_PORTAL_IP` has been moved to the WEBPORTAL settings section.

- Admin Console: the Manage Licenses page now includes the option to search for "Inbox" licenses (REGSERVER-1436).

- Admin Console: the License Report page was not working due to an error when retrieving the report list from the database (REGSERVER-1444).

- When moving spaces to another depot, it was possible to move a space to a depot to which the account manager did not have access (REGSERVER-1442).

- Corrected email template usage when forcing re-login or reseting a user's password. The email templates sent for these actions were reversed.

- Admin Console: corrected "Force Re-Login/Invalidate Password" functionality in the case where `USE_AUTH_SERVICE` is set to `True`, but `AUTH_LOGIN_URL` remains blank. In the case, external authentication is not being used (since `AUTH_LOGIN_URL` is required for external authentication, instead the Registration Server Portal login pages have been activated.

  As a result, the "Force Re-Login" button should read "Invalidate Password" in this case. This is due to the fact that invalidating the user's password has a different effect, depending on whether external authentication is being used or not.

  This will be changed in Registration Server 4.5, where only the "Force Re-Login" functionality will be provided, in both cases, i.e. whether external authentication is being used or not.

11.4 4.1.1 (2019-06-19)

- Admin Console: fixed crash when logging in with email address, instead of username.

- [account] in redirector URL will now be replaced with blank, if the user has no account.

- CSV import results displayed on the "CSV User Imports" page in the Admin Console, can now be viewed in the browser rather than downloaded directly. A success and/or error file is only available for viewing if at least one success or failure occurred during the import (REGSERVER-1398).

- In the Admin Console, the selection method used in dialogs has changed. If multi-select is allowed, then the checkbox are used to indicated which items have been selected. In the single selected case, radio buttons indicate which item has been selected.

  Currently adding members to an account and users to a license allow multi-select. A "Select All" button is also available in these cases. An extra dialog, confirming your selection is presented when adding more than 15 users (REGSERVER-1397/REGSERVER-1418).

  In addition, the paging section above search results has been improved to provide more options. You can now jump to the beginning or end of the result, and also move ahead or back a number of pages by clicking "..." (which is only available when sufficient pages are available).

- Since version 4.0 the Registration Server is compatible with PHP 7.2 / 7.3. However, the Admin Console may have problems after upgrading PHP due to the fact that the "mysql" extension has been removed from PHP 7 and later. Documentation has been added to help users solve this problem: using_php_72.

- Added Web Access to the account-level client settings. The default values of the account-level client settings are now determined by the provider setting `CLIENT_SETTINGS` value (REGSERVER-1401).

- In the Admin Console, on the Depots page, you can now search for the depot owner's email in addition to the owner's username. The owners email address is also display in the list (REGSERVER-1411).

- Added "web-user-deleted" email templated which is sent to the user after the user's account has been deleted (REGSERVER-1405).

- On login to the Admin Console the username is now case-insensitive (REGSERVER-1406).

- Template names are now displayed in the Admin Consoles email queue (REGSERVER-1409).

- An "inbox" type license can now be created in the Admin Console (REGSERVER-1414).

11.4.1 Registration Server API

- Emails sent by the API may now include the following email template variables: `[ORIGIN-$USERNAME]`, `[ORIGIN-USERNAME]`, `[ORIGIN-EMAIL]` and `[ORIGIN-USERNAME-AND-EMAIL]`. These variable will be empty unless the `<changeuser>` tag has been set in the API call. The "web-activationsetpassword" template has been changed to include a reference to the originator of the email, if the `<changeuser>` tag is set (REGSERVER-1413).

- Added an option to change a user's account in the Admin Console (REGSERVER-1407). This function is supported by the addition of the `<removemembership>` tag to the "addusertoaccount" API call.

- Added documentation for the "updateuser" API call (REGSERVER-1408).

11.5 4.1.0 (2019-04-18)

- Fixed a error that prevented users from being removed, after the Provider was deleted. The problem occurred after the Provider was removed from TDNS (which is required in order to delete a Provider).

- Added the provider setting required to connect the Registration Server to a web portal API. This API can now be used to create an inbox service for an account. The user which is hosting the inbox needs a license with the **inbox** feature.

- Removed the banner management page in the Admin Console. The Banner administration of banners. This includes the **Banner** feature bit used by licenses. This feature is still displayed for licenses with this feature bit, but the feature can no longer be set.

- The **Personal** license feature is no longer supported by version 4.1 of the Registration Server. This feature was only used by TeamDrive 3 clients. Users must now use the **Professional** license feature instead of the **Personal** license feature.

- Changes to provider settings are now recording in a change history. The change history of provider settings can be viewed in the Admin Console.

RELEASE NOTES - VERSION 4.0

12.1 4.0.1 (2019-03-29)

- [account] can now be used in the help redirect URL

- Several bug fixes and improvements to version 4.0.0

12.2 4.0.0 (2018-09-19)

12.2.1 Registration Server Functionality

- Removed the provider setting: `HOST_SERVER_URL`. This is no longer required, the Host Server to be used is specified by `HOST_SERVER_NAME`.

- Added support for accounts (REGSERVER-1229). Accounts belong to a provider and include a number of users, groups, depots and licenses. An account is administered by a number of managers. A user can only belong to one account, but may be manager of a number of accounts. Accounts are explained in the the new chapter account concept and in a adminconsole chapter admin_console_accounts.

- Added support for groups (REGSERVER-1196). Users can be invited to join a group, which is administrated by a Group Manager. The user receives an email, which contains a link for joining the group and another link for rejecting the invitation. Users that have rejected invitation 3 or more times can no longer be invited to a group.

  Users can only belong to one group, so when the join a group they are automatically removed from any other group.

  The Manager of a group can assign a license and a Host Server Depot to the group. The group license and the group Depot are used by all members of the group, and have priority over the user's default license and Depot.

  Please notice that group functionality is not available in the 4.0 release of the Admin Console. This will be added in version 4.1.

- The setting `UserNameCaseInsensitive` has been deprecated. All Registration Servers now use case-insensitive usernames. The TDNS entries will be automatically updated of your server had `UserNameCaseInsensitive` set to `False`.

- The Registration Server now uses a new mechanism to synchronise the Depot usage with the Host Server and the TeamDrive Client. The mechanism ensures that changes to Depot usage on the Registration Server is always reflected in the Depot list in the TeamDrive Client, and in the Depot access list on the Host Server.

  Previously, it was possible that there were differences in the Depot configuration for a user between the TeamDrive Client, Registration Server and Host Server. This was due to a number of factors:

  - The Host Server access list for a Depot was previously not updated by the Registration Server API.

– By setting `<sendtoclient>false</sendtoclient>` in an API call the user could previously specify that the changes to the Depot configuration of a user are not sent to the TeamDrive Client. This tag is now deprecated (see below).

– If a TeamDrive Client device was not in use for a long time it was possible that changes to the Depot configuration were lost.

- The following characters are never allowed in usernames: '$', ';', ',', '@' '"' and the single quote ("'").

- When setting up a Registration Server, the server name is not allowed to contain a ".", or any spaces. The server domain must be valid, and contain at least one ".".

- Added the `DEFAULT_ACCOUNT_FEATURE` provider setting which is identical to the `LICENSE_FREE_FEATURE` but applies to users that belong to an account (REGSERVER-1253). If `DEFAULT_ACCOUNT_FEATURE` is empty then the Admin Console will not allow managers to create a new license when adding a user.

- Added the `ACTIVE_SPACES_LIMIT` provider setting which determines the maximum active spaces for users with a restricted license (REGSERVER-1257).

- Improved security by defining a maximum login attempt and interval (see loginmaxattempts and allowed_login_attempts)

- Added the `PROVIDER_LOGIN_ID` setting which is a list of IP addresses of users that may login with provider level or higher privileges (REGSERVER-1333). On upgrade this setting is set to the value of the `LOGIN_IP` value, if this value is not empty. Providers that wish to allow normal users or account managers to access the Admin Console from any IP address must set `LOGIN_IP` to empty.

- License that expire are now also valid on the "Valid Until" date (REGSERVER-1389).

- The Registration Server was sending an incorrect result to the client when a disabled user requested a temporary password (REGSERVER-1237).

- Removed deprecated auto task: "Move Store Forward Messages".

- Fixed possible deadlock involving the Devices table and the "Delete Client IPs" auto task (REGSERVER-1464).

12.2.2 Registration Server API

- The output parameter `<number>` in the searchuser_ref API call, and the getlicensedata_ref API call has been deprecated and will be removed in a future version. Use the license key number now returned in the `<licensekey>` tag.

- The Host Server API URLs returned by the API will now begin with "https://", if the provider setting `API_USE_SSL_FOR_HOST` is set to `true`.

- Added "createdepot" API call.

- Added API calls to support account functionality: "createaccount", "deleteaccount", "addusertoaccount", "removeuserfromaccount", "assignaccounttolicense", "removeaccountfromlicense", "setdepotaccount", "removedepotaccount", "setgroupaccount", "removegroupaccount", and "getaccountdata".

- Added API calls to support group functionality: "creategroup", "deletegroup", "inviteusertogroup", "removeuserfromgroup", "setgrouplicense", "removegrouplicense", "setgroupdepot", "removegroupdepot", "userjoinedgroup", "setgroupclientsettings", and "getgroupdata".

- A number of API calls now also return group related information: "loginuser", "searchuser", "getuserdata", "getlicensedata", "getdefaultdepotdata".

  The "getuserdata" call now return account and group information by default. Set the input tags: `<includeaccounts>` and `<includegroups>` to `false` in order to exclude this information. This call also returns license currently assigned to the user in the `<license>` block in the `<userdata>` block. If `<includegroups>` is `true` then this is the group license if the user belongs to a group with a license.

The calls: "loginuser" and "getlicensedata" return the user's group information by default. Set the input tag: `<includegroup>` to `false` in order to exclude the group information.

The calls "searchuser" and "getdefaultdepotdata" do not include group related data by default. In this case you must explicitly set `<includegroup>` to `true` to receive this information.

- The `<depot>` block returned by the calls "getuserdata" and "getdefaultdepotdata" calls includes a number of new tags:

    - `<globalid>` is the global identifier of the depot.

    - `<iscloud>` is set to `true` if the depot is the user's default cloud storage.

    - `<isaccount>` is set to `true` if the depot is set on the account level.

    - `<isgroup>` is set to `true` if the depot belongs to the user's group.

- When returning information about licenses (`<license>` tag) the `<isgroup>` tag is now included. This tag is set to `true` if the license belongs to the user's group.

- In the "registeruser" API call new supports a number of new tags: `<accountkey>`, `<accountreference>`, `<grouppreference>`, `<featurevalue>`, `<clientsettings>`, `<activate>` and `<sendmail>` (see registeruser_ref for details).

- The `<sendtoclient>` tag is the API calls: "setdepotforuser" and "removedepotfromuser" is deprecated. If present, the tag is now ignored by the Registration Server. Changes to the usage of a Depot are now always sent to the TeamDrive Client.

- Added API calls: "syncdepotdata" and "getdepotdata".

- API calls that send emails now support the `<sendmail>` tag. This allows the caller to override the `API/API_SEND_EMAIL` setting, to determine whether an email is sent or not.

- The `<changeuser>` tag is used to specified the username of the user that is making changes to depots.

- Added `<setpassword>` tag to the "registeruser" API call. When set to `true` (default is `false`), this will send an email using the **web-activationsetpassword** template to the user. This email contains a link to the **set-password** HTML template, which allows the user to set his password, and activate his user account (REGSERVER-1320).

12.2.3 Administration Console

- Rearranged the menu of the Admin Console and extended the user right levels for viewing, creating, editing and deleting objects (see admin_console_user_rights).

- Restricted the view presented by the Admin Console to only those pages that a user has the right to view. Any TeamDrive user with login privileges may login to the Admin Console, and view and manage their resources.

- Added a global provider drop-down menu, so that users with access to more than one provider can select a Default Provider for all operations.

- Added account management.

- Added new categories for Registration Server settings: API, Proxy, RedirectURL and TDNS (REGSERVER-1227).

- Added an automatic redirect to the login page when the login session expires.

- If a Depot is deleted and then undeleted on the Host Server, an "Undelete Depot" button is available in the Admin Console to make the depot available again.

- The "Force Re-Login" function is not always available on the Edit User page. Previously this function was only available if external authentication was in use (REGSERVER-1469).

    The function to "Invalidate Password" is available, in addition if the Super PIN functionality is not enabled, and external authentication is not in use.

"Force Re-Login" is also available in the Manage Users page, where it effects all user in the selection.

Forcing a re-login will require the user to login again on all installed devices.

12.2.4 External Authentication

- All external authentication services (except `vasco`) now use the same functions to evaluate input and generate the authentication token.

  The services can now be deployed by following the instructions in the `*_config.php.example` page to create a configuration file, and then customising the HTML in the `*_login.php` page.

  However, be careful to preserve the PHP dynamic tags in these files, which all have the form: `<?= ....  ?>` and `<?php ...   ?>`.

  Future upgrades will be done (in most cases) by simply replacing all files accept `*_login.php` and `*_lconfig.php`.

  Note that the `auth` directory is now used by all authentication services, and `auth\vendor` is used by the Google and Azure services.

- Added support for Google and Azure OAuth2 external authentication. To use these services:

  - follow the instructions in the `*_config.php.example` page to create a configuration file, and
  - customise the `*_login.php` page to suite your purpose.

- Added domain-based selection of the external authentication service (`domain` directory). The initial page of this service requests the user's email address. Based on the domain of the email address the user is forwarded to the appropriate authentication service.

  The mapping from domains to authentication services is configured in the `dom_config.ini` files (see `dom_config.ini.example` for notes on how to created this file.

  Using this service, the users of a single provider can use various authentication services. If this is the case, then each authentication service must be given a unique name, which is used as a prefix to the external authentication (External Auth. ID) of the user to avoid duplicate IDs.

- The LDAP external authentication has been updated to evaluate options from various clients, including: the TeamDrive client, the Web Portal, and the TeamDrive agent.

  As a result, the `ldap_login.php` page can be used in all cases, and the `ldap_web_login.php` and `ldap_agent_login.php` pages, are no longer needed, and have been removed.

  Follow the instructions in the `ldap_config.php.example` page and read the information about the LDAP encryption parameters (see ldap_encryption_parameters) when upgrading from an older version of the LDAP authentication service.

  In addition, the `$provider_code` setting has been deprecated. When upgrading, copy the value of this variable to the position of the first URL in `$allowed_origins` (see ldap_portal_parameters for details).

RELEASE NOTES - VERSION 3.X

13.1 Change Log - Version 3.6

13.1.1 3.6.8 (2018-02-07)

- Added new Provider EMAIL settings which override the global Registration Server settings (REGSERVER-1226). This makes it possible to specify the SMTP Server to be used to send emails at the Provider level. Support for sending mails using SSL/TLS by prepending the protocol "smtps://" (only supported on CentOS 7 systems due to dependencies of required curl functionality) and authentification with an username and password was added:

    - `SMTP_SERVER`: The SMTP Mail Server address (host name), if empty the `SMTPServer` global setting value will be used.

    - `SMTP_SERVER_TIMEOUT`: the Timeout in seconds when waiting for the SMTP Mail Server, if empty the `SMTPServerTimeOut` global setting value will be used.

    - `SENDER_HOST`: Host name of the email originator. If empty the `MailSenderHost` global setting value will be used.

    - `SMTP_SERVER_USER`: Username for smtp authentification.

    - `SMTP_SERVER_PASSWORD`: Password for smtp authentification.

- Version 3.6.8 requires YVVA runtime version 1.4.5.

13.1.2 3.6.7 (2017-11-06)

- Fixed a crash when sending email due to incorrect SQL statement (REGSERVER-1223).

- Fixed sending of "Future Device" messages which are used to sent invitations to users that do not yet have a device.

- Documentations has been changed to conform to the new TeamDrive CI.

- Some devices were not receiving invitations because the "Demo" flag was set. This flag is now ignored when invitations are sent.

- Replaced TeamDrive logo and colors

- Improved logging of errors when connected to TDNS, Host Servers and other Registration Servers. If an unexpected reply is received, the server will dump the first 420 characters of the response to the log, in order to help debugging proxy related connection errors.

    During setup of a Registration Server details of incorrect results are provided when you press the "Error Details" button. If the server receives an unexpected result when trying to contact other servers then the first 420 characters are display in the dialog window.

- External Auth Service: corrected generation of user secret. Added the "alt user secret" to enable transition to a new method for generating user secrets.

- Added the `SETUP-2FA` conditional variable for the Portal Pages (html and email templates/html templates/portal pages) which is set to "true" if the user selects to setup 2-Factor Authentication during login.

  The default **portal-login** page has been altered to use the variable to indicated if the user has selected to setup 2-Factor Authentication or not.

- Fixed a bug in the Web-based setup of the Registration Server that caused a "Unknown attribute: 'REG_SERVER_BUILD'" exception (REGSERVER-1214).

- Registration Setup as Standalone or Master server now requires as "Setup Code". This is required in order to prevent the accidental installation of a Registration Server that can only be accessed using a customised TeamDrive Client. A Setup Code can be obtained from support@teamdrive.com, but requires an agreement for the deployment of a "white-label" TeamDrive Client.

- Fixed a bug in the Registration Server Setup that prevented the installation of a server when using a proxy to access the Master Registration Server.

- Version 3.6.7 requires YVVA runtime version 1.4.4.

13.1.3 3.6.6 (2017-08-04)

- Fixed an exception that occurred when attempting to wipe a device (REGSERVER-1210).

- Fixed a error that occurred when removing a device installation on the client of a user had already been removed (REGSERVER-1211).

13.1.4 3.6.5 (2017-07-13)

- The Reg Server now handles "store forward" invitations sent by the TeamDrive client, when a user has no active devices (because all devices have been inactive for longer than `InviteOldDevicesPeriodActive`). Previously this only worked if the user had no devices (which can happen if the user was created via the API).

  The first device that becomes active after this point, whether it is a new device or an old device that was re-activated will receive the invitation (REGSERVER-1200).

- API call "removelicense" was not working due to a problem with NULL values (REGSERVER-1197).

- Fixed activation of users and devices via the adminconsole (REGSERVER-1199)

- Uploaded Client log files are now stored in a table created to store all large binary values (TD2LargeBinaries). This prevents a slowdown of access to the TD2BlobData table (REGSERVER-1202).

  On upgrade the log files will be moved from one table to the other. This can take some time.

- Added a new covering index to the TD2BlobData table that includes all columns used to search the table. This will allow the server to avoid reading the entire row during a search.

  The column TD2BlobData.Extension has been shortened to 40 bytes (ascii) and the columns TD2BlobData.SourceChecksum has been removed because it is no longer used (REGSERVER-1201).

- Optimised the queries used in the CSV page in the Admin Console, and fixed a bug that left the 'error' and 'success' file in the database when a CSV file was deleted

- Fixed a bug in the "searchuser" API call. When `<showdevice>` was `false`, the `<total>` was incorrectly set to 0 (REGSERVER-1204).

- Fixed a bug when deleting an user and his depots: If user is not the owner of a depot he must be removed from the depot as an user instead of deleting the depot (REGSERVER-1205).

13.1.5 3.6.4 (2017-05-04)

- Fixed crash in regserverdistribution (REGSERVER-1186).

- Fixed an error that resulted in the `<licensekey>` tag missing from a number of API calls that returned license data (REGSERVER-1187).

- Fixed setting a client update notification using the admin console (REGSERVER-1189).

- The `<intresult>` tag was missing from the result of the "createlicensewithoutuser" API call.

- Several small fixes for the admin console: improved user search speed and added case insensitive search for usernames, fixed regular expression for magic usernames with an ID > 9999, improved client logs download page

- Added hint how to start the apache service after mysql (see enabling service autostart)

- Fixed sending API calls for different provider using the same IP (REGSERVER-1194).

- Fixed license change history in the adminconsole in cases where the 'license created' entry was missing from TD2TicketChanges (REGSERVER-1188)

- Require entry of a confirmation text when deleting licenses (previously this was only required if the license was created in an external system) (REGSERVER-1193)

- The default provider can now view uploaded log files for all providers at once (REGSERVER-1190)

- Installation: set `max_allowed_packet=32M` in order to support the upload of large client log files (REGSERVER-1192)

- Fixed a number of problems with the API functions "searchuser" (REGSERVER-1195): It is now possible to retrieve all users by not specifying any search condition. Previously this caused error -30116.

  The result tags `<current>`, `<total>` and `<maximum>` now refer to the number of users, regardless of whether devices are included in the result or not. Previously these tags referred to the number of devices, when `<showdevice>` was set to `true`.

  Previously it was possible that devices for the last user returned were missing, if the maximum rows (`<total>` value) was exceeded when including devices in the result.

  When you specify a `<startid>` value, the `<total>` value returned now consistently refers to the total number of users with an ID greater than the specified value.

  This means that, in general, if the `<total>` value is greater than the `<current>` value, then the caller knows that more user records are available with the input parameters.

  Previously to version 3.6.4 the result `<total>` was not consistant if `<showdevice>` was set to `true` and should not be used.

- Increased TD2BlobData.Data column size to allow 50 MB uploaded log files (REGSERVER-1191).

- Increased TD2Depots.ReposDoc column size to 4000 characters required to store larger repository files (REGSERVER-1185).

13.1.6 3.6.3 (2017-03-22)

- Added Provider setting `EMAIL/IGNORE_TEMPLATES_LIST`, which contains a list of email templates. Emails will not be sent with the templates specified in this list (REGSERVER-1184).

- Added the `UsePrecedenceBulk` setting which determines whether the "Precedence: bulk" header should be added to outgoing emails (REGSERVER-1182).

- The API documentation now includes a section on the changes to the API based on the Registration Server version. All changes since version 3.5.0 are noted in the documentation of the API calls (REGSERVER-1173).

- Fixed a bug removing users from a depot who had been added to the depot when it was created (REGSERVER-1159)

- Several minor changes and fixes in the Admin Console (fixed spelling License -> Licence, moved "change user license" on the edit user page from device block to user block, fixed 2 SQL statements, added username to client logs download page)

- Added new clients settings `allow-webaccess-by-default` and `enable-space-webaccess` in the documentation

Registration Server API

- The "activatelicense" and "deactivatelicense" API calls no longer return error -30210 (REGSERVER-1177). If the license is already in the state set, then the call is ignored.

- Specifying a user in the "removeuserfromlicense" API call is now optional. If specified, then the user must be the owner of the license or a "Unknown license" error will be returned (REGSERVER-1178).

- Remove the API version number (1.0.006, 1.0.007, etc.) The Registration Server version number is now used to determine when API changes have been made. All API calls now return the `<regversion>` tag which contains the version number of the server (REGSERVER-1173).

- "getdefaultlicense" API call: removed the exception that returned the features of the license in use if it was higher than the features of the default license.

- Added a `<licensereference>` tag to the input parameters of the "loginuser" call. This tag is used if a default license is created for the user. This is only done if the user has no default license, and the Provider setting `DEFAULT_LICENSEKEY` is empty.

- The new reference should now be specified using the `<newlicensereference>` tag in the "setlicensereference" API call.

- Added an optional `<password>` tag to the "removeuser" API call input data.

- The `<featurevalue>` tag value may now also be specified as an integer in the "createlicense", "createlicensewithoutuser", "upgradelicense" and "downgradelicense" API calls.

- Added the `<licensereference>` tag to the `<license>` block in reply of the "getusedlicense" API call.

- Added the `<licensereference>` tag to the `<user>` and the `<device>` block in reply of the "searchuser" API call.

13.1.7 3.6.2 (2017-02-01)

- The Registration Server Portal Pages (see html and email templates/html templates/portal pages) will no longer allow login of users that have previously logged in using an external authentication service (REGSERVER-1180).

- If a user is using external authentication then the server will no longer allow the user to change his password. The server now returns an error -24907: Permission denied, when the TeamDrive client attempts to perform on of these functions (REGSERVER-1179).

- External authentication now first checks wether the authentication token is an internal token used by the portal pages. If not, it checks the URL specified by the `AUTH_LOGIN_URL` setting (REGSERVER-1181).

- Added Provider setting `USER_IDENTIFICATION_METHOD` (REGSERVER-1171). This setting determines how users will be identified (see user_identification_method). `USER_IDENTIFICATION_METHOD` replaces the Provider setting `USE_EMAIL_AS_REFERENCE`, which has been removed.

- Fixed a bug that caused the switch-distributor function to always create a new depot and license even when the checkboxes where not selected (REGSERVER-1170).

- Added new server setting PrivacyURL and Provider redirect page REDIRECT_PRIVACY

- Added fields to select an existing license when creating a new user in the adminconsole (REGSERVER-1166)

- Can now filter the list of devices by the username or email address of the user who owns the device (REGSERVER-1160)

- It is now possible to edit licenses with an "extreference" set (REGSERVER-1168)

Registration Server API

- The `<licensekey>` tag must be used in place of the `<licensenumber>` tag in the API. `<licensenumber>` has been deprecated and will no longer be accepted in Registration Server 4.0.

- Added a `<licensekey>` tag and a `<licensereference>` tag to the input parameters of the "registeruser" API call. One of these tags can be used to specify a license to assign to the newly created user.

- Removed the Provider setting `API_CREATE_DEFAULT_LICENSE` (REGSERVER-1163). A default license is now always created when a user is created by the API, or during TeamDrive Client registration.

  Since the Registration Server version 3.6 now allows a license to be assigned to a user, even when the user has no devices, the default license is also assigned to the user on creation via the API. If the license already has the maximum number of users, the new user will not be created.

13.1.8 3.6.1 (2016-12-02)

- Fixed a crash that occurred when search user was called from a TeamDrive Client that is registered at a different Registration Server (REGSERVER-1161)

13.1.9 3.6.0 (2016-11-25)

TeamDrive Registration Server version 3.6 is the next major public release following after version 3.5.

Version 3.6 of the Registration Server contains the following features and notable differences compared to version 3.5.

Installation

- The Reg Server 3.6 supports CentOS 7. RPM's are available for this version of the OS.

Registration Server Functionality

- Added the "Web Portal Access" capability bit. This bit represents user-level permission to access Web Portals. The capability bit is only used if the `ALLOW_WEB_PORTAL_ACCESS` Provider setting is set to `peruser` (see below).

- Added `ALLOW_WEB_PORTAL_ACCESS` Provider setting. This setting determined whether users are permitted to access a Web Portal or not. Possible settings are:

  - `permit`: All users are permitted to login to Web Portals (this is the default).

  - `deny`: Web Portal access is denied to all users.

  - `peruser`: Access is determined by the "Web Portal Access" capability bit.

- TeamDrive Authentication Services now includes an example of how to connect to Vasco IDENTIKEY Authentication Server. When used in conjunction with the Web Portal, Web Portal version 1.0.6 is required.

- Emails sent by the server now have a maximum size of 16 MB. Previously the limit was 64 K (REGSERVER-1131).

- Implemented support for Two-Factor Authentication using the Google Authenticator App.

- Added the `AUTH_SETUP_2FA_URL` Provider setting. This value must be set to the URL of the page used to setup two-factor authentication.

  See registration server how tos/two factor authentication for details.

- Added `ALLOW_MAGIC_USERNAMES` Provider setting. When set to True, users of the Provider may register with usernames that match the standard "magic username" pattern.

- Added `ISOLATED_EMAIL_SCOPE` Provider setting. When set to True, the users of the Provider may use email addresses that are in use by other users, as long as the email addresses are unique for the Provider (REGSERVER-1125).

- Added the `HIDE_FROM_SEARCH` Provider setting. When set to True, this setting will prevent users from being found by a Client when doing the standard username and email address searches, during login and when inviting users to a Space (REGSERVER-1124).

- Added the `PROVIDER_DEPOT` Provider setting. This setting may be used to specify that a certain Depot should be used as default Depot for all users of a Provider (REGSERVER-1117).

- Added the `SUPPORT_EMAIL` Provider setting. This setting specifies the email address that will be notified if support content is uploaded to the Registration Server.

- Users will now receive "store forward" invitations no matter which Registration Server the invitation is located on. Previously a user had to register on the same Registration Server as the store forward message.

  A store forward invitation is created when a user invites another user via email, but the user is not yet registered.

- HTTPS is now used for all communications with a Host Server if the Provider setting API_USE_SSL_FOR_HOST is set to True.

- Added the Registration Server setting: `EmailGloballyUnique`. When set to `True` the Registration Server will check to ensure that an email address is not in use by any other Registration Server in the TeamDrive Network (REGSERVER-809).

  This value is automatically set to the same value as `UserEmailUnique` on upgrade to version 3.6 or later.

  See emailgloballyunique for details.

- LDAP/AD Connectivity (REGSERVER-506): The LDAP/AD external authentication reference code has been improved so that all important parameters are in one configuration file.

  The file "ldap_config.php.example" must be duplicated and renamed to "ldap_config.php" on installation. The file parameters should then be modified as required. Further instructions and a description of the parameters is provided in the "ldap_config.php" file.

Registration Server API

- Updated version number of API to 1.0.007.

- Added notifications: the Registration Server can be configured to send a notification when a change is made to a user. To do this, the Provider setting `API_ENABLE_NOTIFICATIONS` must be set to `True`, and the setting `API_NOTIFICATION_URL` must be set to the URL that will receive the notification (TRUS-136).

- The tag `<webportal>` has been added to the API functions: "searchuser", "loginuser", "getuserdata" and "registeruser". This tag indicates whether the user is permitted to access a Web Portal.

  Note that if the Provider setting `ALLOW_WEB_PORTAL_ACCESS` is set to `permit` or `deny`, the the value returned in the `<webportal>` tag will reflect this setting, not the value of the user's Web Portal Access capability bit.

  When calling "setcapability" the `<capability>` tag may be set to the value "webportal", in order to set Web Portal Access capability bit.

- The "searchuser" API call now accepts the input tags `<distributor>`, `<reference>` and `<authid>`, which are used to search for users with specific external reference or external authentication ID. These tags

can be used in addition to or in place of other search tags. The '*' search wildcard is not recognised which searching for these values.

When searching by `<reference>` and `<authid>` the `<distributor>` will automatically be added to the search conditions (normally this is only done when you set `<onlyownusers>true</onlyownusers>`).

Note that setting `<distributor>` to a value other than your own Provider code is only permitted if you are the "Default Provider". Web Portals working on the behalf of a Provider may also set the `<distributor>` tag accordingly.

- The "registeruser" API call now returns a `<userdata>` block with the complete details of the user. The `<username>` outside of the `<userdata>` block has been deprecated and will be removed in version 4.0.

- Added the Provider setting `EXT_LICENCE_REF_UNIQUE`, default `True`. If set to `False` duplicate license references are allowed (REGSERVER-1130).

- Removed the Provider setting `CLIENT_DEFAULTLICREF`. The license reference must now be provided as parameter to the API call (REGSERVER-1130).

- The `<licensereference>` tag can now be used to specify the license in place of the `<licensenumber>` tag (REGSERVER-808). Note that the license reference must be unique for each Provider, if `EXT_LICENCE_REF_UNIQUE` is set to `True` (which is the default).

- Added the "sendtemplatemail" API call. This call can be used to sent standard template based emails to user, Providers or some other recipient (REGSERVER-1103).

- Added lookup of an Email on TDNS to the "tdnslookup" call. The result is a list of Registration Servers (REGSERVER-1113).

- Client API: the client version will now be extracted from the path: "/teamdrive/clientversion", in addition to the paths used previously. Command names are case-insensitive.

Administration Console

- Added "Delete Provider" Functionality (REGSERVER-1127). Deleting a Provider will delete all user, licenses and depots that belong to the Provider. If the Reg Server is connected to TDNS, the delete process will be suspended until the Provider has been removed from TDNS.

- If too many failed logins are detected for a user, further attempts are subjected to a delay that increases with the number of login attempts, up to a maximum delay of 2 minutes. The previous system of a constant 5 second delay will still be used if the user login is protected by the LOGIN_IP provider setting (REGSERVER-534)

- Added an option to move spaces from one depot to another (REGSERVER-1116)

- Depot change history can be displayed on the edit-user page, when available (REGSERVER-1040)

- A users Spaces are fetched more efficiently when displaying them on the edit-user page, which solves some browser memory problems when a user has a lot of spaces. Unfortunately this also means that the list of spaces can no longer be sorted (REGSERVER-1122)

- The list of spaces on the edit-user page can now be exported as a CSV document (eg. for opening in Excel) (REGSERVER-1128)

- Users can now be added or removed from a license on the edit-license page (REGSERVER-1129)

- Changing a license owner can now be done only via the edit-license page. The function has been removed from the edit-user and license overview pages to avoid confusion with the 'add user to license' function (REGSERVER-1129)

- The Admin Console now displays the Host Server version number. The version number is only correctly updated with Host Server version 3.6.1 or later. Otherwise, the number displayed is the version of the original Host Server installation. Note that, in this case, the version number displayed is of the form: <major>.<minor>.**.<patch>, for example: Host Server version 3.0.011 (for example) is displayed as: 03.00.**.00011.

13.2 Change Log - Version 3.5

13.2.1 3.5.10 (YYYY-MM-DD)

Registration Server API

- The `<licensekey>` tag should be uses in place of `<licensenumber>` in calls that accept this as an input paramater. `<licensenumber>` will still be accepted, but has been deprecated and will be removed in Registration Server version 4.0.

- The "searchuser" API function returns `<licensekey>` instead of `<licensenumber>` (as added in 3.5.9).

- The API calls: "searchuser", "getuserdata", "getlicensedata", "getdefaultlicense", "getusedlicense", "createlicense" and "createlicensewithoutuser" now return the tag `<licensekey>` in addition to `<number>`. The contents is the same. The `<number>` tag is deprecated and will be removed in a future version.

13.2.2 3.5.9 (2017-01-16)

- Avoid adding or removing the depot owner from the user list (REGSERVER-1158)
- Added a new server PrivacyURL and Provider redirect page

Registration Server API

- Added `<showlicense>true/false</showlicense>` tag to the "searchuser" API call. When set to `true`, license information is returned in the result. This includes `<licensenumber>`, `<featurevalue>` and `<licensestatus>` tags in the `<user>` tag which indicate the current license set for the user, and the features of the license. A `<licenselist>` tag is also returned with a list of the licenses that belong to the user.

13.2.3 3.5.8 (2016-08-26)

Note: Version 3.5.8 will fix an error in the depot documents as described below in REGSERVER-1141. To save the successull update the file /var/opt/td-regserver/StartupCache.pbt will be updated. This might fail in case of the wrong user "root" ownership. Please correct the ownership with:

```
chown apache:apache /var/opt/td-regserver/StartupCache.pbt
```

Note: Updating the registration server on CentOS 7 with "yum update" might update the apache to a newer version. This update could re-install the deleted "conf"-files in the folder /etc/httpd/conf.modules.d/ and will prevent starting the apache. Please follow the modified instruction to disable all modules in the "conf"-files instead of deleting them as described in *Enable "Prefork" Mode (CentOS 8)* (page 14)

- Documented additional client settings and ordered client settings alphabetically.

- Fixed the problem that email notifications, such as comments on files, to users on other Registration Servers were ignored. In future, only registered and activated users will be able to send emails. However, the sender can specify an email address instead of a username, in order to send a notification to non-registered users, or users on other Regisration Servers (REGSERVER-1147).

- The Host Server may return a Depot document with a SERVERFLAGS field with an incorrect terminator. These documents will be corrected in the database and when returned by the Host Server (REGSERVER-1141).

- Fixed a bug in "wipedevice" API call (REGSERVER-1139)

- The adminconsole will make requests to hostservers over the hostserver proxy, if one is configured (REGSERVER-1148)

13.2.4 3.5.7 (2016-07-12)

- Fixed a bug in "createlicense" API call: if the user has no other default license, then the created license will now be correctly set as the default.

- The [[GREETING]] in emails templates: "inv-user-invited-passwd" and "inv-user-invited", incorrectly used the name of the sender of the invitation, instead if the invitee (REGSERVER-1136).

- Deleting users, depots, or spaces in the Adminconsole now requires the user to type the word 'DELETE' in a confirmation dialog, to prevent accidental deletion (REGSERVER-1133)

13.2.5 3.5.6 (2016-06-21)

- The ssl configuration has changed. All settings are now located in a separate configuration file. Please remove the old configuration in your ssl.conf:

```
RewriteEngine on
RewriteLogLevel 0
RewriteLog "/var/log/httpd/rewrite.log"

RewriteRule ^/setup$ /setup/ [R]
RewriteRule ^/setup(.*) /yvva/setup$1 [PT]
RewriteRule ^/pbas/td2as/(.*)$ /yvva/$1 [PT]
RewriteRule ^/pbas/td2api/(.*)$ /yvva/$1 [PT]
```

and add the new include as described in chapter *Configure mod_ssl* (page 15)

- The authenticate call now handles authentication tokens that do not contain an email address. The allows an external Authentication Service prevent the automatic creation of a user if the user does not exist.

  If the email address is missing from the authentication token then the Registration Server will return the "user not found" error if the user ID in the authentication does not match an existing user.

  As before the user ID in the token is compared to the "External Authentication ID" field of the user. This field can be edited in the Admin Console, if USE_AUTH_SERVICE is enabled (set to True). If users are not created automatically then it is most likely that this field must be set manually when the user is created.

  The alternative is to import the value of the "External Authentication ID" when creating and users using the CSV import facility.

- Updated Yvva version to 1.3.6 (required with CentOS 7)

13.2.6 3.5.5 (2016-05-14)

- Add support for CentOS 7 with apache 2.4

- When a user is removed, if the users licenses are not removed, the licenses are now correctly freed so the may be assigned to another user (REGSERVER-1120) . Note that the default license is no longer a default license when freed.

- Corrected handling of default license. This could be overbooked (REGSERVER-1119). If a default license is assigned to the owner, and it is overbooked, then it will now be automatically removed from a number of users as required. Removal begins with less active users (users that accessed a device more recently will be favoured when removing licenses).

When a license is removed, the user license is reset to the user's default. Note that this may fail if the user is not the owner of his/her default license, which may be the case when using the `DEFAULT_LICENSEKEY` Provider setting.

- When changing the Provider of a user update of TDNS was not correct in the case when the case-sensitivity of usernames changed (REGSERVER-361).

- Added `<intresult>` tag to result of "createlicense" API call.

- No longer send email notification message for 4.3.1 clients, because they are able to synchronise user data using the "mod protocol" (REGSERVER-1110).

Registration Server API

- The order of the XML tags in the API documentation now matches the actually order of tags returned by the server. Some tags that were ommitted have been added (REGSERVER-949).

13.2.7 3.5.4 (2016-01-25)

- The contents of the <message> tag in an exception was not correctly encoded which lead to invalid XML returned by the DISTRIBUTOR_REDIRECT (-30004) exception, which includes a URL in the message tag.

- Fixed a crash which could occur when assigning a license to a user with a device that was not activated (REGSERVER-1104).

- /bal/*html and /act/*html URLs were incorrectly returning "text/xml" as content type. This has been changed to "text/html" (REGSERVER-1106).

13.2.8 3.5.3 (2016-01-14)

- Added a "Registration Server How To's" chapter to the Admin Guide.

- The transfer limit for depots on hostservers that do not enforce the traffic limit is now displayed as 'Unlimited' (REGSERVER-742)

- Added ',' to the reserved characters that are not allowed in usernames. This is in addition to ';' and '$'.

- When `DEFAULT_LICENSEKEY` is specified the setting `PROFESSIONAL_TRIAL_PERIOD` no longer has an effect. It is considered to be 0, which means that no trial period is available.

- `ClientPollInterval` was incorrectly stored in the database in seconds by the Admin Console. The unit used in the database is 0.2 seconds (i.e. seconds x 5). This has been corrected. Default value is 60 seconds, as before.

- Fixed a bug editing / deleting depots belonging to a provider other than the default provider

- Implemented "one-off-secureoffice-trial" license purchase. This will allow users to start a trial period when using the SecureOffice version of TeamDrive.

- Removed the following Registration Server settings: `MediaURL`, `NotificationURL`, `RedirectorURL`, `UpdateAvailableURL`. All these Settings now use hard-coded URLs that reference the Registration Server (REGSERVER-1100).

- Removed all references to `providerinfo.html` and `clientinfopage.php`. These were used as default redirect pages. Now, if no redirect URL is set, the Registration Server will return a HTML page with a messsage. For example, if a forum URL is not specified by the Provider (`REDIRECT_FORUM` setting), or in the Registration Server setting (`ForumURL`), then a page with the message: "Sorry, your service provider has not specified a forum page", will be returned (REGSERVER-1080).

- The `LoadBalancerURL` may contain multiple URLs separated by a 'l' character. In this case, the TeamDrive Clients will automatically use a different URL for each call the Registration Server.

- Removed `BalanceURL` Registration Server setting. TeamDrive Clients that still use this setting will be directed to a hard-coded URL on the Registration Server: `http://<reg-server-domain>/pbas/td2as/bal/server.xml` (REGSERVER-917).

- Fixed the "MAIL FROM:" header in emails sent. The Reg Server now correctly sets this field according to the `MAIL_SENDER_EMAIL` Provider setting (REGSERVER-1099)

- Fixed a bug: the language passed to the Reg Server on registration was incorrectly converted to upper case and stripped of the location information. The unconverted language sent by the Client is now stored in the database (REGSERVER-1097)

- Fixed a bug in the admin console displaying the license language when editing (REGSERVER-1096)

- The Reg Server now supports a single Web Portal that manages internet access for multiple providers. This means that Multiple providers can use the same IP number in the `API_WEB_PORTAL_IP` setting (REGSERVER-1095)

Registration Server API

- The "registeruser" API call will now always returns a <username> tag as well as the standard <intresult> tag on success. For example:

```
<teamdrive><username>$NEW1-1061</username><intresult>0</intresult></teamdrive>
```

This is useful if the caller wishes to know the magic username generated by the server (REGSERVER-838).

- If a user is created via the API, or by CSV import, then it may not be known which language the user will use. In this case the language may be set to "-". The "-" will be ignored by the TeamDrive Client. API calls will return the default language in this case (REGSERVER-1097)

13.2.9 3.5.2 (2015-12-04)

- Fixed API function "setdistributor" to handle more than one depot in case of switchdepot = true (REGSERVER-1087)

- Fixed sending a store forward invitation in case of device not found fails, if sender is registered at a foreign Reg-Server (REGSERVER-1088)

- AdminConsole: Fixed misleading error message in case of deleting a user

Registration Server API

- Changed API function "confirmuserdelete": allow using the call without sending the user password (REGSERVER-1089)

- Fixed sending Store Forward invitation for a "standalone" Registration Server (REGSERVER-1092)

13.2.10 3.5.1 (2015-11-04)

- Fixed api call "setdepotforuser" and "removedepotfromuser": The depot information sent to the clients used a wrong format (REGSERVER-1085)

- API log view in the admin console will now display API requests from the Web-Portal (REGSERVER-1083)

- Greetings macro was not replaced in mail templates (REGSERVER-1079)

- Added hint in the admin console to show if the background task for sending mails and processing other background tasks is running (REGSERVER-1078)

- Fixed API access in the Apache configuration using the URL from older API documentations (using ../td2api/.. in the URL instead of ../td2as/..) (REGSERVER-1071)

- Fixed deleting a depot for an user in the admin console. Depot was deleted on the Host Server, but the reference on the Registration Server was not removed (REGSERVER-1070)

- Fixed access to missing language column in the email change confirmation page (REGSERVER-1069)

- Fixed wrong path to tdlibs-library folder in upload.php (REGSERVER-1067)

- Changed the default value for the setting TDNSAutoWhiteList to `True` (REGSERVER-1072) and handle the special case of the Master-Server when changing the setting back to false in the admin console. Master-Server could only be disabled when using a white label client (REGSERVER-1073)

- Fixed api call "getusedlicense" to avoid duplicate usernames in user list (REGSERVER-1066)

- Fixed connecting TeamDrive Master Server during the setup in case of server-type "standalone" (REGSERVER-1064)

- Replaced TeamDrive 3 screenshot with TeamDrive 4 in chapter "TeamDrive Client-Server interaction" (REGSERVER-977)

- Added hint in documentation to enable HTTPS for the API communication between Registration Server and Hosting Server (REGSERVER-499)

Registration Server API

- Added API call "changelicensepassword" (REGSERVER-1075) and use bcrypt for license password encryption (REGSERVER-965)

13.2.11 3.5.0 (2015-09-21)

TeamDrive Registration Server version 3.5 is the next major public release following after version 3.0.018.

Note: Please note the the version numbering scheme for the Registration Server has been changed starting with version 3.5. The first two digits of the version string now identify a released version with a fixed feature set. The third digit, e.g. "3.5.1" now identifies the patch version, which increases for every public release that includes backwards-compatible bug or security fixes. A fourth digit identifies the build number and usus

ally remains at zero, unless a rebuild/republishing of a release based on the same code base has to be performed (e.g. to fix a build or packaging issue that has no effect on the functionality or feature set).

Version 3.5 of the Registration Server contains the following features and notable differences compared to version 3.0.018. This includes all changes made for version 3.0.019, which was an internal interim release used to deploy and test most of the new functionality described below.

Installation

- The initial configuration and initialization of a Registration Server is no longer performed by filling out the `RegServerSetup.xml` file and running the `RegServerSetup.pbt` script on the command line. Instead, a web-based setup process has been implemented, which guides the administrator through the steps involved.

- The Registration Server no longer depends on the PrimeBase Application Environment (e.g. the `mod_pbt` Apache module or the `pbac` command line client), provided by the RPM package `PrimeBase_TD` in version 3.0.018). Instead, it is now based on the Yvva Runtime Environment which is already used for the TeamDrive Host Server since version 3.0.013 and newer. The environment is provided by the `yvva` RPM package, which will automatically replace any installed `PrimeBase_TD` RPM package during an upgrade. The central log file `/var/log/td-regserver.log` is the central log location for all Yvva-based components; the previous log files (e.g. `/var/log/pbt_mod.trace`, `/var/log/pbvm.log` or `/var/log/pbac_mailer.log`) will no longer be used.

- The Apache HTTP Server configuration file for the Registration Server has been renamed from `/etc/httpd/conf.d/pbt.conf` to `/etc/httpd/conf.d/td-regserver.httpd.conf`.

- The installation no longer requires the Apache HTTP Server to be configured using the "worker" MPM, which simplifies the overall installation and configuration of the base operating system and allows for using the PHP Apache module instead of the FastCGI implementation for the Administration Console.

- The login credentials required to access the Registration Server's MySQL database server are now stored in a single configuration file `/etc/td-regserver.my.cnf`, which is consulted by all components (e.g. the Administration Console, Registration Server or the Auto Task background service).

- The background service providing the Registration Server Auto Tasks has been renamed from `teamdrive` to `td-regserver` and is now based on the `yvvad` daemon instead of the PrimeBase Application Client `pbac`. Please make sure to update any monitoring systems that check for the existence of running processes. The configuration of the `td-regserver` background service is stored in file `/etc/td-regserver.conf`.

- The PBT-based code of the Registration Server is no longer installed in the directory `/usr/local/primebase`. The content of the `td-regserver` RPM package has been restructured and relocated to the directory `/opt/teamdrive/regserver`.

Registration Server Functionality

- Added support for the new business model introduced with TeamDrive 4 Clients (e.g. full support for trial licenses with an expiration date, restricted Client functionality via Client settings).

- The CSV import of users is no longer performed by a cron job running a separate PHP script anymore. Instead, there is now an additional "CSV Import" Auto Task that provides this functionality.

- Email and HTML activation page templates are no longer stored and managed in the Registration Server's file system. Instead, they are now stored in the Registration Server's database and managed via the Registration Server Administration Console. During an upgrade from a previous version, any existing template files will be imported from the file system into the database. As a result, the following server settings have have been deprecated and will be removed during an upgrade: `PathToEMailTemplates`, `ActivationURL`, `ActivationHtdocsPath`, `HTDocsDirectory`.

- The "Move Store Forward Messages" Auto Task has been removed, as it's no longer required. Store Forward invitations are now forwarded automatically, when a user installs a new device.

- Some license related provider settings have been moved from the `CLIENT` category to the more appropriate `LICENSE` category, namely `CLIENT_DEFAULTLICREF`, `DEFAULT_FREE_FEATURE` and `DEFAULT_LICENSEKEY`.

- The provider setting `API/API_USE_SSL_FOR_HOST` has been moved into the more appropriate `HOSTSERVER` category.

- A number of Server Settings that used to apply to all providers hosted on a Registration Server can now be defined on the provider level. The following provider settings have been added:

  - `API/API_REQUEST_LOGGING`: Set to `True` to enable logging of API requests in the API log. The value is `False` by default.

  - `EMAIL/USE_SENDER_EMAIL`: Set to `True` if you wish to use the actual email address of the user when sending emails to unregistered users, otherwise the value of `EMAIL_SENDER_EMAIL` is always used.

  - `HOSTSERVER/AUTO_DISTRIBUTE_DEPOT`: Set to `True` if the Depot should be distributed automatically.

  - `LICENSE/ALLOW_CREATE_LICENSE`: Set to `True` to allow the creation of licenses. The value is `False` by default and can only be changed by the default provider.

  - `LICENSE/ALLOW_MANAGE_LICENSE`: Set to `True` to allow the management of existing licenses. The value is `False` by default and can only be changed by the default provider.

- Log messages and errors from the Yvva-based Registration Server components as well as the Administration Console can now be logged via `syslog` as well.

Registration Server API

Numerous enhancements and additions to the Registration Server API, to provide more functionality for integrating with external applications (e.g. web shops).

- Added API call `deletelicense`, which marks a license as "deleted". The API call `cancellicense` will set a license to "disabled" instead of "deleted" now.

- Added API call `tdnslookup`, which performs a lookup at the TeamDrive Name Service (TDNS) to find a given user's Registration Server.

- Added new functions: `deactivateuser`, `disableuser`, `enableuser`, updated API reference documentation accordingly.

- Added new function `setdepartment` to set the department reference for a user.

Administration Console

Various security and usability enhancements as well as modifications to support changes made to the Registration Server API and functionality.

Usability Improvements

- Re-organized the navigation for the various Administraion Console pages, ordered and grouped them in a more logical fashion.

- Error messages when making changes to the Provider or Registration Server Settings are now displayed more prominently.

- The Administration Console now prohibits the manual creation of Depot files for system users such as a Host Server's `tdhosting-<hostname>` user.

- The workflow of the **Create Depot** page has been reworked to be more straightforward, and will perform better validation to prevent users from different providers getting assigned to the same Depot. The form now also allows creating a depot as the default depot for the selected user. (REGSERVER-700, REGSERVER-907, REGSERVER-913)

- The login page now displays a notice to enable JavaScript if JavaScript is disabled in the user's browser. (REGSERVER-916)

- You can now filter the license table by expiry date, contract number, and holder email. The contract number and holder email have been added to the table, and the rest of the columns have been compacted slightly to create more space. (REGSERVER-885)

- Trial licenses are marked with a "Trial: <end date>" tag in the "More Details" section of the user overview table, the user editing page, and the license overview. (REGSERVER-891)

- The user overview will display 'N/A' rather than 'Free' as the user's highest license, if the user has no installations yet. (REGSERVER-904)

- Banner management: Example banner elements are now downloaded with an appropriate file name. (REGSERVER-725)

- Searching for a username on the main user list is now case insensitive when the entire username is provided. (REGSERVER-906)

- Most of the input forms on the Administration Console will automatically trim leading and trailing whitespace from text fields. (REGSERVER-912)

- Can reset/delete multiple messages in the email queue at once (REGSERVER-773)

- Can delete multiple CSV-import log files at once (REGSERVER-990)

- The email templates are sorted into categories which can be shown or hidden. Categories of templates that are not relevant (based on provider settings) are hidden by default (REGSERVER-1026)

- The create-provider dialog will only show the TDNS related fields if TDNS access is enabled in the registration server settings (REGSERVER-1032)

- Multiple spaces can be deleted at once, without requiring a complete page reload (REGSERVER-573)

- Deleted licenses are hidden by default, and can be shown by setting a filter option (REGSERVER-825)

- Merged the "LoginSecurity" server settings group into the "Security" group

- Edited some table column labels to be more descriptive (REGSERVER-1057)

Security Enhancements

- The Administration Console can now be configured to require two-factor authentication via email for users that want to log in. The provider-specific setting `LOGIN/LOGIN_TWO_FACTOR_AUTH` can be used to enable this feature. Two-factor authentication is disabled by default.

- A Password complexity level is now indicated when creating/changing passwords.

- Security relevant events are logged either into a local log file `/var/log/td-adminconsole.log` or via `syslog`. In particular, the following events are logged:

    - Failed logins

    - Failed two-factor authorization attempts

    - Changes to security-related Provider/Server settings (e.g. login timeouts, API access lists, etc.)

    - Password changes

    - Changes to the privileges of users

    - Failed session validations

- If, on login, the user already has an active session, require a two-factor authentication step.

- Added server settings that can be used to limit the number of records that may be viewed in the console. (`SearchResultLimit`, `UserRecordLimit`, `UserRecordLimitInterval`)

- When, on login, the user already has an active session, there is the option to immediately end existing sessions (after completing the two- factor authentication step) (REGSERVER-1036)

- The `Manage Servers` page no longer lists all servers on the TDNS network. Instead, there is an option to either enable/disable communication with all other Registration Servers, and exceptions to the chosen default need to be set by entering the exact server name. This is done so that the name of a customer's Registration Server is not automatically visible to everyone else on the TDNS network (REGSERVER-1042).

Added Functionality

- It is now possible to edit the list of users belonging to a Space Depot on the user editing page (REGSERVER-905). Editing of Depots (change limits, delete, activate, etc.) now takes place in a separate dialogue.

- Added a page that can be used to edit the HTML templates for web pages.

- The Administration Console now adds the `<changeinfo>` tag to the following Host Server API calls: `createDepot`, `(de)activateDepot`, and `createDepot`.

- Added functionality to resend Depot information to the user. (REGSERVER-896)

- The Administration Console now uses the Registration Server API to enable/disable/wipe users. (REGSERVER-803)

- Licenses will now be marked as "deleted" with the new `deletelicense` API function. (REGSERVER-883)

- Removing a user from a license will now also remove that license from the user's devices. (REGSERVER-720)

- Licenses are edited strictly via the API, added the **Send email** button to all forms, made license type editable.

- Added support for the new API calls, added support to manage the new license feature flag "Restricted Client" (which allows to enable configurable Client-side restrictions like the maximum number of Spaces).

- Client log files and support requests can now be viewed on the "Download Client Log Files" page. The default provider can view log files for all providers. (REGSERVER-1025 and REGSERVER-1024)

- If the default provider has assigned a hostserver to another provider via the HOST_SERVER_NAME setting, the other provider will be able to create depots on that server even if the provider would not normally have access to the server

13.3 Change Log - Version 3.0.019

13.3.1 3.0.019.8

- Fixed the key-repository count on the edit-user page (REGSERVER-1020)

- Fixed an issue where the Administration console was not using the correct API functions when adding or removing users from a depot (REGSERVER-1061)

13.3.2 3.0.019.7 (2015-07-08)

- Fix for handling update notifications between version 3.x and 4.x. 3.x clients will not get a 4.x upgrade notification (REGSERVER-985)

13.3.3 3.0.019.6 (2015-07-07)

- Can now set the newsletter capability bit when creating and editing users (REGSERVER-1010, REGSERVER-1015, REGSERVER-1008, REGSERVER-1007)

- Added new templates to confirm recieving a newsletter (REGSERVER-1009)

- Handle messages larger 20K to use 1.0 encryption to avoid timeouts (500x faster than 2.x encryption) (REGSERVER-1014, REGSERVER-1012, REGSERVER-418)

13.3.4 3.0.019.5 (2015-06-23)

- Fixed bug caused by WEB_PORTAL_IP handling (REGSERVER-969)

- Administration Console: Support Host Server version 3.0.010 (REGSERVER-976)

- Extend TDNSRequest to handle provider code returned from TDNS (REGSERVER-980)

- Handling update notifications between version 3.x and 4.x. 3.x clients will not get a 4.x upgrade notification (REGSERVER-985)

- Activation code length for email change reduced (same logic as requesting a new password)

- API: upgradedefaultlicense and downgradedefaultlicense accepts the feature strings instead of license bits

13.3.5 3.0.019.4 (2015-06-02)

- Administration Console: It is now possible to edit the list of users belonging to a Space Depot on the user editing page (REGSERVER-905). Editing of Depots (change limits, delete, activate, etc.) now takes place in a separate dialogue.

- Administration Console: Display a notice to enable JavaScript if JavaScript is disabled in the user's browser. (REGSERVER-916)

- Administration Console: fixed a bug that could cause entries in the license- change history to appear in the wrong order (REGSERVER-943)

- API: Function setreference() use newreference XML tag (REGSERVER-936)

- Fixed access to statistic database (REGSERVER-941)

- API: Added tdnslookup-call (REGSERVER-956)

- API: Fixed searchuser-call (handling user and device status)

- API: Security improvement when to switch distributor

- API: Added WEB_PORTAL_IP to allow API access from the web prtal

13.3.6 3.0.019.3 (2015-04-09)

- Administration Console: Fixed a bug then when editing licenses, the correct license type will now be displayed.

- Administration Console: Select the 'yearly' license type by default when creating licenses.

- Administration Console: Will send the correct license-type identifier to the API when creating TDPS licenses.

- Administration Console: The Administration Console now uses the Registration Server API to enable/disable/wipe users. (REGSERVER-803)

- Administration Console: Added functionality to resend Depot information to the user. (REGSERVER-896)

- Administration Console: You can now filter the license table by expiry date, contract number, and holder email. The contract number and holder email have been added to the table, and the rest of the columns have been compacted slightly to create more space. (REGSERVER-885)

- Administration Console: Trial licenses are marked with a "Trial: <end date>" tag in the "More Details" section of the user overview table, the user editing page, and the license overview. (REGSERVER-891)

- Administration Console: Licenses will now be deleted with the new `deletelicense` API function. (REGSERVER-883)

- Administration Console: The user overview will display 'N/A' rather than 'Free' as the user's highest license, if the user has no installations yet. (REGSERVER-904)

- Administration Console: The **Create Depot** page has been reworked to be more straightforward, and will perform better validation to prevent users from different providers getting assigned to the same Depot. The form now also allows creating a depot as the default depot for the selected user. (REGSERVER-700, REGSERVER-907, REGSERVER-913)

- Administration Console: Searching for a username on the main user list is now case insensitive when the entire username is provided. (REGSERVER-906)

- Administration Console: Most of the input forms on the Administration Console will automatically trim leading and trailing whitespace from text fields. (REGSERVER-912)

- API: Fixed a bug in the `wipedevice` function that prevented the "wipeout pending" flag to be set. (REGSERVER-892)

- API: Fixed a bug in the `sendinvitation` function that caused additional Depots not longer to be sent to a user's devices. (REGSERVER-896)

- API: Fixed a bug creating default licenses for a user belonging to a different provider. (REGSERVER-889)

- Installation: Fixed a minor syntax error in RegServerSetup.pbt

- See the changelog-3.0.018.8 change log for additional changes.

13.3.7 3.0.019.2 (2015-03-05)

- Administration Console: Added support for setting the `restricted` feature flag on licenses (previously labeled `enterprise`).

- Administration Console/Documentation: Updated the TeamDrive logo to the new branding.

- Administration Console: Check a license's `extreference` before allow editing of TDPS licenses. (REGSERVER-855)

- Administration Console: Continue to show only the selected license after jumping to a specific license in `licenceAdmin.php` and then removing a user from it.

- Administration Console: Licenses are edited strictly via the API, added the **Send email** button to all forms, made license type editable.

- API: Added new functions: `deactivateuser`, `disableuser`, `enableuser`, updated API reference documentation accordingly.

- Registration Server: added check to handle an empty `LicenseEmail` field when sending out license change notifications to a provider. (REGSERVER-871)

- See the changelog-3.0.018.7 change log for additional changes.

13.3.8 3.0.019.1 (2015-02-19)

- API: Added new function `setdepartment` to set the department reference for a user.

- Administration Console: Added `<changeinfo>` to the following Host Server API calls: `createDepot`, `(de)activateDepot`, and `createDepot`.

- Registration Server: Fixed bug in returning the Server's capability bits to the Client.

- See the changelog-3.0.018.6 change log for additional changes.

13.3.9 3.0.019.0 (2015-01-22)

TeamDrive Registration Server version 3.0.019 is the next major release following after version 3.0.018 (based on 3.0.018.5).

Version 3.0.019 contains the following features and notable differences compared to version 3.0.018:

- Support for the new business model introduced with TeamDrive 4 Clients (e.g. full support for trial licenses with an expiration date, restrict Client functionality via settings).

- Numerous enhancements and additions to the Registration Server API, to provide more functionality for integrating with external applications (e.g. web shops).

- Administration Console: added support for the new API calls, added support to manage the new license feature flag "Restricted Client" (which allows to enable configurable Client-side restrictions like the maximum number of Spaces).

- API call `removeuserfromlicense` failed in case of empty `<changeid>`

- Added API call `deletelicense`. The API call `cancellicense` will set a license to disabled instead of deleted now.

- Administration Console: The workflow of the **Create Depot** page has been improved and now allows creating default Depots for users that do not yet have a default Depot.

- Administration Console: can set whether or not a user should receive the newsletter when creating and editing users

13.4 Change Log - Version 3.0.018

13.4.1 3.0.018.9

- Administration Console: update copyright date (REGSERVER-915)

- Administration Console: fixed a session-handling issue related to parallel ajax requests (the result would usually be a "session variables not set" error in the adminconsole)

13.4.2 3.0.018.8 (2015-04-07)

- Administration Console: prevent editing of the `valid until` license field for licenses that are not either in the `active` or `expired` phase, as this may cause problems with the `restricted` license feature. (REGSERVER-886)

- Administration Console: the `restricted` license feature flag will be sent to the API as `restricted` rather than `enterprise` (REGSERVER-869)

- Administration Console: Restricted licenses are marked with `(Restricted)` on the user overview and user details pages. (REGSERVER-877)

- Administration Console: Allow displaying and entering language codes longer than two characters on the user editing page. (REGSERVER-898)

- Administration Console: Fixed a bug that caused an incorrect count of a user's installations and invitations on the user overview page. (REGSERVER-901)

- Administration Console: Fixed a bug on the edit-user page that prevented editing users that had been flagged for deletion. (REGSERVER-902)

- Administration Console: The Administration Console will now send the affected user's provider code instead of the provider code of the user logged into the Administration Console when creating Depots and inviting other users to that Depot. (TRUS-61)

- API: The API now allows setting language codes as defined in **RFC 5646** (e.g. `en_US` or `de_DE`) which will be used by TD4 clients when registering a new user. (REGSERVER-898)

- Registration Server: Improved error logging: the output of several error messages (e.g. error codes -24916, -24919, -24909, -24913 or -24912) is now truncated and reduced to the relevant parts.

  Error messages are now dumped in the following form:

  ```
  03/16/2015 15:23:19 #1 ERROR: ERROR -24777: "reg_shared.pbt"@client line 183:
  This is an error! [command=setparcels;device=377]
  ```

  The Registration Server now reads out the log level defined in variable 342 of the `pbvm.env` configuration file so that it is used in code run by the PBT Apache module `mod_pbt` (previously, the log level was ignored by the PBT module). Valid log values are: 0=Off, 1=Errors, 2=Warnings, 3=Trace. (REGSERVER-859)

- Registration Server: When creating a new device, the device now receives the same license as all other devices, independent of the license's status. (REGSERVER-888)

- Documentation: Fixed link structure in the HTML documentation so that clicking **Next** and **Previous** works as expected (REGSERVER-908)

- Documentation: Removed the chapter that describes the MySQL databases and tables that will be installed from the Reference Guide. (REGSERVER-899)

13.4.3 3.0.018.7 (2015-03-05)

- Administration Console: Added support for setting the `restricted` feature flag on licenses (previously labeled `enterprise`).

- Administration console: Updated list of template types viewed in the mail queue view. (REGSERVER-841)

- Administration console: Updated misleading text when viewing device messages from users located on another server. (REGSERVER-839)

- Registration Server: Fixed that `ProfileDataExchangeEnabled` was not checked when changing a user's email address and the Registration Server database schema has not been converted to the 3.0.018 schema. (REGSERVER-849)

- API: Fixed that `UserEmailUnique` was not enforced when registering users via the API. (REGSERVER-730)

- API: Added support for setting the "Restricted" license flag, which can be used to disable/limit certain TD 4 Client functionality. Previously, this feature flag was labeled "Enterprise", but it was not actively used. (REGSERVER-867)

- Registration Server: Added missing provider setting `REDIRECT/REDIRECT_HOME` that sets the provider's home page URL used in the user's start menu. (REGSERVER-851)

- Registration Server: fixed mail template fallback code to fall back to the English templates as a last resort, if a default template in the provider's default language is not available. (REGSERVER-858)

- Documentation: Updated API chapter and replaced the incorrect statement that the temporary password generated by the "sendpassword" API call expires after a time period of 10 minutes with a notice that a generated temporary password remains active and unchanged until the user's password will be changed. (REGSERVER-870)

13.4.4 3.0.018.6 (2015-02-19)

- Installation: To simplify the configuration for new deployments, the default license issued to Clients is now a Professional license including WebDAV support (the value of `LICENSE/DEFAULT_FREE_FEATURE` was changed from `3` to `10`). This change only affects new Registration Server installations, the setting remains unchanged when updating existing installations. (REGSERVER-821)

- Installation: Updated `mysql_install.sh` to re-create InnoDB log files after changing `innodb_log_file_size` in `my.cnf`. (REGSERVER-847)

- Installation: fixed bug in the `setLicenseExpiryDefault()` upgrade routine which inserted incorrect entries into the `td2reg.TD2OwnerMeta` table for existing licenses having a non-NULL value in the `ValidUntil` column. (REGSERVER-848)

  If you have have performed an upgrade from a previous Registration Server version to version 3.0.018 before (which included calling `setLicenseExpiryDefault()`) **and** you have issued licenses with an expiry date, please perform the following steps to remove the incorrect entries. Start the MySQL client `mysql` as user `teamdrive` and enter the following command to delete the entries:

  ```
  mysql> DELETE FROM td2reg.TD2OwnerMeta \
      -> WHERE Name="ENABLE_LICENSE_EXPIRY" AND \
      -> OwnerID NOT IN (SELECT DISTINCT ID FROM td2reg.TD2Owner);
  ```

  Afterwards, verify the setting `ENABLE_LICENSE_EXPIRY` for all providers hosted on your Registration Server and only set it to `True` when this provider intends to issue licenses with an expiry date.

  Note that while it was possible to create licenses with an expiry date in previous versions, the Registration Server did not actually check this date prior to version 3.0.018. To avoid an unexpected expiry of existing licenses after upgrading to version 3.0.018, the upgrade function `setLicenseExpiryDefault()` checks all existing licenses during an upgrade and sets the Provider setting `ENABLE_LICENSE_EXPIRY` to `False` for the respective Provider.

- Administration Console: Added missing `<distributor>` field to the `cancellicense` and `resetpassword` API calls that prevented the default provider from deleting licenses or resetting the user passwords for other providers hosted on the same Registration Server. (REGSERVER-827)

- Administration Console: Fixed bug where **View mail queue** did not show all queued email messages (outgoing invitation emails to unregistered users were not displayed). (REGSERVER-818)

- Administration Console: when importing email templates from the file system into the database, line endings are now automatically converted to be properly terminated with CRLF (`\r\n`)

- Admin Console: Fixed error message `API error code: -30100,message: User name not provided` when deleting a user's default Depot (the Depot was still deleted as requested). (REGSERVER-835)

- Administration Console: updated the regular expression that checks for valid URLs in the the `LogUploadURL` field to accept URLs beginning with `https` as well. (REGSERVER-837)

  Note that this change is not applied automatically to the configuration table during an update. For existing installations, you need to update the field `Format` in table `td2reg.TD2Setting` for this setting as follows, if you want to change the URL via the Administration Console:

  ```
  mysql> UPDATE td2reg.TD2Setting \
  SET Format="^(http|https)://[a-zA-Z0-9\-\./]+/.-$" \
  WHERE NAME="LogUploadURL";
  ```

- Administration Console: Fixed bug that prevented users logged into the Admin Console with their "magic username" to set their password. Also improved session handling to not drop the session when a user logged into the Admin Console changes his own password (which invalidated the existing session before).

- API: The call `getuserdata` failed with `User does not exist`, if `USE_EMAIL_AS_REFERENCE` was set to `True` and the email address was used as the user name. (REGSERVER-824)

- Registration Server: When using external authentication, TD4 Clients could sometimes receive spurious logout events, requiring the user to log in again. Please note that this bug fix may cause Clients that use external authentication to logout again *once* after the upgrade. After that, such apparently random log-outs should no longer occur. (REGSERVER-820)

- Registration Server: Fixed wrong path in the fallback routine that is supposed to use the default mail template for templates missing from a provider's template folder. (REGSERVER-842)

- Registration Server: Fixed bug that caused file comment notification emails to include the recipient's email address in the From:-Header instead of the sender's email address. (REGSERVER-843)

- Registration Server: When changing `HAS_DEFAULT_DEPOT` from `True` to `False`, a user's devices no longer offered a user's already existing default depot for creating Spaces. (REGSERVER-834)

- Registration Server: Outgoing email messages (e.g. Space invitations) could violate **RFC 5321**, if templates did not use the appropriate line termination character sequence (CRLF, `\r\n`). Now, all outgoing email messages are reformatted before submission to the MTA. (REGSERVER-833)

- Registration Server: Fixed bug that prevented users from logging in with their user name in different capitalization if `UserNameCaseInsensitive` was set to `True` (which is the default) (REGSERVER-823)

- Registration Server: Shortened the temporary password that gets generated and mailed to a user when a user's password needs to be changed (e.g. via the "Forgotten Password" option in the Client or via the `sendpassword` API call. Previously, the temporary password consisted of a random MD5 string (32 characters), that turned out to be difficult to handle (e.g. on mobile devices). It now returns a combination of the characters 0-9, a-z and A-Z (excluding 0, O, l and 1, which can be misread). The length of the temporary password now depends on the Client version: 2.x –> 32 characters (unchanged), 3.x –> 8 characters, 4.x –> 5 characters. The 3.x and 4.x Clients have been changed to accept 4 or more characters, the API uses the version of the most recently used device. (REGSERVER-831)

- `upload.php`: Improved security of the PHP script that accepts Client debug log uploads (e.g. to prevent potential XSS attacks), removed absolute path name from the generated upload status file. Note: this script is not included in the RPM distribution and is not installed by default. (REGSERVER-836)

13.4.5 3.0.018.5 (2015-01-23)

- Registration Server: Fixed Space invitation emails to existing users that contained the recipient as the sender in the mail header. (REGSERVER-817)

- Installation: added a new RPM package `td-regserver-doc-html` that contains the Registration Server documentation in HTML format, installed in the Registration Server's Apache document root `/var/www/html/td-regserver-doc/`. Access to the documentation can be restricted by editing `/etc/httpd/conf.d/td-regserver-doc.conf`. (REGSERVER-816)

- Registration Server: disabled banner support for legacy TD 2.x clients

13.4.6 3.0.018.4 (2015-01-13)

- Administration Console: Improved reporting of HTTP errors during API requests. (REGSERVER-798)

- Administration Console: Fixed API error changing a user's email address if the user name contained UTF-8 characters. (REGSERVER-775)

- Administration Console: fixed support for activating/deactivating Space Depots. (REGSERVER-810) This requires Host Server version 3.0.013.8 or later.

13.4.7 3.0.018.3 (2014-12-17)

- Administration Console: fixed incorrect hex encoding of email templates when initially importing them from the file system into the database. (REGSERVER-806)

- Administration Console: added new Reg Server setting `RegServer/RegServerAPIURL` for setting a custom URL to issue Reg Server API requests (e.g. in case of a dedicated API server or if https should be used for API requests). If not set, the API URL will be derived from the `RegServerURL` setting (REGSERVER-799).

- Administration Console: The default provider can now set new passwords for other providers (REGSERVER-768).

- Installation: removed `<APIChecksumSalt>` from `RegServerSetup.xml` and updated the installation instructions accordingly, to simplify the installation process (this value is generated by `RegServerSetup.pbt` automatically during the initial installation).

- Installation: updated installation instructions and VM installation script to install the `php-mbstring` package (required for the email template import into the database). (REGSERVER-802)

- Installation: updated installation instructions and VM installation script to set `date.timezone` in `/etc/php.ini`, to avoid frequent PHP warning messages when using the CSV import cron job. (REGSERVER-801)

- Installation: the RPM now automatically re-creates the file `StartupCache.pbt` and calls `HTTPRequest.pbt` during an upgrade (e.g. to add new Reg Server settings) (REGSERVER-800)

- Installation: added `max_allowed_packet=2M` to the MySQL configuration file `my.cnf`, to support uploading User Profile information containing profile pictures. In order to support this feature, the `PrimeBase_TD` package also needs to be updated to version 4548.120 or newer (TDCLIENT-1663).

- Installation: changed `MaxRequestsPerChild` in `httpd.conf` from `0` to `10000`, to ensure Apache child processes are restarted from time to time (REGSERVER-762)

- Registration Server: Fixed that `SETTING_TDNS_PROXY_URL` gets overwritten by the `SETTING_HOST_PROXY_URL` setting (in case accessing TDNS requires using a different proxy server than accessing the Host Server (REGSERVER-769).

13.4.8 3.0.018.2 (2014-11-12)

- 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 (REGSERVER-734)

- Fixed issue with store forward messages that were not forwarded to a user upon registration (REGSERVER-759)

- Administration Console: Fixed encoding issue when adding users with usernames containing UTF-8 characters (REGSERVER-756)

- Administration Console: Fixed minor bug in the "Add new provider settings" menu (REGSERVER-747)

- RegServerSetup.xml: Fixed missing closing bracket in the `APIChecksumSalt` tag.

- API: fixed `addXMLDepot` call that returned invalid URLs when the setting `SIMULATE_REGSERVER_20` was enabled. (REGSERVER-741)

13.4.9 3.0.018.1 (2014-11-05)

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, 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-4.6.2.0-0.el6.noarch.rpm, files installed in `/usr/local/primebase/setup/scripts`)

  - The PHP-based Administration Console and support files (td-regserver-adminconsole-4.6.2.0-0.el6.noarch.rpm, files installed in `/var/www/html/adminconsole` and `/var/www/html/tdlibs`)

- – The Registration Server documentation in HTML format (td-regserver-doc-html-4.6.2.0-0.el6.noarch.rpm, files installed in the Apache server's document root `/var/www/html/td-regserver-doc/`, access to the documentation can be restricted by editing `/etc/httpd/conf.d/td-regserver-doc.conf`).

  - – 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 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 `UseExternalAuthentification` and `UseExternalAuthentificationCall` 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 sucessful 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 Administration 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 `VIEW-LOGS` 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 Administration 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 Administration Console instead:

    - `LoginSecurity/LoginSessionTimeout` (default: `30`)

    - `LoginSecurity/FailedLoginLog` (default: `/var/log/td-adminconsole-failedlogins.log`)

    - `LoginSecurity/LoginMaxAttempts` (default: `5`)

    - `LoginSecurity/LoginMaxInterval` (default: `60`)

    - `RegServer/ApiLogFile` (default: `/var/log/td-adminconsole-api.log`)

    - `RegServer/RegServerAPIURL` (previously known as `$regServerUrl`, not set by default)

    - `RegServer/ServerTimeZone` (default: `Europe/Berlin`)

  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 admin_console_config for details.

- Disabling a user does no longer provide the **apply to devices** option, as it's sufficient to disable the user to block access to the TeamDrive service.

- A user's Space Depots on a Host Server can be activated/deactivated (added in 3.0.018.4, requires Host Server version 3.0.013.8 or later).

- The default provider can now set new passwords for other providers (added in 3.0.018.3).

- 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.

13.5 Change Log - Version 3.0.017

13.5.1 30017.13 (2014-09-02)

- 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 `ServerTokens` to `Prod` and `ServerSignature` to `Off` in `httpd.conf`, to disable displaying the Apache HTTP Server version and OS version in the HTTP headers and on error pages (REGSERVER-608)

- Added missing tag `<APISendEmail>` in `DIST.xml` template file

- Security improvement: disabled unneeded HTTP methods in `pbt.conf` (only allow GET, POST, disable PUT, HEAD, OPTIONS, TRACE) (REGSERVER-613)

- API: added new API call `removedepotfromuser` extended `setdepotforuser`. Fixed bug in `setreference` and removed deprecated `location`-Support in `getHostForDistributor`. Fixed error handling in `setinviteduser`. Updated API-Version number to "1.0.005".

- For monitoring purposes, calling the Reg Server's ping URL with the optional parameter `tdns=$true`` (e.g. ``http://regserver.yourdomain.com/pbas/td2as/reg/ping.xml?tdns=$t` now also performs a TDNS lookup, to verify that the communication between the Reg Server and TDNS is working properly.

13.5.2 30017.12 (2014-07-09)

- 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)

13.5.3 30017.11 (2014-06-26)

- 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)

13.5.4 30017.10 (2014-06-17)

- 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 reStructuredText/Sphinx

- `RegServerSetup.xml`: Fixed incorrect closing tag (`</ProviderInfoURL>` -> `</DownloadURL>`)

13.5.5 30017.9 (2014-04-17)

- 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"

13.5.6 30017.8 (2014-03-27)

- 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 users in `p1reg.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

- `CSV_IMPORT_ACTIVE` should not add `CSV_UPLOAD_DIR`, `CSV_ERROR_DIR` and `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`

13.5.7 30017.7 (2014-03-14)

- Fixed nasty typo in `RegServerSetup.xml`

13.5.8 30017.6 (2014-03-14)

- 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

13.5.9 30017.5 (2014-02-25)

- 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

13.5.10 30017.4 (2014-02-07)

- 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

13.5.11 30017.3 (2014-02-04)

- 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

13.5.12 30017.2 (2014-01-30)

- 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

13.5.13 30017.1 (2014-01-23)

- First build using the scripted build, updated `RegServerSetup.pbt` and included some Admin Console
  fixes

13.5.14 30017.0 (2013-10-23)

- Not final; Bcrypt is still missing

APPENDIX

14.1 Glossary

Client The software application used by users to interact with the TeamDrive system. Can be customized to various degrees. Every device requires a Client application.

Device A computer used by a user to access the TeamDrive system.

Installation Simply refers to the installation of the client application on a device.

User A person using the TeamDrive System.

Provider (aka Distributor or Tenant) The "owner" of some set of Users. See provider concept for a detailed explanation.

Space A virtual folder containing data that can be shared with other TeamDrive users. This is what TeamDrive is all about.

14.2 Abbreviations

PBT **P**rime**B**ase **T**alk

SAKH **S**erver **A**ccess **K**ey **H**TTP for TeamDrive 2.0 Clients

TDNS **T**eam**D**rive **N**ame **S**ervice

TDPS **T**eam**D**rive **P**ersonal **S**erver

TDRS **T**eam**D**rive **R**egistration **S**erver

TDSV Same as **SAKH**, but for TeamDrive 3.0 Clients: **T**eam**D**rive **S**erver

R

RFC
    RFC 5321,
    RFC 5646,