TeamDrive Registration Server Administration Guide

Release 4.6.4.0

Paul McCullagh, Eckhard Pruehs

2022

CHAPTER

ONE

COPYRIGHT NOTICE

TeamDrive Systems GmbH
https://www.teamdrive.com
Max-Brauer-Allee 50
22765 Hamburg, Germany
Email: info@teamdrive.com

CHAPTER

TWO

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.

CHAPTER

THREE

DOCUMENT OVERVIEW

This document primarily covers usage of the Admin Console, but also includes:

- Importing user data with a CSV import
- Configuring a system with backup, monitoring, failover, and scaling strategies
- Configuring TDNS settings
- and updating a registration server to version 3.5

This documentation describes the functionality of the current release version 4.6 which supports external authentication. The chapters which belong to 4.6 will be marked in this document. You also need a recent client version to use it together with version 4.6 of the TeamDrive Registration Server.

CHAPTER

FOUR

USING THE ADMINISTRATION CONSOLE

The TeamDrive Registration Server Administration Console, also known as the "Admin Console", is a web-based application for the administration of all aspects of the Registration Server.

You can login to the Admin Console either using "provider credentials" or by using your standard user credentials. Provider credentials are a username and password which are created when the provider is created.

Provider credentials can be changed by users with `EDIT-PROVIDER` rights (see *EDIT-PROVIDER* (page 31)).

In order to login as with standard user credentials you must have the `LOGIN-RIGHT` privilege (see *User Rights* (page 28)).

Note that login is also subject to the `LOGIN_IP` and `PROVIDER_LOGIN_IP` provider settings (see adminconsole_settings).

4.1 Access Levels

Admin Console functionality is divided into four access levels:

4.1.1 Superuser-Level:

This access level is reserved for the owner of the Registration Server. Superusers have all rights to the server and have access to all aspects of the server.

When a Registration Server is installed an initial provider is created, called the "Default Provider". Users that login with the Default Provider credentials (provided during installation) have superuser access.

The Default Provider can be changed later by modifying the global server setting `DefaultProvider`.

Users can be explicitly granted `SUPER-USER` rights (see *User Rights* (page 28)). In this case the user has the same privileges as a user that logs in with the credentials of the user's provider.

4.1.2 Provider-Level:

Providers are so-called because they provide a TeamDrive service to a (possible very large) number of TeamDrive users.

Certain aspects of the TeamDrive service can only be controlled at the provider level, for example: access to Hosting Services, and other services such as a Web Portal or External Authentication Services.

A user that logs in with the provider credentials has full control of all servers, settings, accounts and users of the provider.

Users can be granted provider level access by granting them the `PROVIDER-MANAGER` right (see *User Rights* (page 28)).

A provider can be configured to be managed by some other provider. This can be done by setting the **Managed by** field of the Provider details.

Note: New in this release: User's that login with Provider-Level privileges have control of all providers that their provider manages. In this case there is a drop-down menu at the top of the Admin Console page which is used to select the provider. In addition, search filters provide an "Include all providers" option.

4.1.3 Account-Level:

Accounts are group of users and associated resources. Account level access (new in Registration Server 4.0) allows a user to manage a number of users, clients/devices licenses and depots.

To gain Account-Level privileges you must be added as a manager of one or more accounts. The user is then automatically granted the `LOGIN-RIGHT` and `ACCOUNT-MANAGER` rights.

4.1.4 User-Level:

If a user is granted the `LOGIN-RIGHT` privilege, then they will access the Registration Server at the User-Level. This means that they only have access to their own user details, and objects, such as client, licenses and depots that belong to the user.

4.2 Security Considerations

We strongly recommend accessing the Admin Console via SSL/HTTPS only. Our preconfigured Virtual Appliance images provide a self signed SSL certificate and access is possible via HTTPS only. You should replace this certificate with an official one, if this server is publicly accessible.

You can also limit access to the Admin Console to individual IP addresses, by using the built-in provider setting `LOGIN/LOGIN_IP`. This setting defines the IP addresses (as a comma-separated list) that are allowed to connect to the Admin Console as a given provider.

If you require more flexibility in restricing access, e.g. by restricting it to an IP address range or by host/domain names, we suggest using the Apache http Server's built-in functionality:

https://httpd.apache.org/docs/2.4/mod/mod_authz_host.html

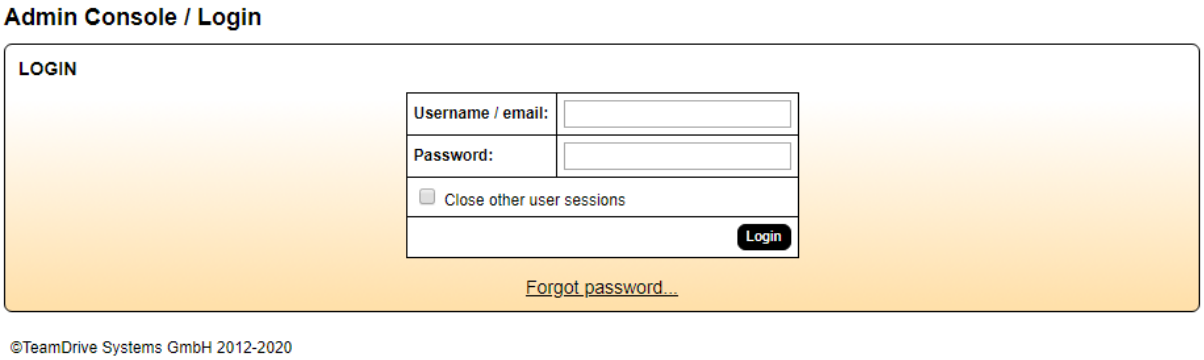
The safest strategy is separating the Admin Console from the Registration Server by installing it on a dedicated server, which is only accessible by you.

4.3 Logging in / Logging out

To log into the TeamDrive Registration Server Admin Console, open the Admin Console's URL in your web browser, e.g.

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

Enter your username or email address and password to log in.



You can login using either provider login credentials or the your standard TeamDrive user credentials.

If provider credentials happen to be identical to those of a TeamDrive user, then the provider login takes priority.

To login with standard user credentials you must have the `LOGIN-RIGHT` privilege (see *LOGIN-RIGHT* (page 32)). This right is granted automatically to users that are account managers.

To log out, select the "Logout" option from the menu in the top right hand corner.

4.3.1 Lost Password

When you login using standard user credentials, you can click on the "Forgot password..." link if you have forgotten your password. Then click on the **Send Temporary password** button to have a temporary password sent to your email address. You can then login and set a new password using the temporary password in the email.

Note: The lost password feature is not available for login using provider credentials.

4.3.2 Changing the Login Password

To change your user password ater login, click on the "Change Password" option in the menu in the top right hand corner.

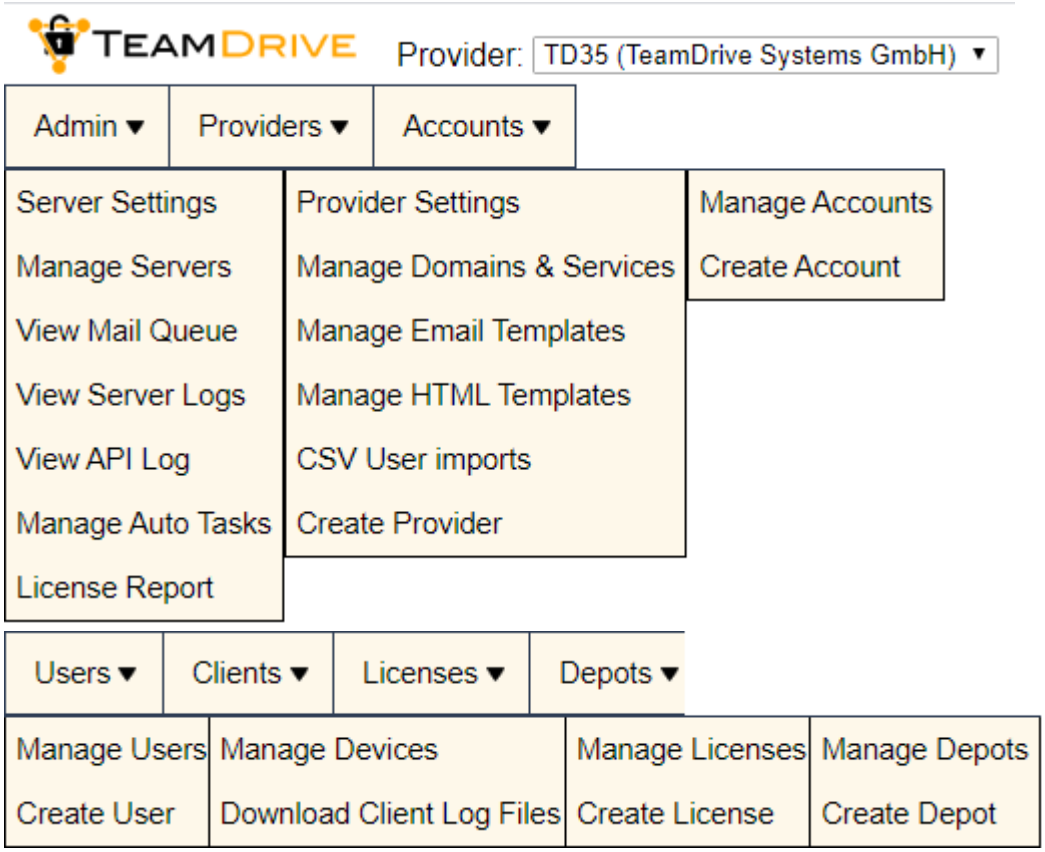
You will see a form prompting you to enter your current password and a new one. Since the password is hidden, you are required to enter it twice, to ensure you have entered it correctly.

Once you have entered the current and new password, click **Change Password** to save the change, or click **Back** at any time to go back to the previous page.

Changing the password of a TeamDrive user on the Admin Console will require all TeamDrive clients to re-login before synchronisation will continue.

Note: If you currently do not have a TeamDrive client installation, but you have previously created spaces using a TeamDrive installation that is now lost, changing your password will cause you to loose access to the space keys stored in the TeamDrive Key Repository managed by the Registration Server.

4.4 Main Menu



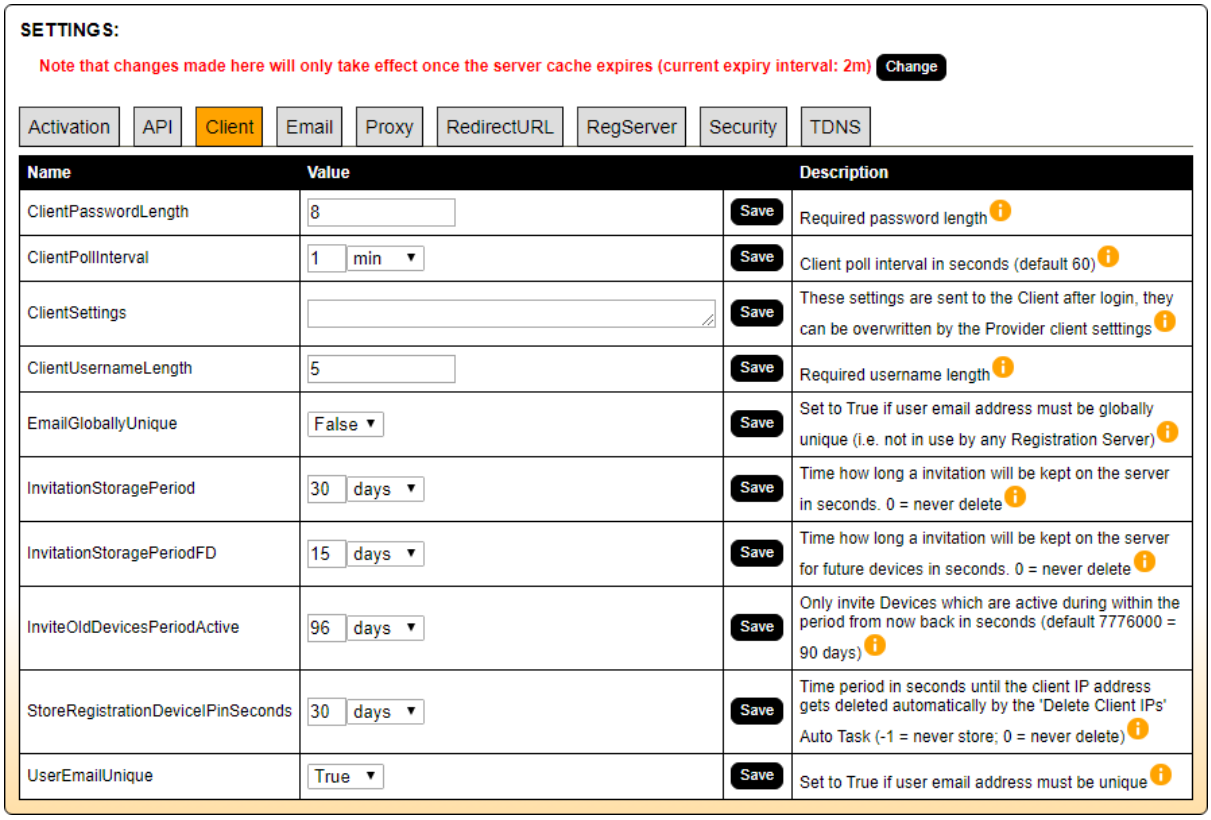
The sub menue entries are described in the following chapters. It depends on the access permissions if all entries are visible for the current user.

4.5 Admin

This section allows you to administrate various aspects of the Registration Server and view several log files.

4.5.1 Server Settings

Click **Admin/Server Settings** in the navigation bar to go to the **Server Settings** page. Only users with the `EDIT-SETTINGS` privilege can view and edit server settings.



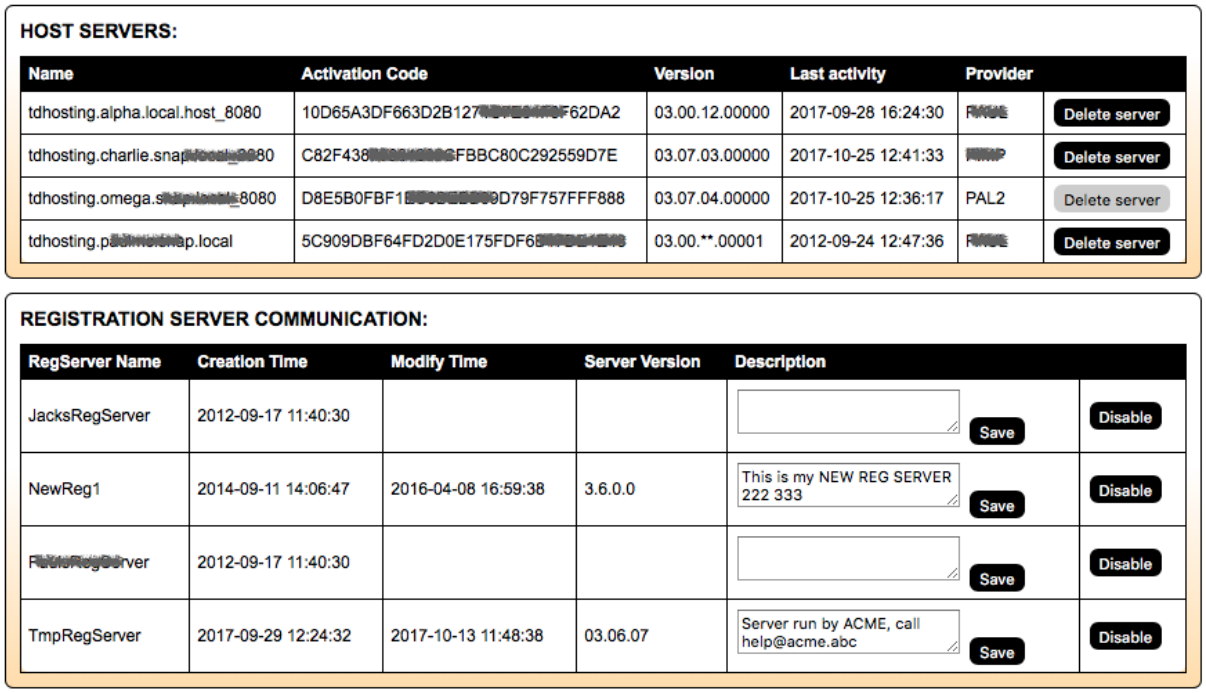
Warning: Changes to the Registration Server settings will only be active after the caching period defined in `RegServer/CacheIntervall` has passed (the default is 1800 seconds or 30 minutes). If no cache interval was set, you need to restart the Apache HTTP Server of the Registration Server to reload these values.

To change a setting, select one of the top level categories (e.g. **Client**, **RegServer**, etc.), change the desired setting either by entering a new value or selecting one from the drop down menu, and click **Save** in that value's row. Do not change more than one value at once — always save your change before modifiying another value. Note that not all settings are editable.

Each setting provides a short description about its meaning and a link to the documentation. All settings and possible values are explained in more detail in registration server settings in the *TeamDrive Registration Server Reference Guide*.

4.5.2 Manage Servers

Click **Admin/Manage Servers** in the navigation bar to perform management tasks related to Host Servers and other Registration Servers.



HOST SERVERS:

Name	Activation Code	Version	Last activity	Provider	
tdhosting.alpha.local.host_8080	10D65A3DF663D2B127[redacted]F62DA2	03.00.12.00000	2017-09-28 16:24:30	[redacted]	Delete server
tdhosting.charlie.sna[redacted]80	C82F438[redacted]FBBC80C292559D7E	03.07.03.00000	2017-10-25 12:41:33	[redacted]	Delete server
tdhosting.omega.s[redacted]8080	D8E5B0FBF1[redacted]D79F757FFF888	03.07.04.00000	2017-10-25 12:36:17	PAL2	Delete server
tdhosting.p[redacted]p.local	5C909DBF64FD2D0E175FDF6[redacted]	03.00.**.00001	2012-09-24 12:47:36	[redacted]	Delete server

REGISTRATION SERVER COMMUNICATION:

RegServer Name	Creation Time	Modify Time	Server Version	Description		
JacksRegServer	2012-09-17 11:40:30				Save	Disable
NewReg1	2014-09-11 14:06:47	2016-04-08 16:59:38	3.6.0.0	This is my NEW REG SERVER 222 333	Save	Disable
F[redacted]rver	2012-09-17 11:40:30				Save	Disable
TmpRegServer	2017-09-29 12:24:32	2017-10-13 11:48:38	03.06.07	Server run by ACME, call help@acme.abc	Save	Disable

The **Host Servers** section lists all Host Servers that have been registered/associated with providers hosted on the Registration Server instance. From here you can also obtain the **Activation Code** that is required to finalize the Host Server installation and registration process (see the *TeamDrive Host Server Installation Guide* for details). It's also possible to remove a Host Server by clicking **Delete Server**, which detaches it from the provider it has been registered with and deletes the corresponding user and device entry.

Note: Only a Host Server which is not already in use by clients can be deleted.

Important: Please enable HTTPS for the API communication between Registration Server and Host Server in case that your Server is configured to allow HTTPS communication (setting `HostServer/API_USE_SSL_FOR_HOST`).

If TDNS access is enabled (setting `RegServer/TDNSEnabled`), the **Manage Servers** page also allows you to enable communication with other Registration Servers.

Enabling a Registration Server allows your local users to directly invite users managed on that other Registration Server into their Spaces.

By default, communication with all other servers is disabled/enabled according to the `RegServer/TDNSAutoWhiteList` setting. This setting can either be changed directly on the **Admin/Server Settings** page, or on the **Manage Servers** page by changing the default to enabled/disabled and clicking "Save".

Note: The communication to `TeamDriveMaster` must always be enabled in case your are using the TeamDrive standard client.

You can set exceptions to the current default rule by entering a specific server name in the form field at the bottom of the page and clicking "Add". The current list of exceptions is displayed along with the the chosen default rule.

Another exception to the default is the Master Registration Server, which is enabled/disabled separately via the selection field at the top of this section of the page.

If communication with other servers is enabled by default, your Registration Server obtains a list of all known Registration Servers from the Master Registration Server, "TeamDriveMaster", every 12 hours via a background

task (see *Manage Auto Tasks* (page 14)).

4.5.3 View Mail Queue

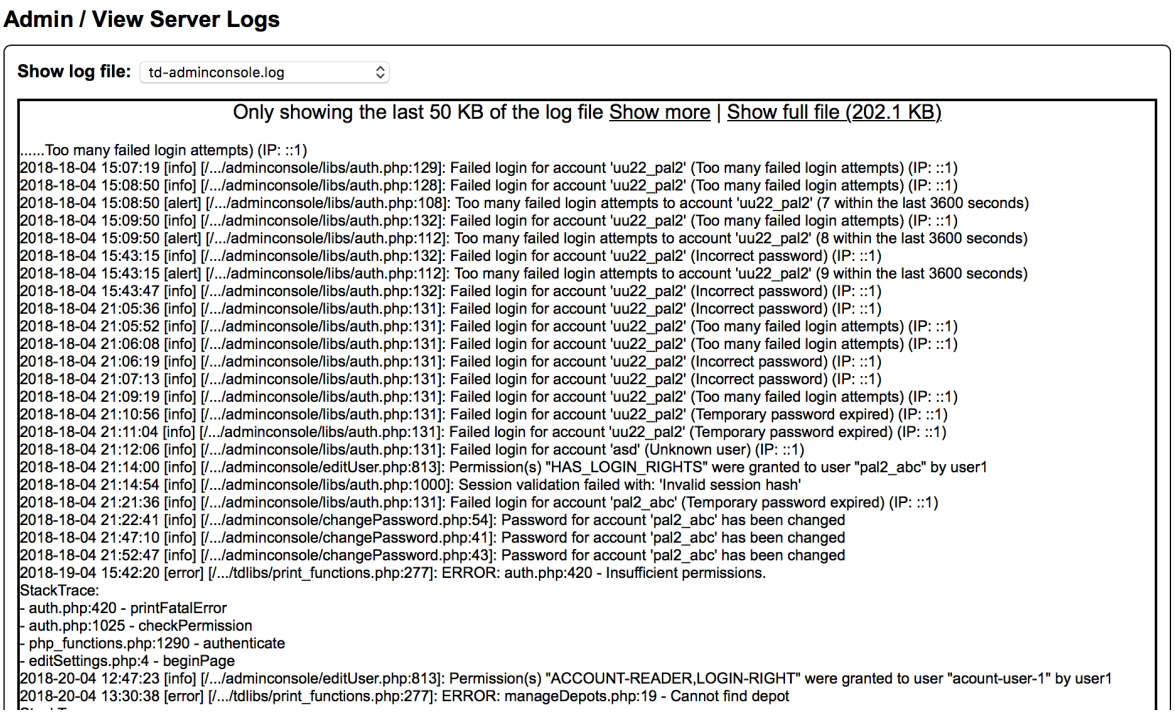
Click **View Mail Queue** to get an overview of the current mail queue, which lists all emails that have not been delivered to the respective users yet.

Pending outgoing emails can be shown here due to the fact that the "Send Emails" auto task hasn't procesed the mail queue recently (such messages have the status "created"), or there were issues with the email address or when submitting messages to the MTA (the status of these messages is "failed").

Click **Reset Status** to enqueue a message for delivery again. Click **Delete** to remove a message from the queue.

4.5.4 View Server Logs

The Admin Console allows viewing selected server log files for troubleshooting purposes. The **View Server Logs** page is only visible for users having the `VIEW-LOGS` privilege.



Depending on the availability and access permissions, the following log files can be viewed by selecting them fromm the **Show log file:**

- `/var/log/httpd/error_log`
- `/var/log/td-regserver.log`
- `/var/log/td-adminconsole-api.log`
- `/var/log/td-adminconsole-failedlogins.log`

As it requires physical read access to these logs, this feature works best when the Admin Console is installed on the same host where the Registration Server instance is running on. Log files can only be viewed if the user that the Apache HTTP Server is running under (usually `apache`) has the required read access privileges to view these files.

The list of log files is defined in the (read-only) Registration Server setting `ServerLogFiles` and can only be modified by updating the setting in the database directly.

`/var/log/httpd/error_log`: The standard apache error log file (change the access rights using the command `chmod 755 /var/log/httpd` to view this file).

`/var/log/td-regserver.log`: The yvva background task will log errors into this file and also the errors which might occur from the client requests will be logged to this file.

`/var/log/td-adminconsole-api.log`: The API requests from the Admin Console to the Registration Server API and host server API will be logged to this file.

`/var/log/td-adminconsole-failedlogins.log`: Failed logins to the Admin Console will be logged to this file.

4.5.5 View API Log

Most of the tasks performed via the Admin Console result in API calls being sent to the Registration Server. You can also utilize API calls in your own applications, if you need to interact with the Registration Server.

See the chapter registration server api calls for an overview of the available API calls.

If you enabled the logging in your provider setting `API/API_REQUEST_LOGGING` and you are either logged in as the Default Provider or with a provider/user that has the `VIEW-API-LOG` privilege, you can view a log of all incoming API requests and their results by clicking **View API Log** in the menu bar.

The API request log is stored in the Registration Server's MySQL database and can be filtered by various criteria, e.g. **Date created**, **User**, and **Command**.

Note: Enabling API logging by default will significantly contribute to the growth of the Registration Server's MySQL database. On a busy site, we recommend to only enable API logging for debugging purposes or to enable the `CleanUp` auto task that removes log entries older than 30 days from the API log table. See *Manage Auto Tasks* (page 14) for details.

4.5.6 Manage Auto Tasks

There is a number of background jobs that are being performed by the yvva-based `td-regserver` service. The individual tasks are explained in more detail in chapter *Auto Tasks* (page 71).

To review and configure these automatic tasks, click **Admin -> Manage Auto Tasks** in the top menu bar. Note that this option is only available to the Default Provider and users having the `MANAGE-TASKS` privilege. In general it's not necessary to change the default values.

You will see a list of currently available tasks, their status and description as well as some run time information.

AUTO TASKS:

Id	Name	Status	Description	Last start time	Last end time	Last result	Procedure text	Frequency	
1	Send Emails	Enabled	Process and send queued email notifications (e.g. invitations, activation notices).	2018-06-01 12:51:34	2018-06-01 12:51:34	OK	TD2RegAutoTask:sendEmails();		Edit
2	Delete Old Messages	Enabled	Delete messages not retrieved by Clients from the message queues within the periods defined in <InvitationStoragePeriod> and <InvitationStoragePeriodFD>.	2018-06-01 12:51:34	2018-06-01 12:51:34	OK	TD2RegAutoTask:deleteOldMessages();		Edit
3	Move Store Forward Messages	Disabled	[DEPRECATED] This task is no longer required, and will be deleted.	2015-11-18 17:24:16	2015-11-18 17:24:16	OK	TD2RegAutoTask:moveSFMessage();		Edit
4	Delete Client IPs	Enabled	Remove client IP addresses from the device table after the period defined in <StoreRegistrationDeviceIPinSeconds>.	2018-06-01 12:51:34	2018-06-01 12:51:34	OK	TD2RegAutoTask:deleteClientIPs();		Edit
5	Update RegServer-List	Enabled	Retrieve the list of known Registration Servers within the TDNS Network.	2018-06-01 12:51:34	2018-06-01 12:51:35	Private/Public key error (Pass phrase required)	TD2RegAutoTask:updateRegServerList();	12h	Edit
6	CleanUp	Disabled	Cleanup task to remove older entries from the Client-Log and API-Log table.				TD2RegAutoTask:cleanUpLogs();	24h	Edit
7	Expire Licenses	Enabled	Check when licenses expire and send emails or disable them as required	2018-06-01 12:51:35	2018-06-01 12:51:35	OK	TD2RegAutoTask:expireLicenses();	12h	Edit
8	CSV Import	Disabled	This task imports user data from CSV files uploaded to the database or the Provider specific hot-folder				CSVImport:startCSVImport();	10m	Edit
9	Delete Providers	Enabled	This task deletes Providers that have been marked for deletion	2018-06-01 12:51:35	2018-06-01 12:51:35	OK	TD2RegAutoTask:deleteProviders();	30m	Edit
10	Send Notifications	Enabled	Send notifications of user change events that could not be sent synchronously	2018-06-01 12:51:35	2018-06-01 12:51:35	OK	RegUser:sendAllNotifications();	5m	Edit
11	License Report	Enabled	Create a report on licenses and usage	2018-06-01 12:51:35	2018-06-01 12:51:35	OK	TD2RegAutoTask:licenseReport();	2h	Edit

To edit a task, click **Edit** next to the desired task. You will see a form that allows you to enable or disable the task and modify some of the task's parameters, e.g. the frequency in which this task will be called.

If no frequency is provided, the task is scheduled to run every time the `td-regserver` background service wakes up (10 seconds by default, as defined in file `/etc/td-regserver.conf`).

We do not recommend to change any other settings of existing tasks or to remove or disable the system's default tasks.

After you are finished, click **Save Task** to save any changes you have made, or **Back** to return to the list of tasks.

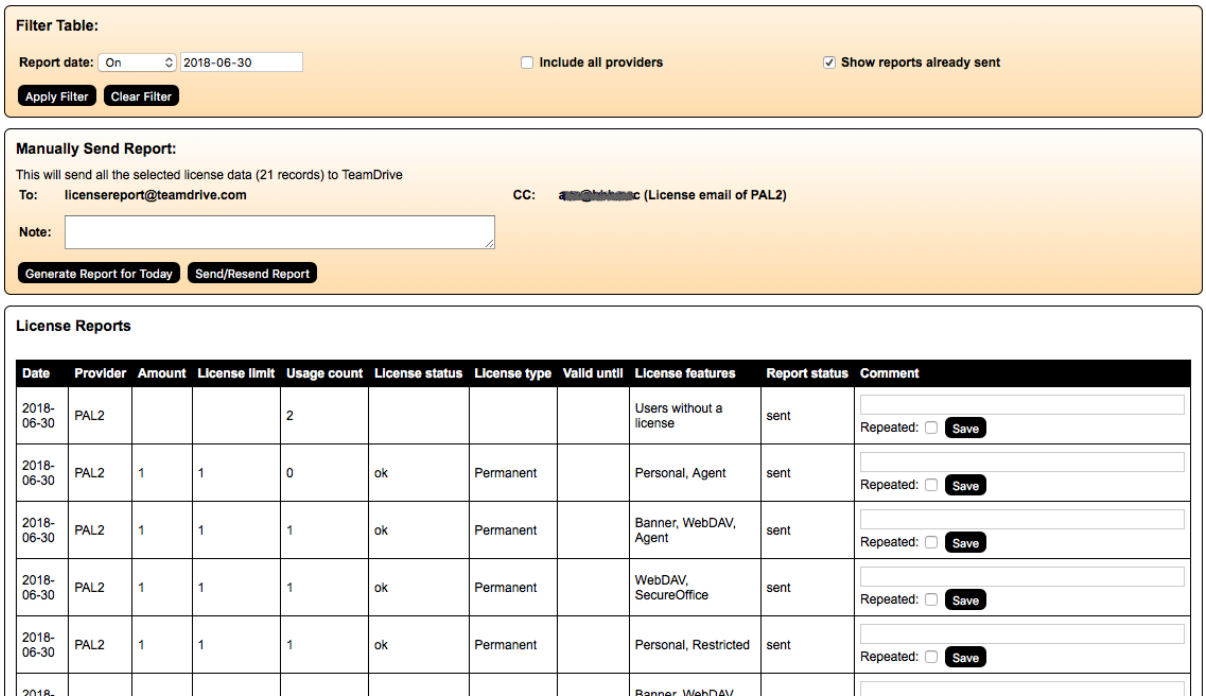
To create a new task, click **Create New Task** on top of the page. Creating new tasks can be necessary to add custom functionality which requires server side processing. New background tasks need to be implemented in yvva code and must be integrated into to Registration Server's code base.

Fill in the form fields with the required values and click **Create Task**.

4.5.7 License Report

License reports are a summary of license usage on a particular day. The Registration Server automatically creates a license reports on twice a month: on the 15th and on the last day of the month.

Select the **License Report** menu item from the **Admin** menu to go the **License Report** page. You must have the `MANAGE-REPORTS` privilege for this option to be available.



The reports are created by the *"License Report" Task* (page 72). The reports are created before 4:00 in the morning, on the day of the report. After 20:00 any reports not yet sent, are automatically emailed to "licensereport@teamdrive.com" and a CC is sent to the provider **License email** address.

Using the Filter Table allows you to select reports by the time created, and whether to include reports that have already been sent. By default you will see all reports that have not been sent.

Manually Send Report

License reports can be sent and/or resent manually. It is also possible to generate an ad-hoc report if one has not already been generated for the current day. Use the **Generate Report for Today** to manually report.

Click the **Send/Resend Report** button to send all the report records that are select in the **License Reports** table below.

You can add a note to be included in the email. In addition, you can add a comment to each line of the report, if the report is a monthly report. Check the **Repeated** checkbox to have a comment repeated automatically in future reports. Click **Save** to save you comment on repeated option for a line.

4.6 Providers

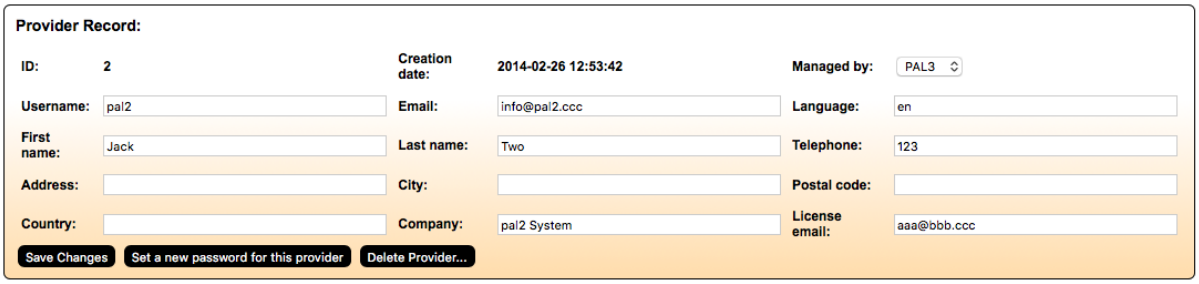
The provider specific data is divided into two parts: the Provider Record with the basic provider informations and the individual settings for the provider to control the features and functionality of the Registration Server.

User's that login with Provider-Level privileges have control of all providers that their provider manages. In this case there is a drop-down menu at the top of the Admin Console page which is used to select the provider. In addition, search filters provide an "Include all providers" option.

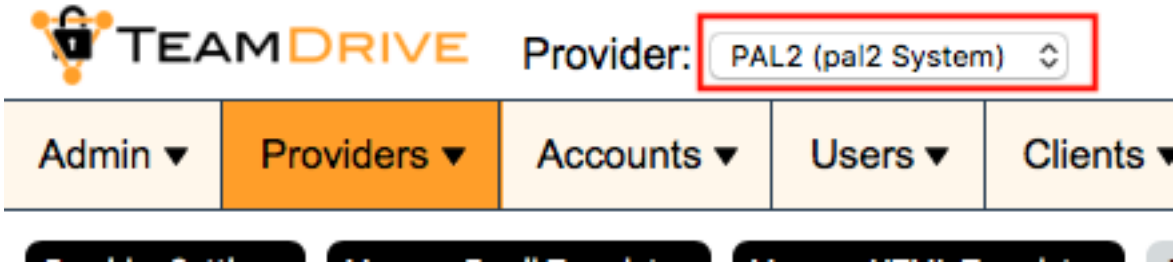
4.6.1 Provider Record

At the top of the **Provider Settings** is the **Provider Record** section which display the details of the provider in a number of editable fields.

If you have the `EDIT-PROVIDER` privilege, the you can change these values and click **Save Changes** to make changes.



If you have access to multiple provider then select the provider you wish to manage in the drop-down menu at the top of the page:



Delete Provider

Deleting a provider will remove all users, licenses and depots belonging to the provider. If you proceed, the selected provider will be scheduled for deletion. The deletion process will start after approximately 30 minutes.

In case your Registration Server is connected to the TDNS (TeamDrive Name Server), contact TeamDrive and request the removal of the selected provider from TDNS. Deletion of the provider will only be completed once the reference to the provider has been removed from TDNS. Once in progress, deletion cannot be undone, i.e. the result is permanent.

4.6.2 Provider Settings

There are a number of provider specific configuration options that can be customized based on your requirements. To edit provider settings, click **Providers/Provider Settings** in the top menu bar.

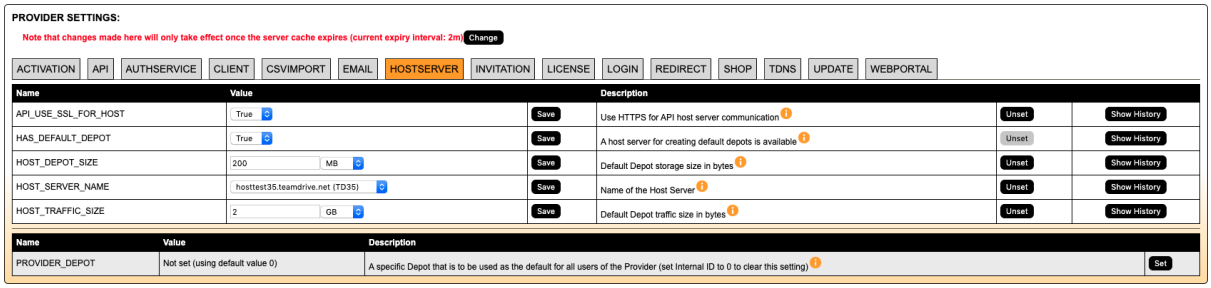
> **Warning:** Changes to the Provider settings will only be active after the caching period defined in `RegServer/CacheIntervall` has passed (the default is 1800 seconds or 30 minutes). If no cache interval was set, you need to restart the Apache HTTP Server of the Registration Server to reload these values.

The lower section of the page shows list of customizable settings for the selected provider, grouped in categories.

The available settings and their function are described in the *Reference Guide* (see provider_settings).

To change a setting, select one of the categories (e.g. **AUTHSERVICE**, **CLIENT**, **EMAIL**, **HOSTSERVER**, etc.). The settings in each group are divided in two blocks:

The upper white marked area with the settings which have been added to the provider. Change the desired setting either by entering a new value or selecting one from the drop down menu, and click **Save** in that value's row. Do not change more than one value at once — always save your change before modifiying another value. Note that not all settings are editable. To remove a setting click **Unset**. The entry will disappear from the upper list and can be found in the lower list now and the pre-defined default value will be used. Note that not all settings can be removed.



The lower grey marked area has additional settings which are currently not set for the provider. These settings use the pre-defined default values.

To change the default value, click on **Set** to add this setting to the provider and change the value as described above in the upper list. If you added all available settings, the grey marked box will disappear.

4.6.3 Manage Email Templates

The Registration Server is shipped with the default set of email templates located in `/opt/teamdrive/regserver/setup/templates/email`.

A new created provider will use the default templates from the file system.

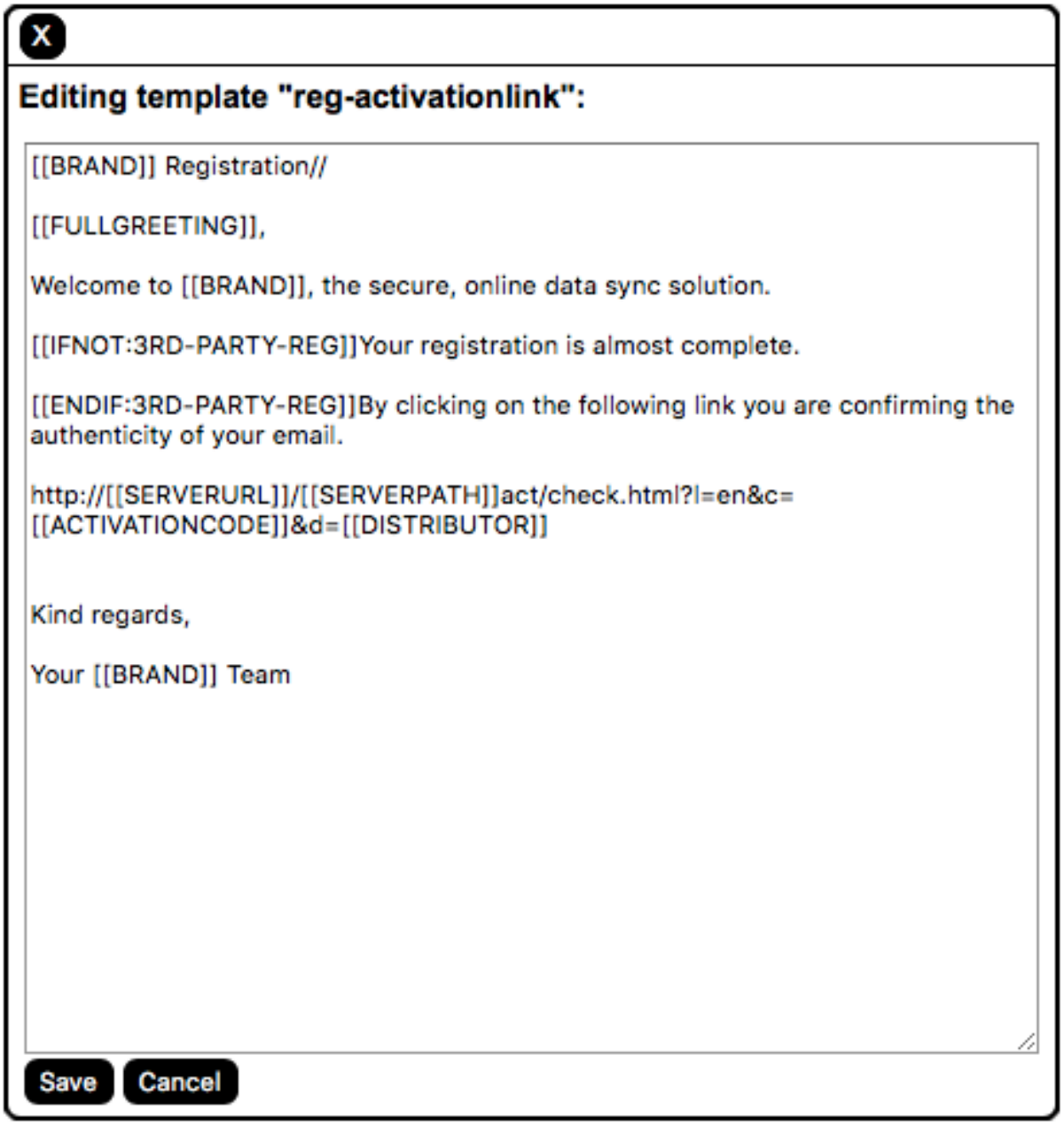
The templates are combined into groups for a better overview. A few groups will be hidden by default if they are not required due to the current provider settings. For example: The mail templates in the group `USER-INVITE-USER` are only necessary, if you define a value for the provider setting `INVITATION/PROMOTION_UPGRADE`. Using the button **Show** you could make the templates visible even you are not using them.

The templates are combined into the following groups:

- CLIENT-INTERACTION: This is the default set of templates which are necessary in all cases. They are important for the client interaction like recieving the activation mail, sending invitation mails to other users and for password and email changes.
- TRIAL-LICENSE: Only necessary if you offer trial licenses to your users.
- USER-INVITE-USER: Only necessary if you offer a referral program for your users.
- SERVER-ADMINISTRATION: These templates will only be used for the server setup and two-factor authentication in the Admin Console.
- API: Only necessary if you will offer an own web interface for your users and you will use the Registration Server API to allow users to register and manage their accounts. Will also be used by the adminconsole in case of changing the email address or password. You have to allow sending mails from the API using the provider setting `API/API_SEND_EMAIL`
- API-LICENSE-CHANGES: Only necessary if you use the API and you want to send confirmation mails for license creation and changes.
- GROUPS: Emails for inviting users to groups.

The provider can edit the default templates by clicking **Edit** next to each template to open it in an editor window. The templates use placeholders which have the form: `[[ ... ]]`. The placeholders are replaces by appropriate values when the template is processed before being sent.

You will find a list of all macros in the chapter templates_for_client_actions.



By saving the changes, the modified template will be stored in the database for this provider and the default template in the file system will no longer be used.

The templates are language specific. For each language you wish to support you have to create a set of email templates. The supported languages for the mail templates will be defined in the provider setting `EMAIL/EMAIL_ALLOWED_LANG`.

4.6.4 Manage HTML Templates

The Registration Server is shipped with the default set of HTML templates located in `/opt/teamdrive/regserver/setup/templates/html`.

For the HTML templates the Registration Server is using the same logic as for the email templates. A newly created provider will use the default template from the file system as long as the provider has not modified the template. Modified templates are stored in the database.

HTML templates are language specific. The supported languages for them will be defined in the provider setting `LOGIN/ACTIVATION_ALLOWED_LANG`.

There are three main template groups:

- activated-*: HTML templates to activate a client installation
- newemail-*: HTML templates to confirm email changes in the client
- portal-*: HTML templates for the web- and 2-factor-authentication (see *How to Setup Two-Factor Authentication* (page 69))

4.6.5 CSV User Imports

In addition to adding users manually, you can automatically create multiple users by importing them via CSV files (which can be created by extracting the user data from another directory service or user information source).

This requires that CSV import is enabled and configured correctly in the provider settings. See chapter *Importing Users via CSV Files* (page 39) for more details on the configuration of the CSV import functionality and the structure of the CSV file.

Since users can only imported directly to a provider, you require Provider-Level privileges in order to have access to this feature.

Upload CSV File

To upload a CSV user list via the Admin Console, go to the **Providers** menu and select the **CSV User Imports** item, or click the button after selecting the **Providers** menu item. Make sure that you have selected the correct provider in the provider drop-down at the top of the page (if you have the right to manager multiple providers).

Clicke the "Choose File" button in the **Upload CSV File** section. A file selection dialogue will pop up, allowing you to select a local CSV file to upload (only files with .txt and .csv extensions will be selectable). Choose a file and click **Open**. To upload the file, click the **Upload File** button. After the upload has finished, you will see confirmation as to whether the upload was successful or if any errors occurred.

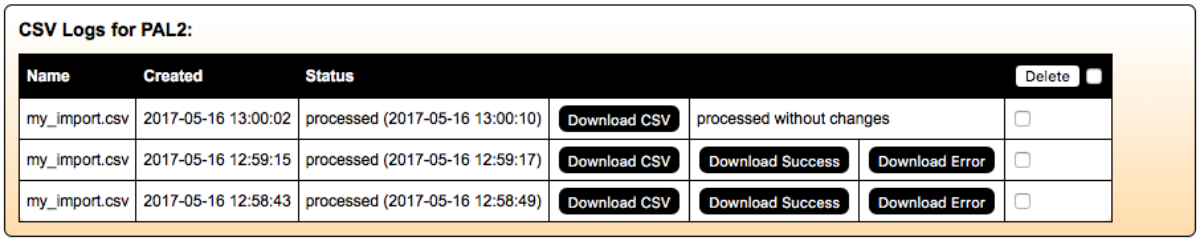
The uploaded file will appear in the **CSV Logs** section with the "wait for processing" status. The file will be processed the next time the import autotask runs. See *"CSV Import" Task* (page 72) for details.

CSV Logs

When data is imported from a CSV file, an import log is created. This log contains information about the success/failure of the import.

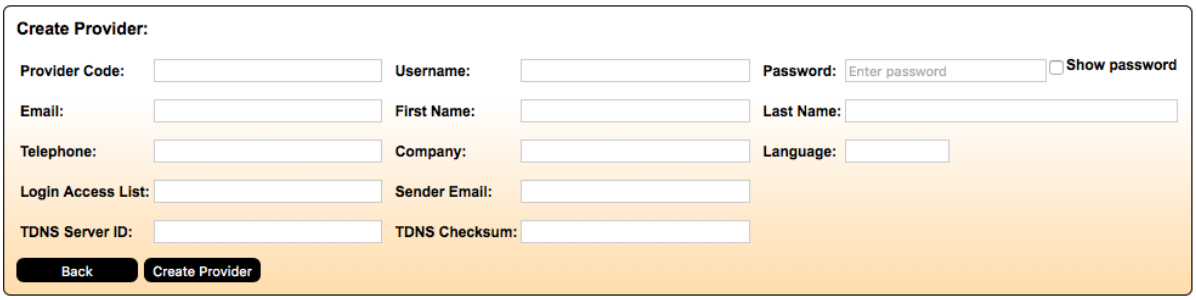
Navigate to the **Providers / CSV User imports** section to view a list of all uploaded CSV files, their status and the log output of the previous import run.

A page will come up that lists available logs. Each uploaded file can be downloaded again by clicking **Download CSV**. The status of each log indicates whether the import was successful, and at what time the log was created and processed. Click **Download Success** or **Download Error** to download a log of the successful or failed import. Click **Delete** to remove CSV files.



4.6.6 Create Provider

A new provider can be added by clicking **Create Provider**. You must have `CREATE-PROVIDER` privileges for this function.



In order to add a provider you need a valid and provider code. Provider codes must be registered on the TDNS (TeamDrive Name Server) before they can be used to create a new provider. Contact TeamDrive in this regard if you need to create a provider. You will then be supplied with the **TDNS Server ID** and **TDNS Checksum** values.

Please provide values for the following required fields: **Username**, **Password**, **Language**, first and last name, **Company** and **Sender email**.

Telephone and **Login Access List** are optional.

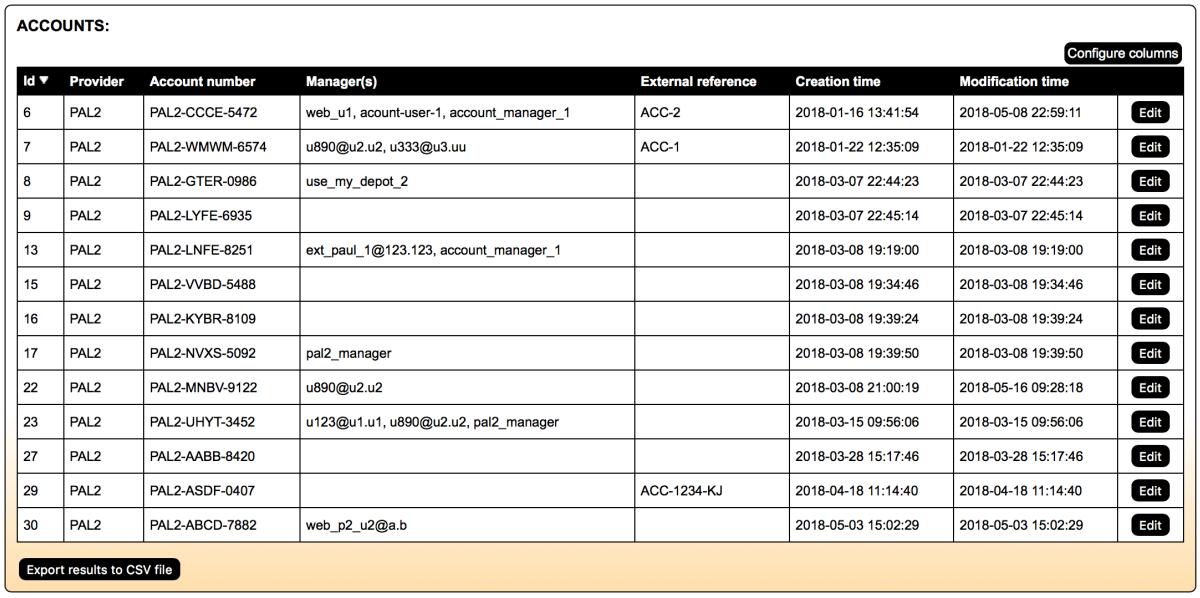
4.7 Accounts

Accounts are a collection of users, licenses and depots that can be managed as a unit. An account is owned by a provider and has one or more managers that need not be members of the account. Accounts are solving the problem, that users cant be deleted without deleting their depot and license (or moving them to an other user before). Managers of an account can be added or deleted without any effects to the depot and license of the account.

The owner of the resources is automatically manager of the account and is granted `LOGIN-RIGHT` and `MANAGE-ACCOUNT` privileges (see *User Rights* (page 28)) which includes several account managing rights. Account managers are allowed to login at the admin console and manage their account users, licenses and depots. But they are only able to see their own account users and have no possibility to see other users.

4.7.1 Manage Accounts

Selecting the **Account** menu item will take you to the **Manage Accounts** page. Here you find a list of all accounts to which you have access.



ACCOUNTS:

Configure columns

Id ▼	Provider	Account number	Manager(s)	External reference	Creation time	Modification time	
6	PAL2	PAL2-CCCE-5472	web_u1, acount-user-1, account_manager_1	ACC-2	2018-01-16 13:41:54	2018-05-08 22:59:11	Edit
7	PAL2	PAL2-WMWM-6574	u890@u2.u2, u333@u3.uu	ACC-1	2018-01-22 12:35:09	2018-01-22 12:35:09	Edit
8	PAL2	PAL2-GTER-0986	use_my_depot_2		2018-03-07 22:44:23	2018-03-07 22:44:23	Edit
9	PAL2	PAL2-LYFE-6935			2018-03-07 22:45:14	2018-03-07 22:45:14	Edit
13	PAL2	PAL2-LNFE-8251	ext_paul_1@123.123, account_manager_1		2018-03-08 19:19:00	2018-03-08 19:19:00	Edit
15	PAL2	PAL2-VVBD-5488			2018-03-08 19:34:46	2018-03-08 19:34:46	Edit
16	PAL2	PAL2-KYBR-8109			2018-03-08 19:39:24	2018-03-08 19:39:24	Edit
17	PAL2	PAL2-NVXS-5092	pal2_manager		2018-03-08 19:39:50	2018-03-08 19:39:50	Edit
22	PAL2	PAL2-MNBV-9122	u890@u2.u2		2018-03-08 21:00:19	2018-05-16 09:28:18	Edit
23	PAL2	PAL2-UHYT-3452	u123@u1.u1, u890@u2.u2, pal2_manager		2018-03-15 09:56:06	2018-03-15 09:56:06	Edit
27	PAL2	PAL2-AABB-8420			2018-03-28 15:17:46	2018-03-28 15:17:46	Edit
29	PAL2	PAL2-ASDF-0407		ACC-1234-KJ	2018-04-18 11:14:40	2018-04-18 11:14:40	Edit
30	PAL2	PAL2-ABCD-7882	web_p2_u2@a.b		2018-05-03 15:02:29	2018-05-03 15:02:29	Edit

Export results to CSV file

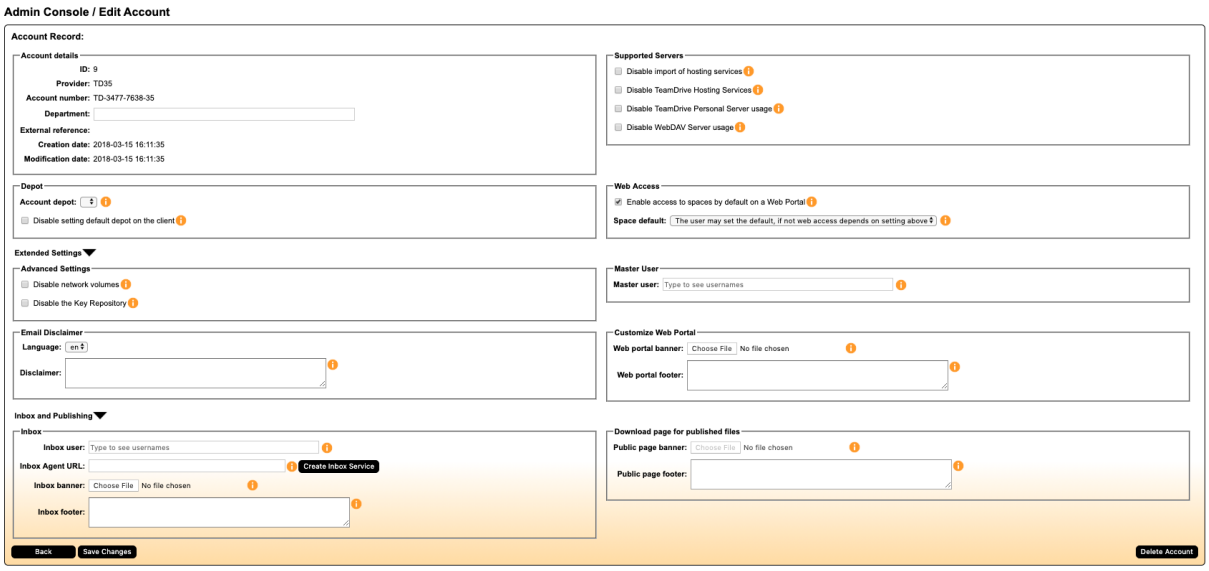
Use the **Filter Table** section to narrow the selection down to the accounts you are interested in if necessary. A "contains", case-insensitive search us done for the values entered in the **Account number**, **External reference**, **Account manager** and **Department** fields.

If you have access to multiple providers, then the **Include all providers** checkbox allows you to search and list accounts from all providers under your control.

Click the **Edit** button in order to view details, and edit an account. This will take to the **Edit Account** page.

4.7.2 Edit Account

On the **Edit Account** page you can view and change all aspects an account.



Account Record

In this section you can set a number of account level options:

- **Master user:** A master user is a user that is automatically invited to all spaces created or joined by the user's of an account. If the master user is run by a TeamDrive agent, then you should set **Enabled auto-accept invitations** for the user (see *User Record* (page 26)).

- **Disable network volumes:** This option prevents users from creating spaces on network volumes.
- **Disable the Key Repository:** Use this option to disable the Registration Server key repository for all users in the account. Note that users that are not using the key repository need to explicitly invite themselves to spaces when they install a new device. They also have to manually backup their space keys backup file which is located in the SpacesBackups folder. Without this file the user cannot rejoin his spaces.
- **Account depot:** Here you can specify a depot belonging to the account as the "account depot". The account depot is automatically distributed to the all devices of all users in the account. It is also set to be the "selected depot" (see below).
- **Disable setting default depot on the client:** Check this option if you want to prevent users from permanently changing the default depot on client devices. The client-side default depot, is the depot that is used to create spaces if no other depot is explicitly selected.

  If the user has not changed the default in the client, then the "selected depot" which is specified by the Registration Server will be set to the default depot on client devices.
- **Disable import of hosting services:** The prevents users from importing TeamDrive Hosting Service access credentials for a particular depot on client devices. This ensures that the only depots that users have access to (for space creation) are those distributor by the Registration Server.
- **Disable TeamDrive Hosting Services:** This disables the creation of spaces on TeamDrive Hosting Services which includes all depots managed by the Registration Server. If Hosting Services are disabled, users must use alternative storage such as a WebDAV server, or the TeamDrive Personal Server (TDPS). This does not prevent users from joining existing spaces that are hosted by TeamDrive services.
- **Disable TeamDrive Personal Server usage:** This option prevents users from adding TeamDrive Personal Server (TDPS) access credentials to a client device, and from creating spaces on a TDPS.
- **Disable WebDAV Server usage:** This option prevents users from adding WebDAV server access credentials to a client device, and from creating spaces on a WebDAV server.
- **Inbox user:** An "inbox" is a published page which accepts anonymous uploads into a space folder. The inbox can be used hosting a stand alone TeamDrive Agent or using the Inbox Service hosted by the Web-Portal (version 2.0.1 requiered). Inboth cases create you need to create an "inbox user" and assign the user a license with the **inbox** feature flag. The name of the inbox user must be specified here which must be a memeber of the account. For the stand alone version deploy a TeamDrive agent with a login as this user and the agent will then publish the page. For the Inbox Service login with the user credentials and the Admin Console will setup the inbox on the WebPortal server.
- **Inbox Agent URL:** This is the URL that references the TeamDrive agent that published inbox pages. The TeamDrive agent must be running as the user specified as the **Inbox user**. Once you have set this value, users of the account are able to create inbox pages for any folder after inviting the inbox user to the space.
- **Inbox banner** An optional banner image which is displayed when opening the Inbox URL. To remove an existing image, upload an empty file with a size of 0 byte.
- **Inbox footer** An optional footer text or html which will be displayed at the bottom of the Inbox.
- **Public page banner** An optional banner image which will be displayed for published files in the spaces. The banner image will be set on the Hosting Server for all Depots of the account. To remove an existing image, upload an empty file with a size of 0 byte.
- **Public page footer** An optional footer text or html which will be displayed at the bottom of the Inbox.

Account Members

This section contains a list of account members and managers. Users may only be a member of one account. Click the **Edit** button in each line to go to the **Edit User** page of a user. The **Remove** button removes the user from the account, but does not delete the user.

Note: If you are an account manager, you may not have the privileges to add the user to the account again. Contact your provider which can add the user to your account again. The **More Info** button will reveal the licenses and

devices belonging to the users.

If the user is a member and a manager, then you have the option to either remove the user completely, or just as a manager of the account.

Click the **Add Member** to add an existing user as a member to the account. In general, adding existing users to an account requires Provider-Level privileges. Users that are already a member of another account cannot be added to an account before they are removed from their current account.

Click the **Add Manager** to add a user as manager to the account. Users that are already members of the account may also be added as manager.

Account Licenses

This is a complete list of all licenses that belong to the account. Licenses can either belong to an account (as well as a provider). If you add a license to an account that belongs to a user, the ownership of the license is transfered from the user to the account.

Click the **Edit** button in each line to go to the **Edit License** page to view all details of the license, and to modify the license parameters if you have the require privileges.

The **Remove** button in each line removes the license from the account, without deleting it. This will leave the license without an owner besides the provider.

Note: If you and are account manager you may not have the privileges to add a license again, after you have removed.

The **Add License** button brings up a list of licenses that you have access to that may be added to the account. "Default licenses" that are single user licenses, may not be added to an account, because they are created exclusively for a particular user. When you add a license to the account, user ownership of the license is removed. Provider ownership of the license is never removed.

If you have the `CREATE-LICENSE` privilege then the **Create License** button takes you to the **Create License** page where you are able create a new license for the account, with the features and attributes you require.

If you have the `PURCHASE-LICENSE` privilege and your provider has an associated shop, then a **Purchase License** button is made available which will take you to the appropriate shop page for purchasing licenses for the account.

Account Depots

This is a complete list of all depots that belong to the account. Unlike licenses, depots may also have a user that is owner of the depot. This user always has access to the depot, even when the depot is not explicitly in-use by the user.

Click the **Edit** button in each line to go to the **Edit Depot** page to view all details of the depot, and to modify the depot parameters if you have the require privileges.

The **Remove** button in each line removes the depot from the account, without deleting it. If the depot still belong to a user of the account, then account managers still have the required privileges to add the depot back to the account.

The **Add Depot** button brings up a list of depots that you have access to that may be added to the account. Unlike "default licenses" you may add depots created by default to the account. Any depots added to an account will not remove the user level ownership, which remains as is.

If you have the `CREATE-DEPOT` privilege then the **Create Depot** button takes you to the **Create Depot** page where you are able create a new depot for the account, with the parameters you require. Note that this requires at least one TeamDrive Hosting Service to be registered with the provider.

If you have the PURCHASE-DEPOT privilege and your provider has an associated shop, then a **Purchase Depot** button is made available which will take you to the appropriate shop page for purchasing depots for the account.

4.7.3 Create Account

If you have the CREATE-ACCOUNT privilege then **Create Account** page is available to you.

To create an account you have to specify an **Acccount code**, which is must consist of 4 characters. One leading capital letter (the letters from 'A' to 'Z') and 3 additional letters or numbers from 0 to 9. Optionally you can add a manager and a number or members when creating the account.

4.8 Users

Your login credentials determine the level at which you manage users. Superuser-Level allows you to manage all users registered on the server. At Provider-Level you manage the users of one or more providers. At the Account-Level you manage the users of one or more accounts, and at the User-Level you only manage yourself.

Click on the **Users** menu item to bring up the **Manage Users** page.

4.8.1 Manage Users

By default, all users are listed. You can narrow down the search by typing in search criteria in the **Filter Table** section at the top of the page, and then clicking **Apply Filter**.

Click **Clear Filter** at any time to go back to displaying all available users.

When filtering results, you can use the percent character ('%') as a wildcard: for example, entering 'john%smith' into the email field will match users with an email like john.smith@td.net, johnsmith@shaw.net, johnDoeSmith@gmail.com, etc.

Depending on the number of results, there may be more than one page of output. Click the numbers and arrows above the table to browse through results. To sort the table by a column value, click the column's name in the title row.

Click on **Force All to Re-login** in order to force all users selected by the current filter to re-login. This means that users will be promoted on all installed client devices to enter their password. Since this can cause some disruption of the service as user's forget there passwords, you are asked to confirm this action.

Click **Configure Columns** to bring up a dialogue that allows you to customize the table output. Select the columns that should be displayed and click **Update** to update the table view.

Click **Export results to CSV file** at the bottom of the result list if you want to save the resulting table output into a comma-separated text file. Your web browser will prompt you for a file name under which the file will be stored locally.

Click the **More Info** button at the end of a user's row of information to view the user's licenses and device details. Click **Less Info** to hide this information again.

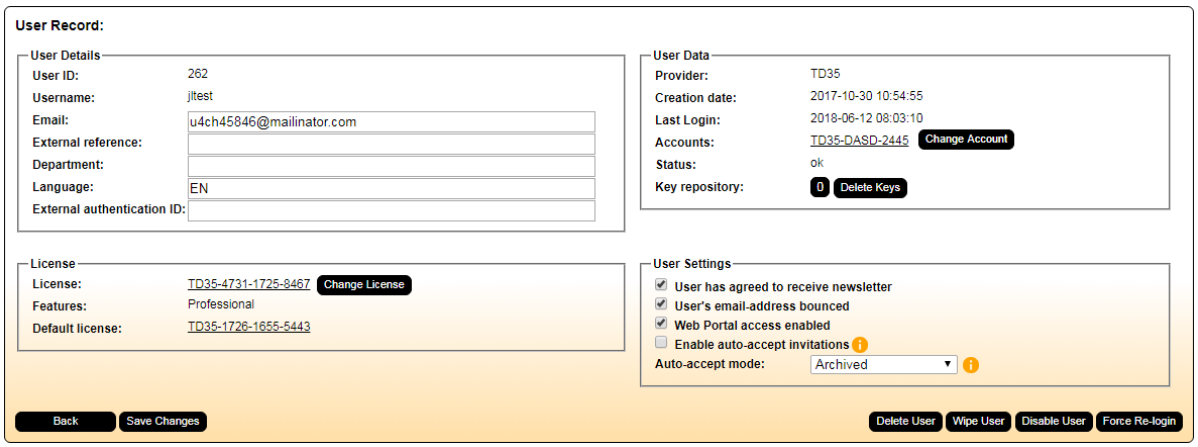
Click the **Edit** button next to a user's email address to open up the user details page, which displays all of the user's information, including licenses and the user's devices in more detail.

4.8.2 Edit User

The **Edit User** page is divided into several blocks and will show user information about:

- User Record
- User Devices
- User Licenses
- User Depots
- User Rights
- Change Provider

User Record



If a newly registered user has not been activated yet (**Status** is set to **not activated** in the user's record details), you can activate the user manually by clicking **Activate User**. If the user was already activated, this option will not be displayed.

You can view and change the user's details like email address, external reference, department or the preferred language. Click **Save Changes** to commit any changes you made to these fields.

You can move the user to a different provider (only possible for the default provider) by clicking **Set New Provider**. You can define if the user will get a new depot and license based on the new provider default settings.

You can temporarily disable a user by clicking **Disable User**. If you disable a user, the user's client devices will receive a notification from the Registration Server and will inform the user about the deactivation. At this point the client disables all functionality and activity and the user can no longer use the TeamDrive service (e.g. creating spaces, inviting users, etc.) until the user has been enabled again (access to the spaces in the filesystem is still possible).

Clicking **Wipe User** will wipe all of the user's devices, delete the user's key repositories, and disable the user. The devices of the user will delete all local data (space directories in the filesystem, caches, registration information) and will delete itself in the Registration Server' database. Licenses and depots will be preserved.

Clicking **Delete User** will delete the user record and all of the user's devices. Additionally, you can choose to delete the user's depots and licenses by selecting the appropriate checkboxes in the confirmation dialogue.

You can reset a user's password by clicking **Invalidate Password** in the bottom right-hand corner. The user's client devices will then automatically logout and ask the user to request a new temporary password which must be used to login and specify a new password.

If the provider of the user is using an external authentication service, then **Invalidate Password** will just force the user to re-login.

Return to the main user list at any time by clicking **Back** in the bottom left-hand corner.

User Devices

The device list shows information about all of the user's installed TeamDrive clients with details including: status, creation and last active times, IP address during installation, the client software version, platform and number of pending messages from other users. Clicking the message number (if the value is greater than zero) displays a list of users that sent messages to this device.

Please note that it is normal for inactive devices to have pending messages, these messages will be picked up when the device becomes active again or will be automatically deleted if `InvitationStoragePeriod` is reached.

Devices will not longer be sent messages if the are inactive for the time specified by the global `InviteOldDevicesPeriodActive` server setting. In addition, messages posted to devices are automatically deleted once the message store reaches the value specified by the `InvitationStoragePeriod` server setting.

You can delete one or multiple devices by checking the **Delete** checklist item for the device(s) in the **User Devices** section and clicking the **Delete** button on top of the column.

Note: A deleted device can be re-activated by the user. If you don't want the user to re-activate his installation, you have to deactivate the user.

The **Wipe Device** functionality deletes the device's entry in the Registration Server database after the client software has confirmed that all local data were deleted successfully (space directories, caches, registration information).

User Licenses

This section shows all licenses that are either owned and/or in use by the user. This is indicated in the **Usage** column. If the user has a "default license", then this is shown in the **Status** column.

Users are given a default license, if no other license is assigned to the user when the user is registered. The features of the default license are determined by provider settings (see default_free_feature and default_account_feature).

A license that is in use by the user may also be in use by other users as indicated by the **Used/Limit** column.

Clicking the **Edit** button will take you license details page (see *Licenses* (page 33) for more details).

Clicking the **Remove** button removes ownership of the license from the user. In order to change the license the user has in use, click the **Change License** button in the **User Record** section. A license that is no longer in use, can be assigned a different user.

If you have the `CREATE-LICENSE` privilege then the button **Create new license for ...** will open the license creation page. See *Creating License* (page 35) for details.

User Depots

If the user has depots on a TeamDrive Hosting Sevice then the information is displayed here. The **Usage** column indicates whether the user is owner of the depot and whether it is in use, or is the user's "default depot".

A user may have a number of depots in use. One of the depots may be "selected". The selected depot is the selected by default on the client devices. As a result, unless a user specifically selects a different default on the client device, this depot will be used when creating spaces. However, a user is also free to select any other depot that is "in-use".

The "default depot" is the depot that is created automatically when the user is registered. Whether this is done, depends on a provider setting (see has_default_depot and api_create_default_depot). If there is no "selected" depot, then the default depot is considered to be selected.

Depot information for the user is retrieved from the Host Server via an API call. This information is stored by the Registration Server, but will be "refreshed" if it has not been retrieved for a while. The last retrieval time is indicated in the **Last update** column, where it is possible to manual refresh the depot information.

In order to see all information pertaining to a depot, click in the **Edit** button. This will take you to the depot details page which display all details of the depot including the change history and space list retrieved from the Host Server.

On this page you can also delete or deactivate a depot if you have the required privileges.

The button **Deactivate Depot** allows you to temporarily disable a the depot on the Host Server. The client devices will no longer be able to synchronize the spaces contained in this depot (the spaces will be marked as "Disabled"), until you click **Activate Depot** again.

Clicking **Create new depot for ...** brings up the depot creation page. See *Create Depot* (page 35) for more details.

Clicking **Open Host Admin** (only available if you have provider level privileges) opens the respective TeamDrive Host Server's administration console in a new browser window/tab. You will be required tologon with valid Host Server credentials. Please refer to the Host Server documentation for more information.

User Rights

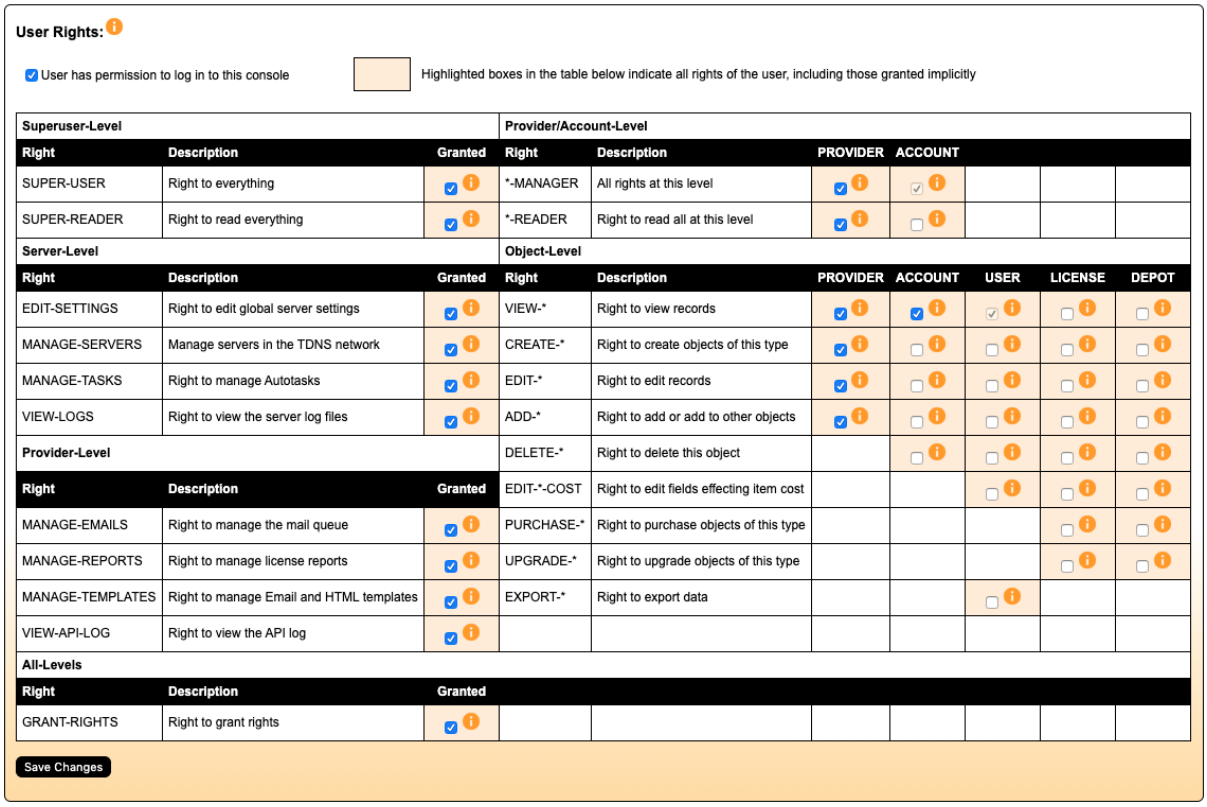
Depending on what user you log in as, you have different rights and privileges.

When you log in using provider credentials, you are automatically granted rights depending on whether you are the Default Provider or not. The Default Provider is granted the `SUPER-USER` privilege, which includes all rights. Other providers are granted the `PROVIDER-MANAGER` privilege, which includes all rights except: `SUPER-USER`, `SUPER-READER`, `EDIT-SETTINGS`, `MANAGE-SERVERS`, `MANAGE-TASKS`, `VIEW-LOGS` and `CREATE-PROVIDER`

By default, standard users cannot login to the Admin Console unless they are an account manager. When an account manager logs in he/she is automatically granted the `MANAGE-ACCOUNT` privilege.

Other users must be explicitly granted the `LOGIN-RIGHT` privilege in order to access the Admin Console.

You grant the `LOGIN-RIGHT` privilege by clicking on the checkbox with the title: **User has permission to log in to this console**. The privilege details of the user are then displayed.



After the box is checked, a list of additional available rights are displayed. The rights that are enabled depend on your own privileges — you can only grant/revoke rights that you possess yourself.

In order to grant or revoke privileges, check or uncheck the desired privileges you want to assign to or remove from the user and click **Save Changes** to apply the changes. Note that it is not possible to remove privileges that are automatically granted. For example, the SUPER-READER grants all read privileges for the entire server. Removing one aspect, such as VIEW-LICENSE is not possible.

Whether you are able to view or manipulate an object (provider, account, user license or depot) depends on two things:

- whether you have access to the object, and
- whether you have the required privilege.

What objects you have access to depends firstly on your privilege leve. Superuser-Level grants access to all objects. At Provider-Level you have access to objects belonging to the providers you control. At Account-Level you may access the users, devices, licenses and depots that belong to the account. Finally, at the User-Level you only have access to objects belonging directly to the user.

However, as noted above, access is not sufficient to either view, create or manipulate an object. For this, you must also have the required privilege. All users are automatically granted the VIEW-USER privilege. This means that, at the very least, if a user has login privileges, then the user is able to view his/her own user record.

All available privileges are described in the sections below.

- **Superuser-Level Privileges:**

  SUPER-USER This is the privilege level of the Default Provider. In other words, users that login as the Default Provider are automatically granted SUPER-USER rights. Users with this right have **all** other rights (explicit grants are not required. This means the user can view/edit/delete and create records that are associated with all providers, and also has the **Server-Level** privileges.

  SUPER-READER This is similar to SUPER-USER rights, but not include the right to change anything.

  The following are **Server-Level** privileges:

  These privileges are must be granted explicitly to any user that does not have SUPER-USER privileges.

  EDIT-SETTINGS This means that the user can edit server-wide settings (the **Edit Settings** menu). By default, only the Default Provider has this privilege. See *Server Settings* (page 10) for details.

  MANAGE-SERVERS The user has access to the **Manage Servers** page where he can en-/disable communication between the own Registration Server and all other servers available in the TDNS network. See *Manage Servers* (page 11) for details.

  MANAGE-TASKS User can access the **Manage Auto Tasks** page. See *Manage Auto Tasks* (page 14) for details.

  VIEW-LOGS User can access the **View Server Logs** page. See *View Server Logs* (page 13) for details.

- **Provider/Account-Level Privileges:**

  PROVIDER-MANAGER Users that login as a provider are automatically granted this right. A user with this privilege has all rights accept the **Superuser-Level** privileges, the **Server-Level** privileges and the CREATE-PROVIDER right.

  PROVIDER-READER This right is similar to PROVIDER-MANAGER but the user does not have the right to modify or create any objects.

  ACCOUNT-MANAGER This privilege is granted automatically to all managers of an account. It includes the following:

  - VIEW, *EDIT*‘ and ADD privileges for accounts, users, licenses and depots.
  - DELETE privilege for users, licenses and depots.
  - PURCHASE and UPGRADE privileges for licenses and depots.
  - CREATE-USER and GRANT-RIGHTS privileges.
  - All **Provider-Level** privileges (see below).

ACCOUNT-READER This right is similar to ACCOUNT-MANAGER but prevents any changes from being made to the account or its associated objects. An account manager that is granted ACCOUNT-READER, loses the automatically granted LOGIN-RIGHT and ACCOUNT-MANAGER privileges.

- **Provider-Level Privileges:**

These privileges are included as part of the PROVIDER-MANAGER privilege. Users with the PROVIDER-READER privilege can view the associated pages, but may not make changes.

MANAGE-EMAILS User can access the **Manage Emails** page to administer the email out queue. See *View Mail Queue* (page 13) for details.

MANAGE-REPORTS User can access the **Admin / License Reports** page. See *License Report* (page 15) for details.

MANAGE-TEMPLATES User can access the **Providers / Manage Templates** page. See *Manage Email Templates* (page 18) and *Manage HTML Templates* (page 19) for details.

VIEW-API-LOG User can access the **View API Log** page. See *View API Log* (page 14) for details.

- **Object-Level privileges:**

ADD-ACCOUNT This right is required to add users, licenses and depots to the account.

The ADD-ACCOUNT right is also required to remove users, licenses and depots from an account. Note that, to remove a user from an account you also need the ADD-USER right, to remove a license you need the ADD-LICENSE right and to remove a depot you need the ADD-DEPOT right.

ADD-DEPOT This is the right to add the depot to the "in-use depot list" of a user, and to add the depot to an account.

Adding a depot to a user also requires the EDIT-USER privilege, and adding a depot to an account requires the ADD-ACCOUNT privilege.

ADD-DEPOT is also required to remove a depot from a user and from an account.

ADD-LICENSE This right is required to set a license as in-use by a user and to add a license to an account.

Adding a license to a user also requires the EDIT-USER privilege, and adding a license to an account requires the ADD-ACCOUNT privilege.

ADD-LICENSE is also required to remove a license from a user and from an account.

ADD-PROVIDER This right is required to add users, licenses and depots to a provider.

The ADD-PROVIDER right is also required when creating an account.

ADD-USER This right is required to set the user as an owner of an existing license or depot, and to add a user a provider or an account.

CREATE-ACCOUNT Right to create new accounts.

CREATE-DEPOT The right to create new depots.

You must also have *ADD-USER'* rights and access to the user that is assigned as owner of the account.

To create a depot for an account you also need ADD-ACCOUNT privilege and an account level access privilege such as ACCOUNT-READER (and you must be manager or member of the account).

See *Create Depot* (page 35) for details.

CREATE-LICENSE The right to create new licenses.

To create a license for a provider (i.e. not for a specific user or account) you need the ADD-PROVIDER right, and a provider level privilege, such as PROVIDER-READER.

To create a license for an account you also need ADD-ACCOUNT privilege and an account level access privilege such as ACCOUNT-READER (and you must be manager or member of the account).

If a user is assigned as owner of the license (which is required if the license is not created for a provider or an account) you must also have *ADD-USER'* rights and access to the user.

Creating license must be enabled (see allow_create_license) for the provider. See *Licenses* (page 33) for further details.

CREATE-PROVIDER Right to create a new provider.

CREATE-USER This is the right to create new users.

To create a user for a provider you also need the ADD-PROVIDER right, and a provider level privilege, such as PROVIDER-READER.

To create a user for an account you also need the ADD-ACCOUNT privilege and an account level privilege such as *ACCOUNT-READER*' (and you must be manager or member of the account).

These rights are also required in order to upload CSV files.

DELETE-ACCOUNT Right to remove an account.

DELETE-DEPOT Right to delete a depot.

DELETE-LICENSE Right to delete a license.

DELETE-USER Right to remove a user and associated data.

EDIT-ACCOUNT Right to edit accounts data, including adding existing objects to the account.

EDIT-DEPOT Right to edit depots and to set the depot owner and add users to the depot. Setting the owner and adding users additionally requires the ADD-USER right.

This does not include the rights to change the fields covered by the EDIT-DEPOT-COST right.

EDIT-DEPOT-COST Right to edit cost sensitive depot fields, including: storage limit, traffic limit and external reference.

EDIT-LICENSE Right to edit license details, and set the owner of the license. In order to set an owner you also need ADD-USER rights.

This does not include the rights to change the fields covered by the EDIT-LICENSE-COST right.

Managing license must be enabled (see allow_manage_license) for the provider. See *Licenses* (page 33) for further details.

EDIT-LICENSE-COST Right to edit cost sensitive license fields, this includes: license type, features, user limit and external reference.

EDIT-PROVIDER With this right, a user can edit the details and settings associated with the selected provider (the **Providers / Provider Settings** menu item). See *Provider Settings* (page 17) for details.

EDIT-USER This is the right to edit a user, and to assign licenses and depots for usage by the user. EDIT-USER is also required to remove license and depots used by a user.

The right is also required in order to upload CSV files as well as the CREATE-USER right (to import user records). See *Manage Users* (page 25) for details.

EXPORT-USER User can export the user table to CSV fromat.

PURCHASE-DEPOT Right to purchase a depot in the associated shop.

PURCHASE-LICENSE Right to purchase a license in the associated shop.

UPGRADE-LICENSE Right to purchase an upgrade to the license user limit.

UPGRADE-DEPOT Right to purchase an upgrade to the depot storage limit.

VIEW-ACCOUNT Right to view the records an the account page.

VIEW-DEPOT Right to view all records on the **Depots** page.

VIEW-PROVIDER Right to view the records on the provider page.

VIEW-USER Right to view all records on the **Users** page.

VIEW-LICENSE Right to view all records on the license page.

- **All-Level Privileges:**

GRANT-RIGHTS The user is able to modify the permissions of other users. Note that even with this right, users can only grant/revoke rights that they have themselves.

Users can revoke there own rights, but note that there is no way to regain these privileges once they have been removed.

LOGIN-RIGHT This right is required in order to login to the Admin Console. This right is automatically granted to account managers and users that login with provider credentials.

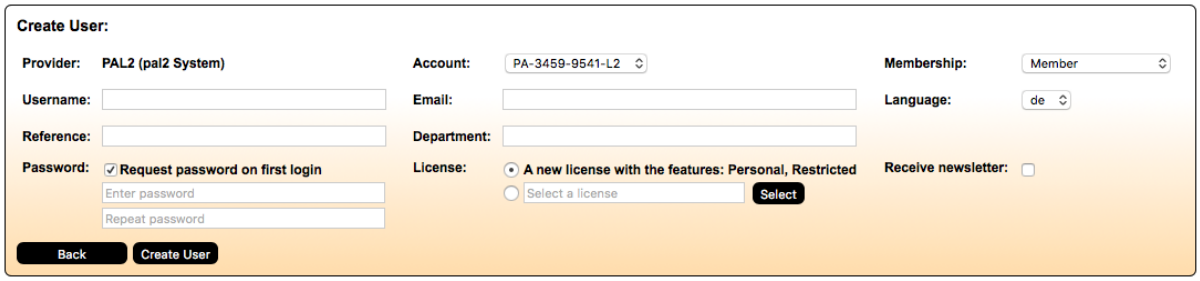
Change Provider

If you control more than one provider, you can move a user to a different provider selecting the provider and clicking **Set New Provider**. You can define if the user will get a new depot and license based on the new provider default settings.

4.8.3 Create User

To add a new user, click **Create User** at the top of the **Manage Users** page, or select the item from the **Users** menu.

This brings up a form where you can enter the new user's details. Click **Create User** when you are done, or **Back** to cancel the operation and return to previous page.



In you control a number of providers, you can select the provider of th user from the drop-down menu at the top of the page. If you manage a number of accounts, a drop-down menu in the form allows you to select one of the accounts. When adding a user to an account you must also specify the membership type.

Note: Note that usernames need to be unique, not just locally, but across the TDNS if your Registration Server is connected to the TDNS. If you enter a username that is already registered on another Registration Server, the Administration Console will return an error.

You can either specify a password by deactivating **Request password on first login**, or have the user request a temporary password upon first log in (the default).

Note that the user will still not be able to login on a client device, or the Admin Console until the user has been activated. This can be done using the Admin Console, or it can be done by the user who may require an activation email when installing the TeamDrive client.

4.9 Clients

This section allows administrate all client and device related tasks, including: client Updates, banners and client log files.

4.9.1 Managing Devices

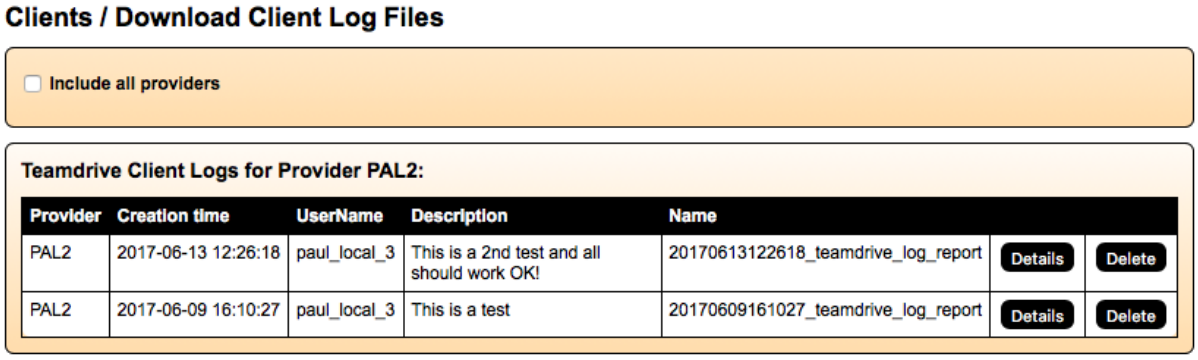
Select the **Manage Devices** item from the **Clients** to see a view of all client device installations. Different filters can be defined to limit the results, e.g. by client version, OS platform or last active date. If you have access to multiple providers you can click the checkbox **Include all providers** to list the devices of all your users.

You can wipe or delete multiple devices by checking the respective devices and clicking the **Wipe** or **Delete** button on top of the column.

The result set can also be exported as a CSV file by clicking **Export results to CSV file** on the bottom of the table.

4.9.2 Download Client Log Files

The Admin Console allows downloading client log files for troubleshooting purposes.



4.10 Licenses

The **Licenses** menu is only visible if the provider setting `LICENSE/ALLOW_MANAGE_LICENSE` is set to `True`. In addition, users may only access this manage licenses page if they have the `VIEW-LICENSE` privilege.

If no other license is specified, then a user receives a default license on creation. The features enabled for the default license is specified by the `CLIENT/DEFAULT_ACCOUNT_FEATURE` provider setting if the user is created as a member of an account, and by `LICENSE/DEFAULT_FREE_FEATURE` otherwise.

Instead of defining a default feature, it's also possible to define a default license using the `LICENSE/DEFAULT_LICENSEKEY` provider setting. This license is automatically assigned to users of the provider on creation.

4.10.1 Manage Licenses

To manage licenses, click **Licenses** in the top menu bar. A list of licenses is displayed on the **Manage Licenses** page.

Enter your search criteria in the **Filter Table** form at the top of the page in order to find specific licenses. Click **Apply Filter** to apply the selected criteria. Click **Clear Filter** to return to the full table view.

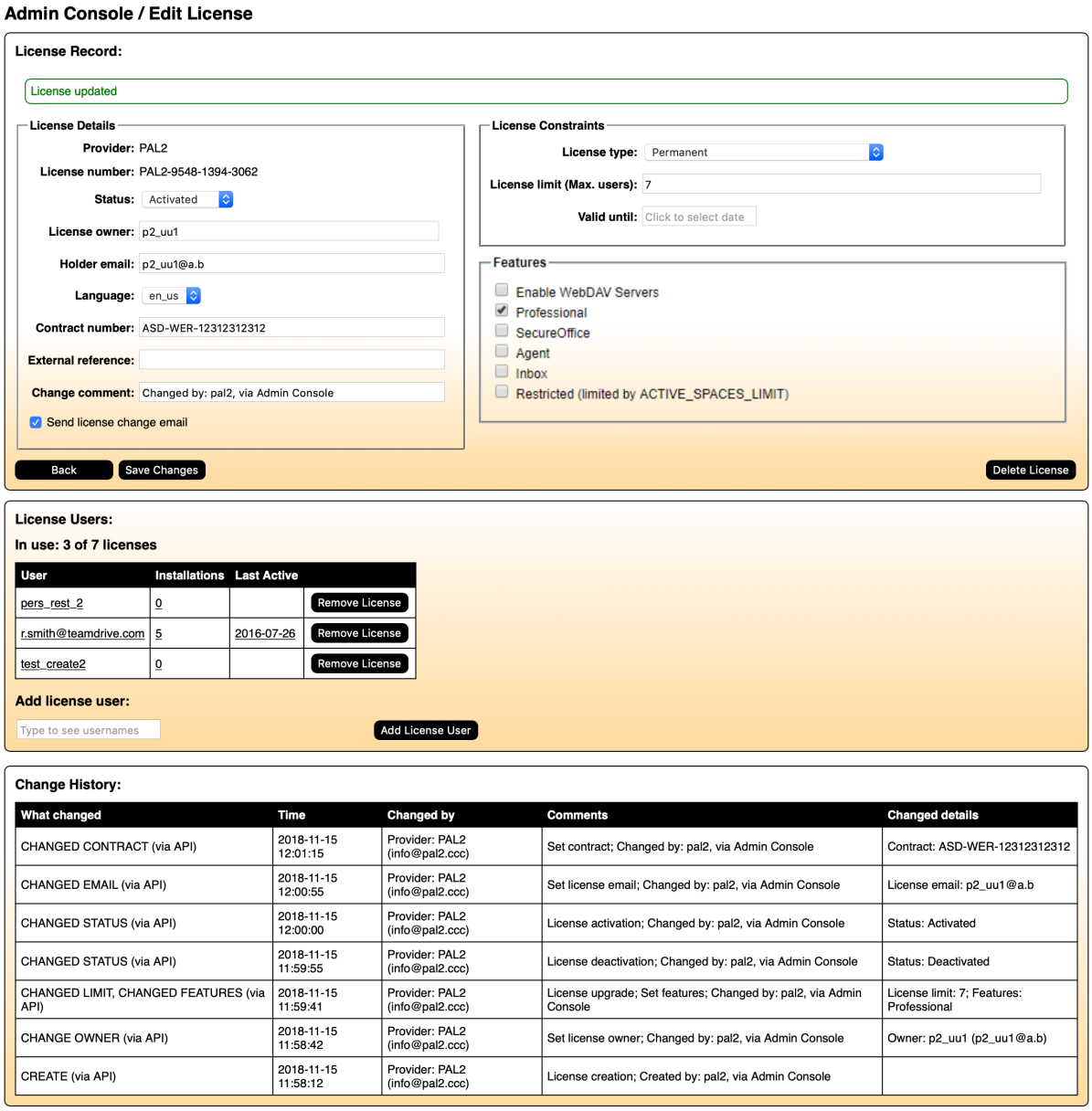
To customize the columns displayed, click **Configure columns** on the top right of the table. Select the desired columns and click **Update** to refresh the table view.

As with the user page, the search results may be displayed over several pages. To export the result set in a CSV file, click **Export results to CSV file** at the bottom of the table. This will bring up your browser's file saving dialogue.

To display additional details about a license, click **More Info** on the right side of the row. This will list all users of the license, as well as the change history of the license. Click **Less Info** to hide these details again.

4.10.2 Editing Licenses

To edit a license, find the license in the **Licenses** table and click **Edit**. This takes you to the **Edit License** page.



On this page, you can change various features of a license, e.g. the Client features, number of users, owner, user as well as an expiry date.

If you have the `EDIT-LICENSE-COST` privilege you will be able to change the license limit, valid until date, features and external reference of the license. Changing the external reference should be done with due care, since this may disrupt the operation of external systems that are connected to the Registration Server.

Once you have finished making changes, click **Save Changes** to apply them. Delete a license by clicking **Delete License**.

Each modification creates an entry in the license's **Change History**, which is displayed below the editing dialogue.

If a shop is associated with the provider (see provider_shop_settings for more details) of the license, and you have the `UPGRADE_LICENSE` privilege, then a button titled **Upgrade License** will be made available which will take you to a page in the shop where the license can be upgraded.

4.10.3 Creating License

For creating new licenses the provider setting `LICENSE/ALLOW_CREATE_LICENSE` must be set to `True`.

To create a new license, click the **Create License** button on the **Manage Licenses** page, or select this item from the **License** menu.

Customize terms and features of the license according to your requirements.

If you are not creating a license for an account, you specify an owner of the license by clicking on the **Select** button next to the **Owner** field.

Click **Create License** to create it. Clicking **Back** will return you the previous page.

4.11 Depots

4.11.1 Manage Depots

Select the **Depot** menu item to go to the **Manage Depots** page.

Depots are provided by a TeamDrive Host Server which must be registered with a particular provider on the Registration Server. Once registered, the Registration Server makes functions available for the creation and modification of depots.

Depots access information is distributed automatically the TeamDrive client software of users that have the depots in use.

If the `HOSTSERVER/HAS_DEFAULT_DEPOT` setting is set to `True`, or if this is specified on the account level, then new users receive a default depot created automatically by the Registration Server.

On the **Manage Depots** page you can use the filter table to search for depots under you control, using various criteria.

4.11.2 Create Depot

Click **Create Depot** to create new Space depots on a Host Server and assign it to selected users.

If there is more than one Host Server associated with your provider, you can choose the location of the depot by selecting the Host Server from a dropdown list all registered servers.

Type in a letter in the Depot owner field to get a list of available user names. A select list below the field will show all matching user names.

You can define a **Storage size** by entering the desired amount in the input field. Also enter a **Traffic limit** value which should be about 10 times the storage siye.

If required, you can modify these limits later on **Edit Depot** page.

It is possible to assign a depot to multiple users. Type in a letter the field next to **Add depot users:**. Users with match usernames and emails will appear in a list for you to select. Click on the '+'-sign to add more users of the depot.

Click **Create Depot** to finalize the depot creation.

The user's client devices will automatically be notified about the additional depot.

CHAPTER

FIVE

SETTING UP A PROVIDER

You must specify a Provider (formerly called “Distributor”) when setting up a Registration Server. The first Provider will be the Default Provider and has all rights to administrate all additional provider and their users and licenses. This first provider will be created during the server setup as described in (see provider setup)

After setting up the Registration Server, more Providers can be added as required. Adding a new Provider is explained in (see *Provider Record* (page 16)). After setup, changes can be made to the Provider settings using the Admin Console as described in the same chapter.

CHAPTER

SIX

IMPORTING USERS VIA CSV FILES

Instead of manually creating individual users via the Administration Console as described in chapter *Create User* (page 32), it is possible to import multiple users into the Registration Server's database from a file containing the user information as a CSV (comma-separated values) list.

The CSV import can be enabled and configured via the Provider Settings located in the `CSVIMPORT` group. See the *TeamDrive Registration Server Reference Guide* for more details on these settings.

There are two options on how to upload the CSV file:

- Upload the import file to a "hot folder", which can be configured using the `CSV_IMPORT_DIR` setting. The upload can be performed by an external system, e.g. via `rsync`, `scp` or `sftp`, or via a local cron job (e.g. using `wget` or any other tool to pull the file from a remote location).
- Upload the import file manually via the Registration Server Administration Console.

The data import via a hot folder is performed by an Auto Task which polls a directory for files containing CSV data at a defined interval (once every hour by default). See *"CSV Import" Task* (page 72) for details.

6.1 CSV File Structure

A CSV import can be used to create new users and update existing user records.

When a user is created, the setting `LOGIN/USER_IDENTIFICATION_METHOD` (see user_identification_method) specifies how a user will be identified. In other words, the name used during login. This can either by a username or an email address. Your import file must conform to this specification.

If a user already exists, then the import can recognise this fact and update a user record, instead of creating a new one. In this case the setting `CSVIMPORT/CSV_IDENTITY_COLUMN` (see csv_identity_column) specifies the import column that uniquely identifies the user.

Note that the value of this column may not change, in order for the update to work. Only set `CSV_IDENTITY_COLUMN` to a value other than `username`, if you know that this field is never updated.

The CSV file must contain the following fields, separated by comma or semicolon:

username This is the a globally unique name for a user. Usernames are unique over all TeamDrive Registration Servers. A username must be specified if `USER_IDENTIFICATION_METHOD` or `CSV_IDENTITY_COLUMN` is set to `username`. This field may be omitted if `USER_IDENTIFICATION_METHOD` is set to `default` or `email`. In this case you will create a user which is only identified by an email address. In order to do this, `CSV_IDENTITY_COLUMN` must be set to a value other than `username`. Once set, the username may not change. Registration Server version 3.6 will not allow users to be created with usernames that look like email addresses (that contain the "@" character). Such usernames are still allowed as reference to users created by previous versions of the Registration Server. By default, this is also the `CSV_IDENTITY_COLUMN` column.

email Registration email address of the user. This value is not optional. The Registration Server ensures that the email is unique per Provider. There may be additional uniqueness contraints imposed by the global settings `EmailGloballyUnique` and `UserEmailUnique`. If the

Provider setting `ISOLATED_EMAIL_SCOPE` is set to `True` then the global settings are ignored (see isolated_email_space).

password A password for the user. If empty, the user can define a password during the initial registration process as described in the *Reference Guide*. Changes to an existing user password will be ignored.

distributor The 4 letter Provider Code of the user's Provider (optional, see note below).

reference A free text field which can be used to assign an external reference ID (e.g. a cost center). If `CSV_IDENTITY_COLUMN` is set to this column, then `CLIENT/EXT_USER_REFERENCE_UNIQUE` should be set to `True` for the Provider.

department A free text field which can be used to set a department reference for the user. May be changed, if the provider setting `CSVIMPORT/CSV_ALLOW_SET_DEPARTMENT` is set to `True` (which is the default). If `CSVIMPORT/DISABLE_MISSING_CSV_USERS` is set to `True` then this field must be identical for all records in the import file (i.e. you must use one import file per Department).

language Language code of the user. This value may change.

authid This field is optional. It can contain a unique ID that can be used as an alternative reference. If `CSV_IDENTITY_COLUMN` is set to this column, then the value in this column will be used to identify the user on update In this case, the value in this column may not change. This is the same ID that is used by an external authentication service such as AD or LDAP. As a result, if the column is used you must be certain that it conforms to the usage of any existing or future external authentication service (this is the value `$ldap_user_id_attr` referred to in *Configuring External Authentication using Microsoft Active Directory / LDAP* (page 53)).

Example file structure (without an authid field):

```
username;email;password;distributor;reference;department;language
TeamDriveUser1;TD_User1@yourdomain.com;password1;;1234;Int1;EN
TeamDriveUser2;TD_User2@yourdomain.com;password2;;1342;Int2;DE
TeamDriveUser3;TD_User3@yourdomain.com;password3;;1452;Int2;DE
```

Note: Note that even though the CSV file contains a field to define a user's provider code, this value is currently not used. Instead, the provider code is defined by the user that uploads the CSV file via the Administration Console or by the directory the file is located in. If you need to upload users for multiple providers, create one file per provider and upload them separately.

6.2 Enable CSV Upload via the Administration Console

You can enable the CSV import functionality via the Administration Console by adding the provider setting `CSVIMPORT/CSV_IMPORT_ACTIVE` and setting it to `True` via the Administration Console.

Additionally, the Auto Task "CSV Import" must be enabled, by setting its status to `Enabled` via the Administration Console (**Admin -> Manage Auto Tasks**). Change the frequency to the desired time interval in which this Auto Task should be executed. For testing purposes, it might make sense to set it to a very short frequency (e.g. 1 minute).

Note: The import of a single user requires about 1 second. Make sure that the Auto Task's Frequency allows enough time for the currently running job to finish before another task is started.

If your list of users does not change frequently, it might make sense to keep the Auto Task disabled and only activate it temporarily, after a new CSV file has been uploaded.

In this mode, the CSV files and result logs are stored in the Registration Server's database and can be managed via the Administration Console.

To upload your CSV user data manually via the Administration Console, follow the instructions outlined in chapter *Importing Users via CSV Files* (page 39).

6.3 Uploading CSV Files to a Directory

As an alternative to the manual upload via the Administration Console, you can define a directory on the Registration Server that will be scanned for new CSV files periodically.

This so called "hot folder" allows for an automated process to create or disable users by uploading updated CSV files using tools like `scp`, `sftp` or `rsync` from another server. An example directory structure can be created in `/var/tmp` using the following command (replace `XXXX` with your provider code):

```
[root@regserver ~]# install -m 700 -d /var/tmp/csvimport/XXXX/error
[root@regserver ~]# install -m 700 -d /var/tmp/csvimport/XXXX/success
[root@regserver ~]# chown -R apache:apache /var/tmp/csvimport
[root@regserver ~]# tree /var/tmp/csvimport
csvimport/
`-- XXXX
    |-- error
    `-- success

3 directories, 0 files
```

In addition to activating CSV import via the `CSV_IMPORT_ACTIVE` setting as outlined above, you need to add and configure the following Provider Settings via the Administration Console:

CSVIMPORT/CSV_USE_FILESYSTEM: Set this option to `True` to use a directory on the Registration Server for uploading user information in a CSV file. You should only enable this setting after you created the required directory structure and updated the following settings accordingly. Changing this setting to `True` will automatically add the following settings to your Provider Settings.

CSVIMPORT/CSV_UPLOAD_DIR: This directory is the location for uploading new CSV files that should be processed by the import script (e.g. `/var/tmp/csvimport/XXXX/` in the example above). The name must end with a slash. Each provider must to use a different directory. It must be readable and writable for the Linux user that the CSV import job is running under (`apache` by default).

CSVIMPORT/CSV_SUCCESS_DIR: This directory contains the log files for successful CSV imports (e.g. `/var/tmp/csvimport/XXXX/success` in the example above). The name must end with a slash. It must be readable and writable for the Linux user that the CSV import job is running under (`apache` by default).

CSVIMPORT/CSV_ERROR_DIR: This directory contains the log files for failed CSV imports (e.g. `/var/tmp/csvimport/XXXX/error` in the example above). The name must end with a slash. It must be readable and writable for the Linux user that the CSV import job is running under (`apache` by default).

Now copy the CSV file containing your users into the directory defined in `CSV_UPLOAD_DIR` (e.g. `/var/tmp/csvupload/XXXX` in the example above).

Note: Please ensure that the file's ownership and permissions are set correctly, so that the Auto Task can delete the file after it has been processed.

After the Auto Task has been executed, the file will be imported into the database and processed. Afterwards, you can review the processing status via the Administration Console (**Manage Servers -> CSV user imports**).

6.4 Customizing the CSV Import

The CSV import can be further customized using the following Provider settings:

CSVIMPORT/CSV_ALLOW_SET_DEPARTMENT: Set this to `False`, if the department may not be changed by the CSV import of an existing user.

CSVIMPORT/CSV_IDENTITY_COLUMN: This setting specifies which field in the CSV file will be used to identify users during the CSV import (options are: `username`, `email`, `reference` and `authid`)

CSVIMPORT/DISABLE_MISSING_CSV_USERS: If set to `True`, any user not present in the CSV file will be disabled on the Registration Server. In this mode, your CSV user file always needs to contain all active users. In addition, an import file may only contain the users of one Department, if the Department field is used.

LOGIN/USER_IDENTIFICATION_METHOD: This setting determines how a user is identified by the user. In other words, what name is used on login to TeamDrive. See user_identification_method for more details.

CLIENT/EXT_USER_REFERENCE_UNIQUE: If you wish to use the `reference` columns to identify a user when updating users during import, then this value must be set to `True`. See ext_user_reference_unique for more details).

CHAPTER

SEVEN

BACKUPS AND MONITORING

7.1 System Backup Strategies

The most important asset of a live Registration Server is the content of its MySQL database.

The Registration Server's MySQL databases that need to be backed up are named `td2reg` and (optionally) `td2apilog`. They use MySQL's InnoDB storage engine to provide transaction support, fast recovery and consistency.

The backup schedule depends on the amount of users, their activity and your recovery point objective. We recommend to run a backup at least once a day. The backups should be safely stored on another system.

Ideally, the time and frequency of the Registration Server backup should be synchronized with the backup schedule used on the associated Host Server(s) — this ensures that the information about Users and their Space Depots is consistent across these servers.

In a virtualized environment, the usage of VM snapshots is highly recommended, as these provide atomic and instant full-system copies across multiple systems that can be backed up offline.

The MySQL backup can be performed using any established MySQL backup method, e.g. running a `mysqldump` via a cron job, or using more sophisticated tools like Percona XtraBackup or Oracle's MySQL Enterprise Backup. Other commercial backup solutions usually offer MySQL-specific plugins or extensions as well.

An example MySQL backup job using `mysqldump` could look like as follows. The SQL dump is piped through `gzip` for compression before it is written to a directory `/backup`, using a time stamp for the file name:

```
[root@regserver ~]# mysqldump -u root -p --single-transaction \
--databases td2reg td2apilog \
| gzip > /backup/td-regserver-mysql-$(date +%Y-%m-%d_%H.%M).sql.gz
```

See the MySQL documentation at https://dev.mysql.com/doc/refman/5.1/en/backup-and-recovery.html for more details and hints on how to define a MySQL backup strategy.

If the I/O overhead introduced by running the backup job on the production database is a concern, we recommend setting up a MySQL replication slave on another host and use this one to perform the backup. This second MySQL instance can also function as a hot standby server for high-availability purposes.

More details about MySQL replication and high availability can be found in the MySQL reference manual at https://dev.mysql.com/doc/refman/5.1/en/replication.html and https://dev.mysql.com/doc/refman/5.1/en/ha-overview.html.

In addition to the MySQL databases, we recommend to create backup copies of the Server's configuration files. Please refer to the *TeamDrive Registration Server Installation Guide* for details on the relevant configuration files.

These files should be backed up at least every time you changed them. These backups can be performed using any file-based backup method, e.g. using `tar`, `rsync` or more sophisticated backup tools, e.g. Amanda or Bacula.

7.2 System Monitoring

It's highly recommended to set up some kind of system monitoring, to receive notifications in case of any critical conditions or failures.

Since the TeamDrive Registration Server is based on standard Linux components like the Apache HTTP Server and the MySQL database, almost any system monitoring solution can be used to monitor the health of these services.

We recommend using Nagios or a derivative like Icinga or Centreon. Other well-established monitoring systems like Zabbix or Munin will also work. Most of these offer standard checks to monitor CPU usage, memory utilization, disk space and other critical server parameters.

In addition to these basic system parameters, the existence and operational status of the following services/processes should be monitored:

- The MySQL Server (system process `mysqld`) is up and running and answering to SQL queries
- The Apache HTTP Server (`httpd`) is up and running and answering to http requests. This can be verified by accessing the following URL: https://regserver.yourdomain.com/yvva/reg/ping.xml?tdns=\protect\T1\textdollartrue (remove the `?tdns=true` part, if your Registration Server is not connected to the TeamDrive Name Service TDNS)
- The `td-regserver` auto task is running (process name `yvvad`)
- The mail service (e.g. a local `postfix` instance) is up and running and mails are sent out correctly

CHAPTER

EIGHT

REGISTRATION SERVER FAILOVER AND SCALABILITY CONSIDERATIONS

8.1 Scaling a TeamDrive Registration Server Setup

A first step in increasing a single Registration Server's performance would be to monitor and review the system's CPU and RAM utilization, and to adjust the server configuration by adding more RAM or CPUs, if necessary (also called "scale-up strategy").

Adding more CPUs typically increases the maximum number of possible concurrent connections to the service and reduces the latency. However, the ability to handle more connections also requires more memory, as the system needs to spawn more concurrent Apache instances. So usually both parameters need to be adjusted.

Adding more RAM can also help to improve database throughput and latency, as it allows the database to keep more of its working set in memory, which enables it to return query results quicker.

If your setup has reached the physical limits of a single server instance, you can further improve the scalability as well as the redundancy of a TeamDrive Registration Server by implementing a "scale out" strategy.

In this setup, you distribute the load across several independent systems, by deploying multiple virtual or physical Apache server instances of the TeamDrive Registration Server behind one or more load balancers.

This configuration also mitigates the risk of a service outage, e.g. if an instance fails or needs to be taken offline for maintenance purposes.

A migration from a single instance setup to such a scaled-out configuration can usually be performed with very little downtime, so you can start small and grow your setup as the need arises.

However, you must ensure that in case of a node failover/outage, the remaining nodes can handle the load that is usually distributed across all server instances.

Note: In a scale-out scenario, the Registration Server's MySQL database server must be set up as a separate instance, so each Registration Server node has access to the same data set.

To avoid the MySQL database to become a single point of failure, we recommend to set up MySQL in a redundant configuration, too (e.g. by using MySQL replication or other clustering technologies like Galera/Percona Cluster).

Note: The TeamDrive Registration Server configuration does not support accessing more than one MySQL Server; you need to use a floating/virtual IP address that gets assigned to the currently active MySQL instance.

If you intend to run multiple independent Registration Server instances (e.g. to serve a globally distributed user base), you can assign users to local Registration Servers using different Provider Codes. Use TDNS to facilitate collaboration (e.g. exchanging Space invitations) between these independent TeamDrive Registration Server instances (which can in turn be scaled using the strategies above).

In a single instance configuration, a re-appearing server can suffer from a "thundering herd problem", as a large number of TeamDrive Clients will try to synchronize their accumulated pending changes simultaneously. This can

lead to a peak in the number of concurrent connections to this server and its MySQL database, as well a noticable increase in network and disk I/O.

This effect can be mitigated by temporarily extending the poll interval used by the Clients, by increasing the number of Apache instances, or by temporarily assigning more resources like vCPUs or vRAM to a virtual machine.

The MySQL server's configuration might also need to be reviewed in order to support more concurrent database connections.

8.2 Registration Server Failure Scenarios

This chapter discusses the most likely outages that can occur on a TeamDrive Registration Server, if no additional redundancy is provided.

Chapter *Registration Server Failover Test Plan* (page 48) outlines some possible tests you should perform, and what results to expect.

8.2.1 Entire Registration Server Outage

An outage of the entire TeamDrive Registration Server can be triggered by any of the following events:

- Failure of the entire Registration Server host system (e.g. a hardware or OS crash/failure)
- Network failure that renders the Registration Server unavailable
- Failure of the Registration Server's Apache HTTP Server
- Failure of the Registration Server's MySQL Database

Running Clients will indicate that the Registration Server can not be reached (for example, the TeamDrive 3 Desktop Client has an LED-like indicator icon in the bottom right corner, which will turn from green to red in case the Registration Server cannot be reached).

The following Client operations will continue to work:

- Running Clients can still operate on their existing Spaces (e.g. adding/removing files, uploading new versions)
- Clients can create new Spaces and delete existing Spaces
- Creating Space invitations to users stored in the Client's local addressbook

The following operations will be not be possible while the Registration Server is unavailable:

- Performing a login after having logged out of the TeamDrive Client
- Registration of a new device/Client
- Sending out Space Invitations to other users
- Changing the password or email address, requesting a temporary password
- Distributing comments on files via email
- Enabling/disabling the Key Repository

Once the Registration Server is reachable again, the Clients will proceed with sending out any pending invitations. The notification icon will change from red to green.

Except for the MySQL Server outage, this failure scenario can be avoided by setting up multiple instances of the Registration Server behind a load balancer with failover capabilities.

8.2.2 MySQL Database Outage

A failure of the Registration Server's MySQL Database could be triggered by one of the following events:

- Failure of the entire MySQL Server host system (e.g. a hardware or OS crash/failure)
- Network failure that renders the MySQL Server unavailable for the Registration Server
- Failure of the MySQL Server's `mysqld` process

The failure will be indicated by error messages in the following Registration Server log file.

`/var/log/td-regserver.log`:

```
[Error] -12036 (2002): Can't connect to local MySQL server through
socket '/var/lib/mysql/mysql.sock' (2)
```

A MySQL Database server failure will affect the entire Registration Server functionality as described in chapter *Entire Registration Server Outage* (page 46).

The service will return to normal operations as soon as the MySQL service is reachable again.

To mitigate the risk of a MySQL Server outage, consider setting up a cluster of MySQL Servers, using MySQL replication, DRBD or other replication and HA technologies like Pacemaker/Corosync to provide synchronization and redundancy.

8.2.3 SMTP Server Outage

If the local or remote SMTP server is unavailable for sending out email, the Registration Server will no longer be able to send out invitations, registration email notifications or file comment notification to the TeamDrive users. These messages will be kept in the Registration Server's internal mail queue until the SMTP service is available again.

Note: Note that sending out messages from a TeamDrive Client perspective still works — the Client receives a success notification as soon as the Registration Server has queued the message in its database for delivery.

Failures to connect to the SMTP server will be logged in file `/var/log/pbvm.log` as follows:

```
[ERROR] Connect to 'localhost:25' failed, getsockopt(SO_ERROR) returned
(111): Connection refused
```

The pending messages can also be viewed from the Registration Server Administration Console by clicking **Manage Emails -> View mail queue**.

Once the SMTP service is back online again, pending messages can be rescheduled for delivery by clicking **Reset Status** in the mail queue overview page.

Currently, there is no automatic method for rescheduling all pending messages in a bulk operation.

8.2.4 Outage of the `td-regserver` Background Service

The `td-regserver` background service is responsible for running a number of tasks, see the chapter *Auto Tasks* (page 71) in the *TeamDrive Registration Server Reference Guide*.

If the `td-regserver` background service (process name `yvvad`) has failed or was not started at bootup time, a number of operations will be affected, including the following:

- Emails won't be delivered anymore, including invitations, activation and email change messages.
- Licenses that have expired will not be processed.
- CSV imports will not be processed.

- Client change notifications that have been delayed will not be sent.
- Old messages will not be removed from the device-to-device message queues.
- Old entries won't be purged from the API log table (if enabled).
- Providers marked for deletion will not be deleted.

Restarting the `td-regserver` background service will pick up where the previous process has stopped.

For increased redundancy, it is possible to run this service on each TeamDrive Registration Server instance in a multi-server installation. In this setup, each instance needs to have a functional SMTP configuration, to ensure that email messages can be delivered.

8.3 Registration Server Failover Test Plan

Based on the failover scenarios described in chapter *Registration Server Failover and Scalability Considerations* (page 45), the following tests should be performed to verify the correct behaviour and recovery from failures of individual TeamDrive Registration Server components.

This test plan assumes an environment consisting of two virtualized TeamDrive Registration Server instances (regserv01 and regserv02), located behind a load balancer and using a dedicated single MySQL Server instance (td-mysql). Other setups/configurations may require additional tests, depending on the environment.

Note: Note that this configuration contains several components for which no redundancy is provided, therefore these components are considered single points of failure (SPOF). In particular, the following components can become a SPOF:

- The MySQL database instance (td-mysql). If this instance becomes unavailable, the entire TeamDrive service will be affected and rendered unavailable until the service is restored.
- The load balancer/firewall. If the public-facing load balancer/firewall fails, the TeamDrive service will be unavailable.

8.3.1 Single Registration Server Instance Failure

An outage of one of the TeamDrive Registration Server instances (regsrv01 or regserv02) should be simulated/triggered in the following ways:

- Shutting down the Apache HTTP Server running `service httpd stop`.
- Shutting down the network connection, e.g. by running `service network stop`, `ifconfig eth0 down` or by disconnecting the virtual network interface via the virtual machine management console.
- Shutting down the entire virtual machine e.g. via the virtual machine management console or by running `poweroff`.

Expected results:

- The load balancer should detect that the Registration Server instance is no longer available and redirect any incoming traffic to the remaining instance instead. If configured, a notification about the outage should be sent out to the monitoring software.
- The monitoring software should raise an alert about the Registration Server instance being unavailable, specifying the nature of the outage (e.g. httpd process missing, network unavailable, etc.).
- The remaining Registration Server instance should handle all incoming Client requests. The TeamDrive Service should not be impacted/affected in any way.

Once the outage has been resolved and the instance has recovered, the following is expected to happen:

- The load balancer should detect that the Registration Server instance is available again. Incoming traffic should be spread across both instances again.

- The monitoring software should detect the service recovery and perform the respective actions (e.g. resetting the alert, sending an update notification).
- The TeamDrive Service should continue unaffected throughout this process

8.3.2 Multiple Registration Server Failures

An outage of **both** of the TeamDrive Registration Server instances (regsrv01 and regserv02) should be simulated/triggered in the following ways:

- Shutting down the Apache HTTP Servers running `service httpd stop` on both instances.
- Shutting down the network connections, e.g. by running `service network stop`, `ifconfig eth0 down` on both instances, or by disconnecting the virtual network interfaces via the virtual machine management console.
- Shutting down the entire virtual machines e.g. via the virtual machine management console or by running `poweroff`.

Expected results:

- The load balancer should detect that the Registration Server instances are no longer available and stop redirecting any incoming traffic to the instances. Incoming requests should be answered with an appropriate error code (HTTP error code 503 - Service Unavailable). If configured, a notification about the outage should be sent out to the monitoring software.
- The monitoring software should raise an alert about the Registration Server instances being unavailable, specifying the nature of the outage (e.g. httpd process missing, network unavailable, etc.).
- The TeamDrive Service will be impacted/affected as outlined in chapter *Entire Registration Server Outage* (page 46).

Once the outage has been resolved and at least one of the Registration Server instances has been recovered, the following is expected to happen:

- The load balancer should detect that a Registration Server instance is available again. Incoming traffic should be redirected to this instance and incoming requests should no longer result in HTTP errors.
- The monitoring software should detect the service recovery and perform the respective actions (e.g. resetting the alert, sending an update notification).
- Once the TeamDrive Clients have noticed the service being available again, operations should proceed as before.

8.3.3 Testing MySQL Server Failures

An outage of one of the MySQL Server instance (td-mysql) should be simulated/triggered in the following ways:

- Shutting down the MySQL Server by running `service mysqld stop`.
- Shutting down the network connection, e.g. by running `service network stop`, `ifconfig eth0 down` or by disconnecting the virtual network interface via the virtual machine management console.
- Shutting down the entire virtual machine e.g. via the virtual machine management console or by running `poweroff`.

Expected results:

- The Registration Server instances will no longer be able to handle incoming Client requests as outlined in chapter *MySQL Database Outage* (page 47).
- The monitoring software should raise an alert about the MySQL Server instance being unavailable, specifying the nature of the outage (e.g. `mysqld` process missing, network unavailable, etc.).

Once the outage has been resolved and the MySQL Server is available again, the following is expected to happen:

- The TeamDrive Registration Server instances will continue to operate where they were interrupted by the MySQL Server outage. The TeamDrive Clients will pick up where they left, synchronizing all accumulated/pending changes.
- The monitoring software should detect the service recovery and perform the respective actions (e.g. resetting the alert, sending an update notification).

8.3.4 Testing Load Balancer Failure

Since all TeamDrive instances are accessed through a load-balancer, an outage of this component should be tested as well:

- Shutting down the load balancer
- Removing the network connections to the TeamDrive Server components

Expected results:

- The Registration Server instances will no longer be able to handle incoming Client requests as outlined in chapter *Entire Registration Server Outage* (page 46).
- The monitoring software should raise an alert about the load balancer instance being unavailable, specifying the nature of the outage.

Once the outage has been resolved and the load balancer is available again, the following is expected to happen:

- The TeamDrive Registration Server instances will continue to operate as soon as they receive incoming Client requests again. The TeamDrive Clients will pick up where they left, synchronizing all pending changes that have accumulated in the meanwhile.
- The monitoring software should detect the service recovery and perform the respective actions (e.g. resetting the alert, sending an update notification).

CHAPTER

NINE

CONNECTING USERS BETWEEN DIFFERENT REGISTRATION SERVERS

The TeamDrive Name Server (TDNS) settings are one of the more important settings which must be defined during the setup and which can not be enabled later on when users are already registered on your Registration Server.

The TDNS helps send invitations between users which are registered on different Registration Server by mappings the user to their respective servers. This is necessary because invitations must be send to the Registration Server for which the user is registered with their devices.

Usernames, unlike email addresses, are unique within the TDNS network. If you enable TDNS access, any username that is already in use by a server within the TDNS network can not be used by your own Registration Server.

TDNS access will modify the registration, login, search and invitation calls in the Registration Server (as well as the API calls) and check the TDNS, determining which username exists on which Registration Server in the TDNS network.

Every Provider requires a record on the TDNS. A record will have a *ServerID* and a *checksum*. All requests will contain the *ServerID* and *checksum* to verify that the request is coming from a valid Registration Server.

You have to enable outgoing access on the HTTP-Port 80 to `tdns.teamdrive.net` to enable the communication from your Registration Server to the global TDNS.

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

CHAPTER

TEN

CONFIGURING EXTERNAL AUTHENTICATION USING MICROSOFT ACTIVE DIRECTORY / LDAP

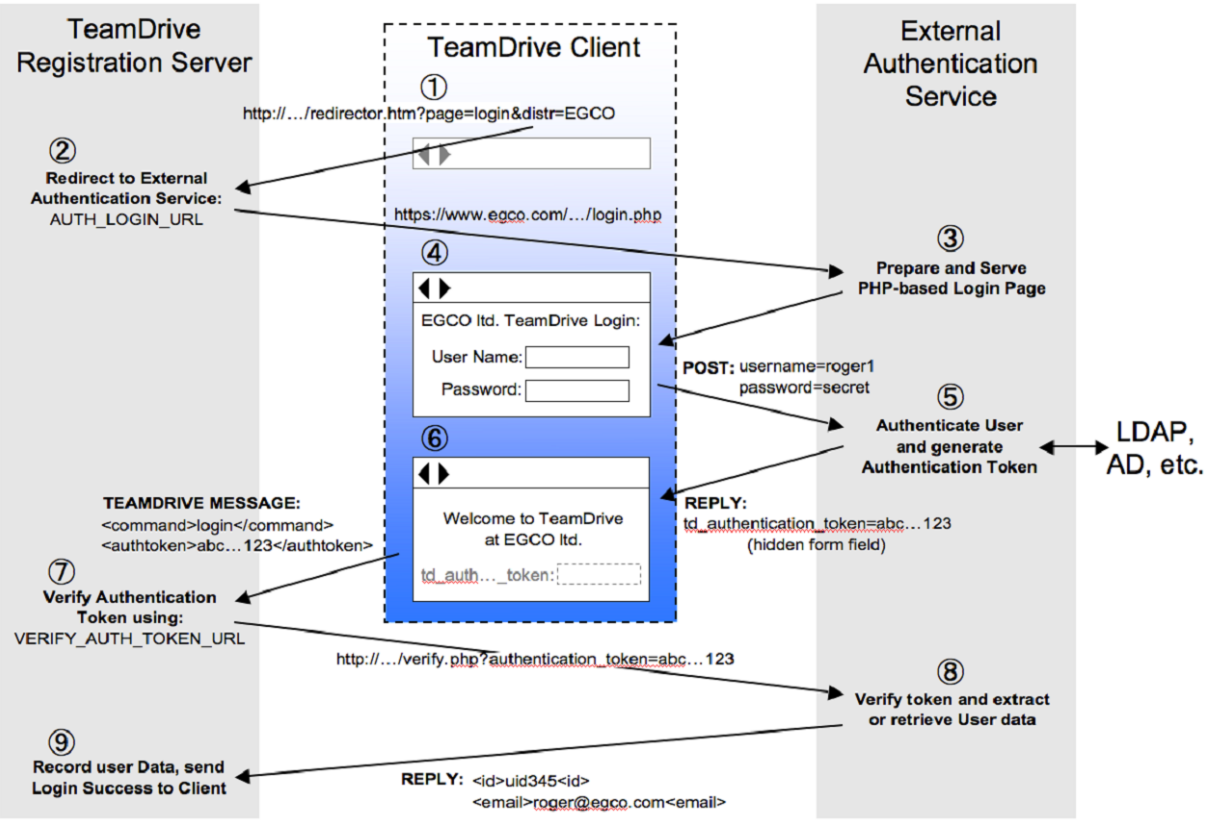
10.1 Overview

TeamDrive supports "External Authentication", where the authentication data is not stored on the Registration Server.

TeamDrive Client versions 3.1.1 and higher offer an alternative login window in an embedded browser, which resides in a different panel than the standard login dialogue. By default this window is disabled. It must be explicitly activated in the Client settings of the Registration Server. This process is described in detail below.

External Authentication is performed by an external web service, hosted on a web server separate from the TeamDrive Registration Server. This instance and the related web pages are referred to as an "Authentication Service".

Below is a general overview of the TeamDrive Client login process.



If a sign-in attempt was successful, the Authentication Service will return an "Authentication Token" which is received by the client and sent to the Registration Server. The Registration Server then uses a pre-defined URL to verify the token. If the token is valid, the login phase ends successfully and the TeamDrive Client is registered.

This service can be configured to work with various authentication mechanisms, such as NIS, LDAP, Active Directory, Shibboleth and others. Only the Authentication Service needs to contact your directory server in order to verify the user names and passwords provided. The Registration Server has no knowledge of these values. See the chapter see external authentication for more details.

The TeamDrive Registration Server installation ships with a PHP-based implementation of an Authentication Service for LDAP and Microsoft Active Directory Server.

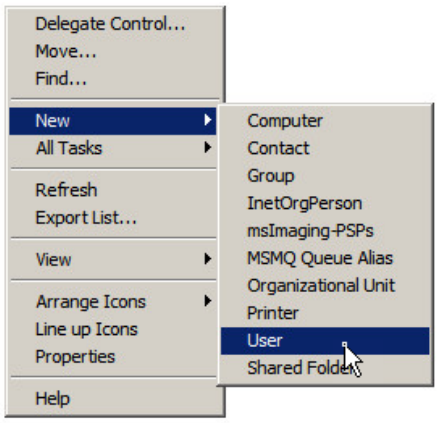
10.2 Active Directory

This section covers the Authentication of a TeamDrive Client using the Active Directory directory service offered by Microsoft Windows Servers. Since Windows Server 2008, this is also referred to as ADDS. ADDS manages various objects on a network such as users, groups, computers, services, servers, and shared folders. With the help of Active Directory, an administrator can organize, deploy, and monitor the information of these objects.

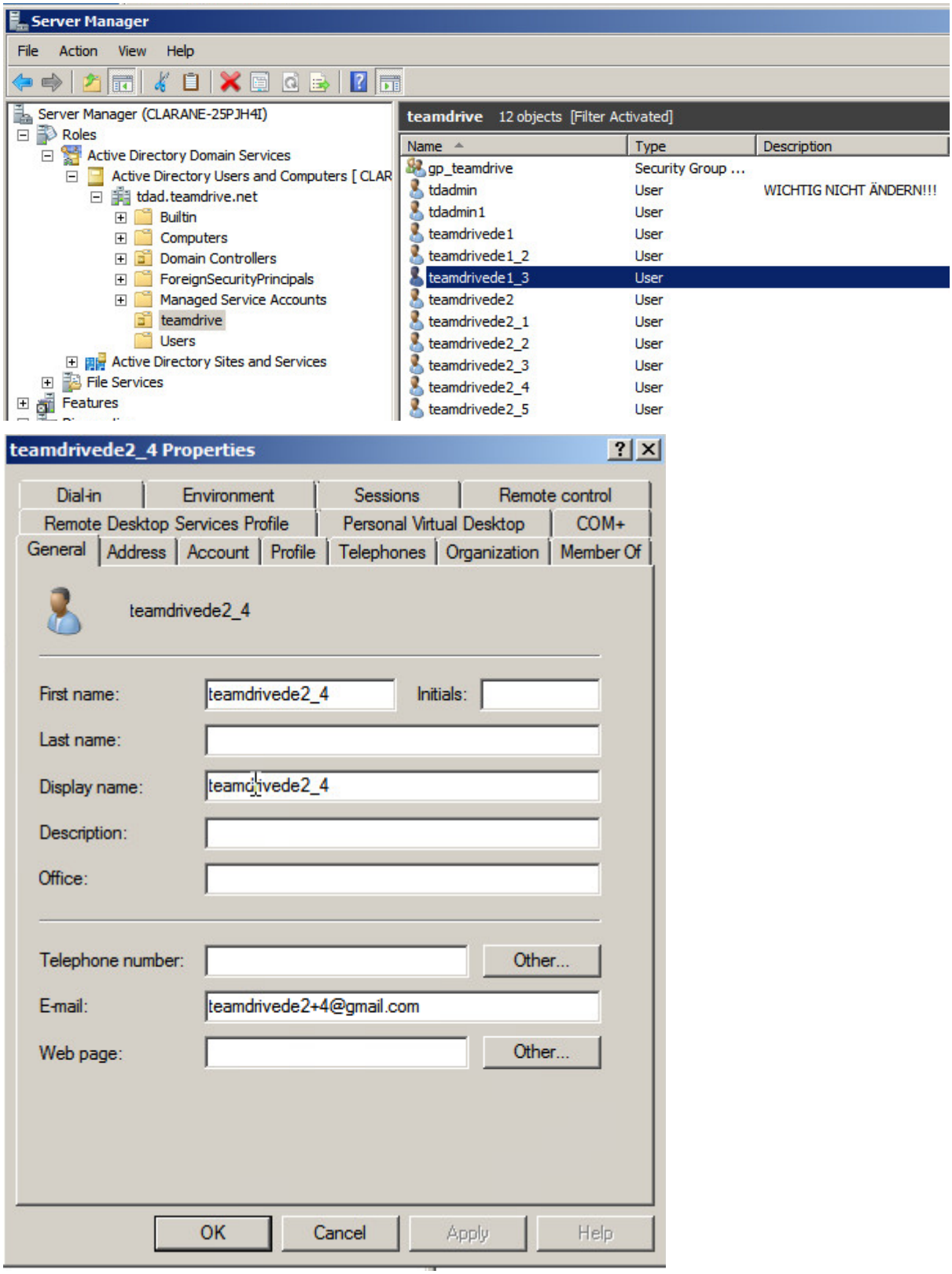
10.3 Configuring Microsoft Active Directory Server

10.3.1 Managing Users

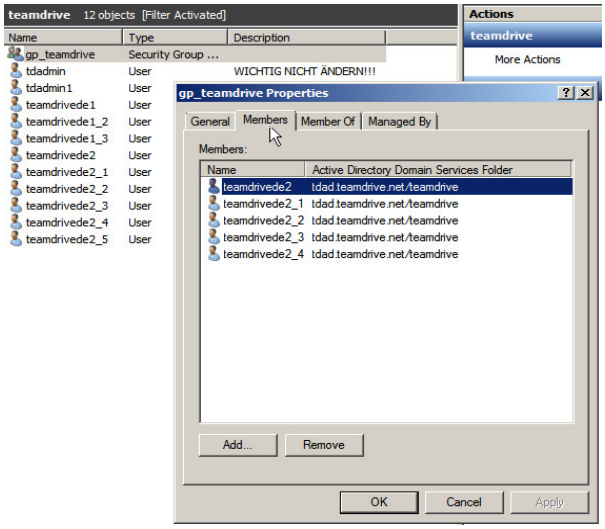
In the Server Manager, in the "Active Directory User and Computer" branch, create a new Organizational Unit with a meaningful name ("TeamDrive" for example). Select this newly created Organizational Unit and right click the middle panel to create a new user.



All users require an email address and need to be part of the group "gp_teamdrive".



In this picture it can be seen that the user is a member of the group “gp_teamdrive”.



10.4 Authentication Service Installation

Reference code to interface between an Active Directory or LDAP Server and the Registration Server is included in the TeamDrive Registration Server installation package (see Installing the Registration Server External Authentication) and is also installed on the TeamDrive Registration Server Virtual Appliance (in directory `/var/www/html/authservice`).

However, for security reasons, we strongly recommend to set up a dedicated system for the Authentication Service and to not use the Registration Server's web server for this.

The Authentication Service code can be deployed on any Linux system that supports running an HTTP Server (e.g. Apache) with the PHP scripting language and the PEAR extension enabled. It is strongly recommended to enable SSL for accessing the authentication web service, to protect the transmission of usernames and passwords from the TeamDrive Client to the Authentication Service.

The host providing the Authentication Service needs to be reachable by the TeamDrive Clients via HTTPS (TCP Port 443) as well as by your Registration Server via HTTP (TCP Port 80, if both systems are in a trusted environment) or HTTPS (TCP Port 443). Additionally, the Authentication Service needs to be able to access your directory service in order to verify usernames and passwords.

The detailed setup and configuration of this framework is out of the scope of this document; please refer to the installation instructions of your operating system and your local environment.

The following example assumes Red Hat Enterprise Linux 8 or a derivative like CentOS 8 Stream, Oracle Linux 8 or Scientific Linux 8. The names of packages or directories might differ on other Linux distributions.

On a minimal system, make sure that the following packages have been installed with `dnf`: `httpd`, `php`, `php-pear`, `php-ldap`, `php-mcrypt`, `openldap-clients`.

The "Log" PEAR module must be installed using `pear install`.

For installation details see chapter Installing the Registration Server External Authentication

For testing purposes, it's possible to simply open the login PHP page ("/authservice/ldap_login.php") in a regular web browser.

10.5 Authentication Service Customisation

You may change the user interface and behaviour of the LDAP Authentication Service by modifying the layout and content of the following files:

- `/authservice/ldap/index.html`: This is the default page, which redirects to `ldap_login.php` by default.
- `/authservice/ldap/ldap_login.php`: You can change this page to present a login page that would be recognised by your users. For example, change the page to conform to you companies CI (Corporate Identity).
- `/authservice/ldap/ldap_verify.php`: The only reason to change this file is to change the data returned to the Registration Server when an authentication token has been verified.

  For example, returning the email address is optional. If you do not return an email address, the Registration Server will return an error if a user with the specified "User ID" does not already exist.

  If an email address is returned, a user will be automatically created with the given User ID.

Note: All changes you make to other files under the `authservice` directory will be **overwritten** when you update to the latest td-regserver-ext-auth RPM.

10.6 Authentication Service Configuration

Once you have installed the external Authentication Service code, you must duplicate the file `ldap_config.php.example`, and rename it to `ldap_config.php`. The settings in this file must then be edited to access your LDAP or AD service.

Optional parameters may be set to "".

During testing of the LDAP connection, set the variable `$enable_debug` to `true`, in `ldap_config.php`. When set to `true` a trace of the LDAP/AD login attempt will be printed to the HTML page. In production this variable should be set to `false`.

10.6.1 Registration Server Parameters

The Registration Paramaters are required.

- `$reg_server_name`: Set this parameter to the name of your Registration Server. On successful login, the TeamDrive Client is passed this value in the `td_registration_server` hidden field.
- `$provider_code`: This is the Provider Code of the Provider associated with this external login service.

10.6.2 Agent/Portal Parameters

- `$allowed_origins`:

  This setting is an array of URLs, that are the permitted origins for calls to the external authentication service.

  When the TeamDrive Agent or Web Portal embeds the `ldap_login.php` page it will pass it's own origin (`<protocol>://<host>:<port>`) as a query parameter. After login, the browser is re-directed back to this page. The login page will check that the provided origin is in the following whitelist, if not an error will be reported. As a result, all legitimate origin URLs should be added to this list.

  Note that older version of the Web Portal did not provide this information. In this case, the first URL in the `$allowed_origins` list will be assumed to be the origin of login requests from the Web Portal, and after login the browser will be re-directed back to this URL.

- `$webportal_domain`:

  This setting is deprecated, and is no longer used. When upgrading to the latest version of the LDAP authentication service, copy the value of this variable to the position of the first URL in `$allowed_origins` (see above) array.

10.6.3 Encryption Parameters

Note: The parameters `$user_secret_salt` and `$token_encryption_key` are random sequences that **must** be changed for every new installation. Failure to do this results in a major security failure.

In Registration Server 4.5 or later you may leave these values empty during installation, and they will be set to random sequences automatically.

- `$prev_user_secret_ver`:

  This variable need only be set when upgrading from a previous version of the LDAP authentication service. In this case, you need to maintain compatibility with previously used versions of the user secret. In order to do this, set the variable to one of the following:

  - "v2": user secret generation v3 will be used, and clients that previously used v2 will be ugraded.
  - **"v1": user secret generation v3 will be used, and clients that previously** v1 will be ugraded.
  - **"v2,v1": user secret generation v2 will be used, and clients that previously** v1 will be ugraded.

  Please call TeamDrive for support for advice when upgrade you LDAP external service, if you are unsure which value to use.

  If this is a new installation, then leave this value empty, and user secret generation v3 will be used. This is the most secure version of user secret generation which uses the SHA256 HMAC algorithm.

- `$user_secret_salt`:

  This random sequence of characters **must be unique** for each installation. Once set, this value may **never be changed again**. It is also important that this value **remain secret** at all times as it is used to generate the, so-called, "user secret" value, which is used to encrypt the user's key repository stored on the Registration Server.

  On installation, leave this value blank, as found in the `ldap_config.php.example` file. The value will then be automatically set to a 54 character random sequence when first used.

  Changing the value will result in the user not being able to access their key repository, which means that the user will not have access to their spaces after a new TeamDrive installation.

  However, access can be restored for the new device if the user has an old device and performs a re-login (which can be initiated from the Admin Console for a user). Re-login forces the TeamDrive Client to re-encrypt the data in the key repository which will then make the Space keys available to new devices.

- `$token_encryption_key`:

  This random sequence of characters is used as a key to encrypt the authentication token sent to the client. The value **must be unique** for every installation.

  On installation, leave this value blank, as found in the `ldap_config.php.example` file. The value will then be automatically set to a 54 character random sequence when first used.

  This string may be changed at any time since authentication tokens are only valid of a short time.

10.6.4 LDAP/AD Parameters

For querying LDAP/AD, this implementation uses the LDAP functionality of the PEAR "Auth" module. Since this module is no longer externally maintained, an updated version (compatible with PHP 7.2 / 7.3) is included directly in the TeamDrive distribution. More information can be found at the URL http://pear.php.net/package/Auth/docs.

The configuration file contains parameters which the set the PEAR Authentication fields. The examples use values for an Active Directory query.

Connection Parameters

These parameters are required.

- `$ldap_server_domain`:

  This is domain name of the host on which the LDAP server is running.

- `$ldap_server_port`:

  This is the port on which the LDAP server is listening. The default port for LDAP is 389.

Note: Please check if the LDAP server can be reached from the Apache Web Server. The access might be blocked if SELinux is enabled. You can check this:

```
        [root@authserver ~]# getsebool -a | grep http | grep ldap

If you get ``httpd_can_connect_ldap --> off`` you have to allow the
communication or disable SELinux as described in :ref:`disable-selinux`.
```

- `$ldap_basedn`:

  This is the base "distinguished name" of the part of the Directory that will be searched. Examples: "dc=teamdrive,dc=com", "dc=egco,dc=teamdrive,dc=net"

Authentication

If the LDAP server does not allow anonymous connections then you must provide credentials for the connection here.

- `$ldap_binddn`:

  This is the name of the user that has access to the part of the directory that is to be searched. For example: "cn=Manager,dc=teamdrive,dc=com", "cn=TDAdmin,ou=global,ou=kkh,egco=tdad,dc=teamdrive,dc=net",

- `$ldap_bindpw`:

  The password of the user.

User Attributes

A number of user attributes can be sent to the TeamDrive Client after successful authentication. The user_id and email are required.

- `$ldap_user_id_attr`:

  This is the name of the attribute which contains a unique identifier of the user. Usually this value is "uid".

  In order for external authentication to work the LDAP/AD server must store a unique, unchanging, identifier for each user. Note that if the identifier changes TeamDrive will fail to recognise The user, and assume the user is new. For this reason the email address is not a choice.

- `$ldap_email_attr`:

  The name of the attribute that contains the user's email address. Note that this email address is used within TeamDrive in order to invite users to a Space.

- `$ldap_common_name_attr`:

  This is the name of the attribute that contains the user's common name. If provided, TeamDrive will display this name, instead of the email address in the user interface.

- `$ldap_telephone_attr`:

  This attribute contains the user's home or work telephone number.

- `$ldap_mobile_attr`:

  This attribute contains the user's mobile telephone number.

User Identification

These parameters are required. They are used to locate a user in the directory.

- `$ldap_userdn`:

  This specifies the user distinguished name to be searched. `$ldap_userdn` is added to `$ldap_basedn` when performing the search.

- `$ldap_userattr`:

  This parameter specifies the attribute that will be searched for the user's "login name". Any parameter that uniquely identifies the user may be used.

- `$ldap_userfilter`:

  This is added to the search filter when searching. It is usually used to specify the object type of the users, for example: "(objectClass=inetOrgPerson)" or "(objectClass=posixAccount)"

Group Specification

By specifying a group you can ensure that only users of a particular group are authorised to access TeamDrive.

Use the parameters below to to determine how to check whether a user is a member of a group.

- `$ldap_groupdn`:

  If this variable is empty, then no group check will be performed. The value of `$ldap_groupdn` is added to `$ldap_basedn` when searching for a group.

- `$ldap_groupscope`:

  The scope for group search, either `one`, `sub`, or `base`. `sub` is the default.

- `$ldap_groupfilter`:

  This is added to the search filter when searching for a group. It usually identifies the group object type, for example: "(objectClass=groupOfUniqueNames)".

- `$ldap_group`:

  This is the name of the group to be searched. User's that wish to Login to TeamDrive must be members of this group.

- `$ldap_groupattr`:

  This variable specifies the group attribute to searched forto find the group name: `$ldap_group`.

- `$ldap_memberattr`:

  This is the attribute in the group object that specifies the names of the members.

- `$ldap_memberisdn`:

  Set this variable to `true` if the `$ldap_memberattr` is the complete distinguished name (dn) of the user. If not, it is assumed to be just the value of the the memberattr is the dn of the user (default) or the value `$ldap_userattr` attribute.

10.7 Authentication Procedure

The `ldap_login.php` page generates an HTML form with standard fields to collect the user's credentials and generate the required query with it. If the authentication was successful, the PHP code of the login page generates the authentication token based on information returned from the directory server (Active Directory) and returns it to the client.

The HTML form also includes some hidden fields, which are evaluated by the TeamDrive Client. In these fields the Registration Server's name and the Provider Code are included.

The values are taken from the `$reg_server_name` and `$provider_code` settings.

```
<div id="loginFormWrapper">
        <form id="loginForm" action="ldap_login.php" method="post" enctype=
↪"multipart/form-data">
                <input type="hidden" id="td_login_page" value="login" />
                <input type="hidden" id="td_registration_server" value=
↪"TeamDriveMaster" />
                <input type="hidden" id="td_distributor_code" value="EGCO" />
```

Note: The communication between the Authentication Service and the directory service (e.g. LDAP or Active Directory) is performed without encryption by default. If these services communicate via an untrusted network, we strongly advise to enable some form of encryption, to protect against the potential eavesdropping of usernames and passwords. For example, LDAP supports encryption via SSL (LDAPS), other alternatives would be using a VPN or an SSH tunnel.

The content of the authentication token that is returned to the client is encrypted with a secret key. This key is stored in the `$token_encryption_key` parameter.

For debugging the generated query for the Active Directory, it is helpful to have the debugging information display in the browser, by setting `$enable_debug` to `true`.

Output is written to the login page, for example:

Logging Output:

```
DEBUG:AUTH: Auth::start() called.
DEBUG:AUTH: Auth::assignData() called.
DEBUG:AUTH: Auth::checkAuth() called.
DEBUG:AUTH: No login session.
DEBUG:AUTH: Auth::login() called.
DEBUG:AUTH: Loaded storage container (LDAP)
DEBUG:AUTH: Auth_Container_LDAP::fetchData() called.
DEBUG:AUTH: Auth_Container_LDAP::_connect() called.
DEBUG:AUTH: Connecting with host:port
DEBUG:AUTH: Successfully connected to server
DEBUG:AUTH: Switching to LDAP version 3
DEBUG:AUTH: Switching LDAP referrals to false
DEBUG:AUTH: Binding with credentials
DEBUG:AUTH: Binding was successful
DEBUG:AUTH: Auth_Container_LDAP::_getBaseDN() called.
DEBUG:AUTH: UTF8 encoding username for LDAPv3
DEBUG:AUTH: Searching with ldap_search and filter (&(mail=Teamdrivede2+1@gmail.com)(objectClass=user)) in dc=tdad,dc=teamdrive,dc=net
DEBUG:AUTH: User(s) found
DEBUG:AUTH: Saving attributes to Auth data in AUTH format
DEBUG:AUTH: Storing additional field: cn
DEBUG:AUTH: Storing additional field: uid
DEBUG:AUTH: Storing additional field: mail
DEBUG:AUTH: Bind as CN=teamdrivede2_1,OU=teamdrive,DC=tdad,DC=teamdrive,DC=net
DEBUG:AUTH: Bind successful
DEBUG:AUTH: Checking group membership
DEBUG:AUTH: Auth_Container_LDAP::checkGroup() called.
DEBUG:AUTH: Searching with ldap_list and filter (&(samAccountName=gp_teamdrive)(member=CN=teamdrivede2_1,OU=teamdrive,DC=tdad,DC=teamdrive
DEBUG:AUTH: User is member of group
DEBUG:AUTH: Auth_Container_LDAP::_disconnect() called.
DEBUG:AUTH: disconnecting from server
INFO:AUTH: Successful login.
DEBUG:AUTH: Auth::setAuth() called.
DEBUG:AUTH: Auth::checkAuth() called.
INFO:AUTH: Session OK.
DEBUG:AUTH: Auth::checkAuth() called.
INFO:AUTH: Session OK.
```

After the Authentication Service has confirmed the credentials of a user, an authentication token is passed to the TeamDrive client. The client then sends the token on to the registration server to complete the registration. Before the login process can be successfully completed, the registration server then verifies the authentication token by sending it to the Authentication Service.

This is done via the URL specified in the `VERIFY_AUTH_TOKEN_URL` setting (see verify_auth_token_url). The page referenced by the URL is referred to as the "verification page."

Note: If you use SSL to encrypt the token verification communication between the Registration Server and the Authentication Service (by providing an URL starting with `https://` in the `VERIFY_AUTH_TOKEN_URL`), you must install properly signed SSL certificates on the Auth Service's web server — using self-signed certificates will result in an authentication failure, displaying the error message `REG SERVER EXCEPTION "-24918" ( "0" ) "Verify authentication failed: result file not found"` in the Client log file. You can use the command line tool `curl` on the Registration Server to test opening the verification page. It should not complain about `SSL certificate problem: self signed certificate` or other SSL-related problems when opening the URL. Check 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 for the apache web server will create an A-rating and make sure that the handshake is working for current platforms and browser:

```
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
```

To complete the registration process, the registration server requires the user's ID and e-mail address. If the validation is successful, this information is sent back from the site as confirmation.

10.8 Web Portal Configuration

How to configure the TeamDrive Web Portal to use an external Authentication Service is described in the TeamDrive Web Portal Administration Guide.

Note: When an Authentication Service is used by the Web Portal, the authentication token will be verified twice: once by the Web Portal and once by the TeamDrive Agent (running in the Docker container).

10.9 TeamDrive Client Configuration

Enabling external authentication requires various settings to be adjusted using the Registration Server's Admin Console. For more information, see external authentication and/or settingsChapter in the *Reference Guide*.

Log in as the user that has the privileges to modify your provider settings.

Under "Providers/Provider Settings" the following parameters need to be set. Add the setting `AUTHSERVICE/USE_AUTH_SERVICE` and set `USE_AUTH_SERVICE` to **True**.

If external authentication only applies to user with specific email domains, then you should setup the "Domains & Services" accordingly in the Admin Console. In this case, you should leave the `AUTH_LOGIN_URL` and `VERIFY_AUTH_TOKEN_URL` described below empty. And the `PRE_LOGIN_SETTINGS` will be set automatically be the Registration Server.

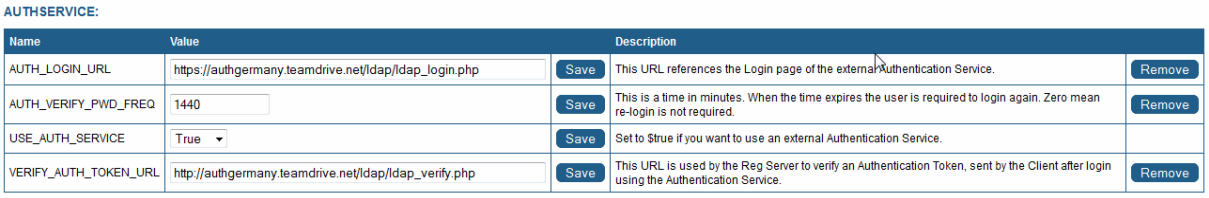
If the external authentication service applies to all users of a provider, then the `AUTH_LOGIN_URL` must be set to the URL of the webpage that handles Authentication. This page is the so called "Web-Login-Panel" and will be displayed to the user in the TeamDrive Client.

Set `AUTH_LOGIN_URL` to the Authentication Service's login URL, e.g. `http://authserver.yourdomain.com/authservice/ldap/ldap_login.php`.

Set `VERIFY_AUTH_TOKEN_URL` to the Authentication Service's token verification URL, e.g. `http://authserver.yourdomain.com/authservice/ldap/ldap_verify.php`.

Now the TeamDrive Client needs to be informed to use external Authentication Service for this Provider. In the Provider Settings, set `LOGIN/PRE_LOGIN_SETTINGS` as follows:

```
enable-login=false
enable-lost-password=false
enable-registration=false
enable-web-login=true
```



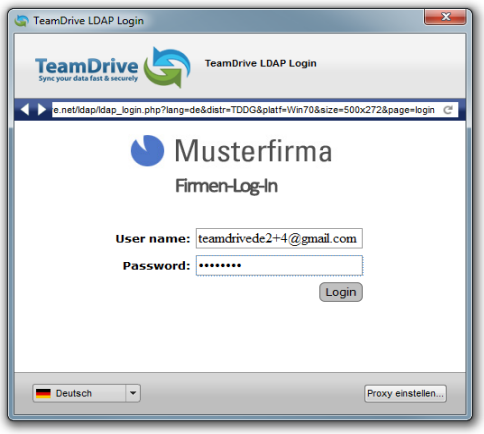
The web-login-panel will be displayed if the "`enable-web-login`" setting is set to "`true`" or "`default`" (see provider concept/login and registration settings/enable-web-login in the *Reference Guide*).

If the standard-login panel is also activated (see provider concept/login and registration settings/enable-login in the *Reference Guide*), `enable-web-login` should be set to `default`. This ensures that when the client is started, the Web-Login-Panel is shown to the user (as opposed to the Standard-login-panel).

The TeamDrive Client now calls the alternative login page within an embedded browser.

When logging in, AD users must enter their e-mail address (as opposed to their username) into the username field.

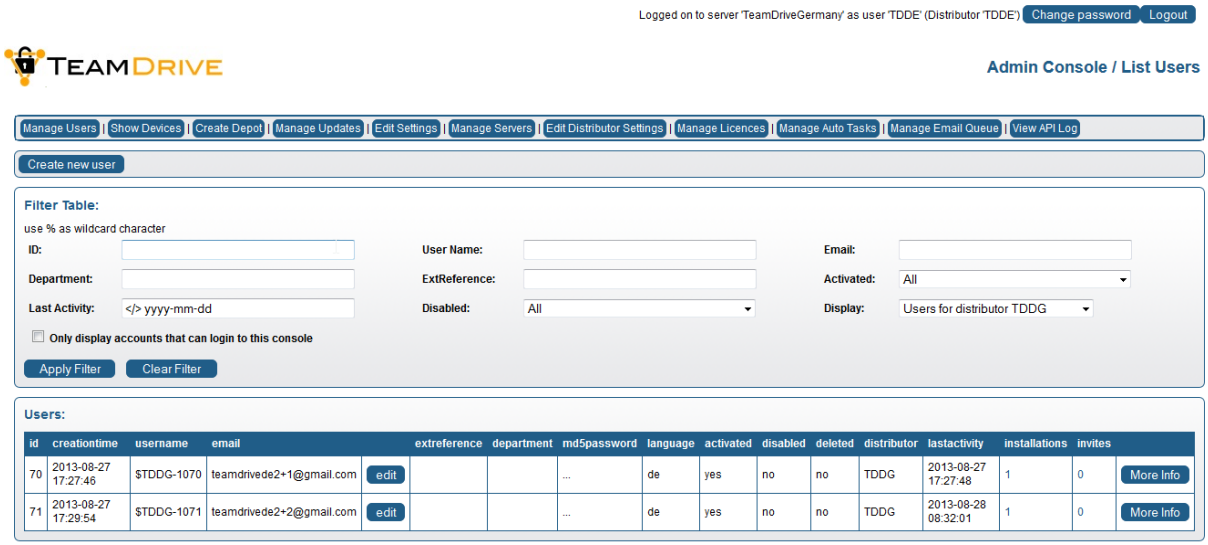
You can use the email address to reference users when making API calls. But, it is also possible to use the `authid`, which is set to the value specified by the `$ldap_user_id_attr` setting.



CLIENT

Name	Value		Description	
ALLOWED_DIST_CODES	*	Save	Permitted client Distributor Codes (besides TicketPrefix). '*' means accept all, '*.' means all on this Reg Server (multiple entries separated by ',').	
CLIENT_NETWORKS		Save	Networks (CIDR notation) or IP addresses that correspond to this Distributor (multiple settings must each be placed on a new line).	
CLIENT_SETTINGS	enable-login=false enable-web-login=true enable-registration=false enable-web-registration=false enable-lost-password=false	Save	Client settings which are applied after login (multiple settings must each be placed on a new line).	
DEFAULT_FREE_FEATURE	3	Save	Default feature which will be used to create a default license.	
FREE_LIMIT_SIZE	2147483648	Save	The free limit of a client. Should be identical to HOST_DEPOT_SIZE, but it's not mandatory.	Remove
PRE_LOGIN_SETTINGS	enable-login=false enable-web-login=true enable-registration=false enable-web-registration=false enable-lost-password=false	Save	Client settings which are applied before login (multiple settings must each be placed on a new line).	
USE_EMAIL_AS_REFERENCE	True	Save	In case that the email address will be used to identify the account, an username will be generated automatically in the API.	

Upon successful first authentication, the user will be automatically created on the Registration Server. The user can then be managed through the Registration Server's Management Console under the "Manage Users" tab.



For more information about managing users, see *Manage Users* (page 25).

CHAPTER

ELEVEN

CONFIGURING AND TESTING THE MYSQL DATABASE CONNECTIONS

11.1 Configuring the Registration Server's MySQL configuration

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

To change the MySQL login credentials for the Registration Server's database connections, open the file `/etc/td-regserver.my.cnf` in a text editor.

The `user` field identifies the user name, while the `password` field contains the MySQL user's password in plain text:

```
#
# This configuration file defines the MySQL login credentials (e.g. username,
# password, host name) used by the TeamDrive Registration Server Apache module
# (mod_yvva), the TeamDrive Registration Server Auto Tasks (service
# td-regserver) and (optionally) the PHP-based TeamDrive Registration Server
# Admin Console. You need to restart httpd and the TeamDrive Registration
# Server background process after making changes to this file.
#

[regdb]
database=td2reg
user=teamdrive
password=teamdrive
host=localhost
socket=/var/lib/mysql/mysql.sock
```

Note: Please note that this file contains the MySQL login credentials in plain text. Make sure to restrict the access permissions to this file so that only the root user and the Apache HTTP Server (`mod_yvva` in particular) can open this file. The file ownerships should be set to `apache:apache`, the file permissions should be set to "600".

After making changes to the credentials, you have to restart the Apache HTTP Server and the `td-regserver` background service.

If you're seeing any errors at this stage, please consult the chapter troubleshooting for guidance. Double check that the MySQL login credentials are correct. Also try to connect to the MySQL database using these values from the `mysql` command line client.

11.2 Admin Console MySQL Configuration

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

By default, the Administration Console uses the same configuration file as the Registration Server (`/etc/td-regserver.my.cnf`), so any changes made in this file also apply to the Administration Console, if it's located on the same host as the actual Registration Server.

The location of the MySQL configuration file is specified in the configuration file `/var/www/html/tdlibs/globals.php`. The distribution ships with an example configuration file `/var/www/html/tdlibs/globals-sample.php` — just copy it to `globals.php` and modify it to match your environment:

```
<?php
  /*
   * This file specifies how the TeamDrive Registration Server
   * Adminstration Console connects to the MySQL database.
   *
   * Please change these settings to suit your environment, and then
   * save this file as "globals.php"
   */

  /*
   * Specify a path to a local MySQL configuration file (default).
   * If found, these values override any settings provided in $dsn2import
   * below.
   *
   * The file should look as follows (MySQL INI-style format):
   *
   * [regdb]
   * database=td2reg
   * user=teamdrive
   * password=teamdrive
   * host=localhost
   */
   $mysqlConfigFile = '/etc/td-regserver.my.cnf';

  /*
   * Alternatively, enter the connection string to connect the MySQL database.
   * Use this option if the Admin Console is installed on a separate host and
   * there's no TeamDrive specific MySQL configuration file
   *
   * The format is: mysqli://<username>:<password>@<host>/<database>
   */
   //$dsn2import = 'mysqli://teamdrive:teamdrive@127.0.0.1/td2reg';
?>
```

As an alternative to providing the location of a MySQL configuration file (e.g. when installing the Administration Console on a different host), you can define the username, password and hostname required to connect to the MySQL database server in `globals.php` directly, by commenting out the `$mysqlConfigFile` variable and updating the connection string in the variable `$dsn2import` accordingly:

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

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

Note that the `mysqli:` protocol is being used since `mysql:` has been removed in PHP 7.x.

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

CHAPTER

TWELVE

REGISTRATION SERVER HOW TO'S

This chapter covers a number of common tasks that you may want to or need to perform with the Regisration Server.

12.1 Managing Client Updates

To inform your users about the availability of a new version of the TeamDrive client, you use the UPDATE/CURRENT_CLIENT_VERSION setting. This value determines whether a TeamDrive client receives an update notification. If the version of the client is less than CURRENT_CLIENT_VERSION then the user will be notified that a new version of the TeamDrive client is available.

Note that this notification only applies to desktop clients (mobile clients will be informed by the Google Play Store or by the Apple App Store).

As of Registration Server version 4.1, the Admin Console no longer supports update notifications for TeamDrive 3.

The ENABLE_UPDATE_TEST allows you to test how the client responds to an update notification. Setting the value to True will send an update notification to the user specified by the UPDATE_TEST_USER setting.

You can specify the client version to be sent to the update test user by setting the UPDATE_TEST_VERSION setting (available in version 4.1.3). If this setting is empty, then CURRENT_CLIENT_VERSION will be returned.

The update notification will direct the user to internet page where the latest version can be downloaded. The URL of the download page is defined in the provider setting REDIRECT/REDIRECT_DOWNLOAD. Set this to point to the download location where your users can obtain a new version of the TeamDrive client, e.g. http://www.yourdomain.com/download.html.

12.2 Configuring a Default License

A default license is generated for each user on registration. The features of this license are determined by the either the LICENSE/DEFAULT_FREE_FEATURE (see default_free_feature) or the LICENSE/DEFAULT_ACCOUNT_FEATURE Provider settings.

If a user is registered as a member of an account then the DEFAULT_ACCOUNT_FEATURE setting is used, otherwise the DEFAULT_FREE_FEATURE setting is used. This allows you to specify users that are created for a specific account receive different default license features to those that register themselves.

Alternatively, it is possible to create a single license which is to be used as a default for multiple users. To do this, first create the license using the Admin Console (see *Creating License* (page 35)).

Then set the Provider setting LICENSE/DEFAULT_LICENSEKEY to the key of the newly created license. Note that you will must ensure that the "license limit" (number of users) is sufficiently high to cover the number of users that will register and use the license.

The DEFAULT_LICENSEKEY applies to all newly registered users, including those assign to an account.

12.3 Changing the Default Depot Size

A default Depot for storage of Space data, may be created for a user on registration. For this purpose, a Hosting Service must be connected to the Registration Server. If this is the case, then you will be able to set the `HOSTSERVER/HOST_SERVER_NAME` Provider setting by selecting the Hosting Service from a popup menu.

The default size of the Depot is specified using the `HOST_DEPOT_SIZE` setting. By default, this value is 2 GB.

If you change this value then, for TeamDrive 3 users, you should also change the `CLIENT/FREE_LIMIT_SIZE` setting to the same value.

TeamDrive 3 clients limit the amount of data that will be processed by the Client when not using a Personal or Professional license. This means that if you do not increase `FREE_LIMIT_SIZE` in accordance with the *HOST_DEPOT_SIZE* value, users will not be able to use all the disk space available in the default Depot.

12.4 Setting up a Master User

A master user is a user that is automatically invited to all spaces of users of a provider. This has a number advantages, for example:

- All spaces keys used by users can be collected as a backup, in case the keys are lost.
- It creates a central repository where an Administrator can enter any Space used by any of the users.

A disadvantage is that anyone with access to the master user has access to all spaces.

You create a master user by setting the `master-user` client setting to the username of the master user. The value must be set in the `CLIENT/CLIENT_SETTINGS` Provider setting (see client_settings). This user will now be automatically invited to all Spaces with the "Master User" rights.

Note: In case of using the email as username (see user_identification_method) you have to use the magic username as master username.

It is now possible to install a TeamDrive client, login as the master user and setup the client to automatically accept invitations sent to it. This can be done by setting the client setting `auto-accept-invitation` to `true`.

Do not set this setting in the `CLIENT_SETTINGS` Provider setting as this would mean that users, in general, will loose control of how they wish to handle Space invitations. Instead, it is possible to set this setting in a local configuration file, so that it only applies to the master user installation.

This is the "/Users/Shared/teamdrive.ini" file on Mac OS X, "/etc/teamdrive.ini" on Linux and "%ProgramData%/TeamDrive3/teamdrive.ini" (usually "C:\ProgramData\TeamDrive3\teamdrive.ini") on Windows.

When run on a machine that is "always on" (i.e. a server) this will ensure that all invitations are received when sent to the master user from other clients.

The behaviour, whether files are downloaded directly after accepting the invitation, or just the "meta-data" of the Space, is determined by the `auto-accept-invitation-mode` client setting. This can be set to one of the following values: `non-offline-available`, `offline-available` or `archived`. The default is `archived`, which means the Space key is stored, and the Space will be marked as "Inactive". The Space can then be activated manually at a later stage.

12.5 Using a "Restricted" Client License Model

The Restrict License Model is intended to provide users with a limited but free version of TeamDrive. For this reason a restricted license is can to be the default license which a user receives on first time registration.

Note: The Restricted Client License Model is only supported by TeamDrive 4 Clients.

A restricted license tells the TeamDrive Client that certain restrictions apply. Currently this may only be a restriction to the number of Spaces that may be active at any one time.

To setup a Restricted Client License Model, do the following:

Set the Provider settings `DEFAULT_FREE_FEATURE` and `DEFAULT_ACCOUNT_FEATURE` to the **Restricted** and **WebDAV** features, depending on whether you want a non-commercial or a commercial license.

If you include the **Personal** feature the license will be usable by commercial/business users. Alternatively you could include the **Professional** feature which is considered identical to the **Personal** feature by TeamDrive 4 clients (see default_free_feature for details).

If you only want non-commercia/private users to be able to use the license then include the **WebDAV** feature instead of the **Personal** or **Professional** feature. This will ensure that the user can still use WebDAV hosting services, which is automatically included in the **Personal** or **Professional** features.

To ensure that the *DEFAULT_FREE_FEATURE*' and `DEFAULT_ACCOUNT_FEATURE` settings take effect you mus set `DEFAULT_LICENSEKEY` is blank.

Finally, ensure that the `LICENSE/ACTIVE_SPACES_LIMIT` provider setting is set to a value greater than 0 (by default the value is 1). This setting automatically adds the `active-spaces-limit` to the `CLIENT/CLIENT_SETTINGS` value sent to the client. The value determines the number of active Spaces allowed by the TeamDrive Client when the **Restricted** license feature is set.

The `active-spaces-limit` setting only has an effect if the **Restricted** feature is set on the user's license. This means that users with a standard Professional License (that have just the **Professional** license feature) are not effected by this limitation.

In order to upgrade such a user to the a fully commercial license you can either remove the **Restricted** feature manually in the Admin Console, or it can be done using the "downgradedefaultlicense" API call (see downgradedefaultlicense_ref), which can be used to remove features from a license.

12.6 How to Restrict Device Registration

As a Provider you may wish to restrict the creation of new TeamDrive installations by your users. For example, the users of a certain Provider may be prevented from using private devices, in order to control the proliferation of company data.

In order to do this, you can configure the Registration Server to require manual approval for every new device registration.

The details are explained in chapter login_without_activation

12.7 How to Setup Two-Factor Authentication

The Reg Server version 3.6 supports two-factor authentication (2FA) using the Google Authenticator App (https://support.google.com/accounts/answer/1066447?hl=en).

You can enable the use of 2FA for a particular Provider by setting `USE_AUTH_SERVICE` to `True`. You must then add the following settings to `LOGIN/PRE_LOGIN_SETTINGS`:

```
enable-login=false
enable-web-login=true
```

This will ensure that the user is directed to the "external" (web-based) login page when logging in to the TeamDrive Client.

The external pages use templates stored by the Registration Server and can be modified for each Provider. Use the Admin Console to upload customised versions of the pages for your users as described in *Manage HTML Templates* (page 19)

Two-factor authentication must be activated individually by each user by entering the following URL in a Web-browser:

```
https://regserver.yourdomain.com/yvva/portal/setup-2fa.html
```

In the future, a link to this page will be made available directly in the client application. Follow the instructions for downloading the Google Authenticator App and activating the 2FA functionality.

Two-factor authentication can also be configured to work with the TeamDrive Web Portal. Following the instructions on how to do this provided by the Web Portal documentation.

Web-Portal users must use the `/portal/setup-2fa.html` page to setup two-factor authentication.

Note that, since the Register Server external authentication pages do not yet support LDAP or Active Directory, it is not possible to use two-factor authentication in combination with LDAP or any other external authentication service.

12.8 How to migrate existing Users, Depots and Licenses to an Account

1. Create a new account as described in *Create Account* (page 25). You can already choose the manager and account members, but both are optional and not required. When moving existing users to an account their licenses and depots will not automatically be moved to the depot. Both are still bound to the user.

2. Click on `Edit Account` to change the account record itself and / or managers, members, licenses and depots.

3. You can create a new license with `Create License` (depends on your access rights) or you can move existing licenses to an account with `Add License`. The license select list is limited to licenses which:

- are not assigned to a TeamDrive user or
- belong to an user which is already a member of this account, but is not the default license of the user and has a license limit > 1.

When you move an existing license to an account, the account will be the new owner of the license and not the user anymore (this is important, if you remove the user from the account, because the user will not be able to use this license anymore).

4. You can create a new depot with `Create Depot` (depends on your access rights) or you can move existing depots to an account with `Add Depot` . The depot select list is limited to depots which:

- are not assigned to a TeamDrive user or
- belong to an user which is already a member of this account.

When you move an existing depot to an account, the depot will be shown under the account, but also still have a Teamdrive user as an owner of the depot, because the TeamDrive Clients need this information to set/change the Admin-User of a space.

CHAPTER

THIRTEEN

AUTO TASKS

There are a number of background jobs that are performed by the Yvva-based `td-regserver` service.

You can review and manage them via the Registration Server Administration Console by clicking **Admin -> Manage Auto Tasks**. See *Manage Auto Tasks* (page 14) for details.

The overall frequency of how often the background service will wake up can be changed by modifying the setting `repeat` in file `/etc/td-regserver.conf`. The default value is 10 seconds.

Note that the frequency of the individual tasks can be defined differently, by changing each task's **Frequency** setting (if required).

13.1 "Send Emails" Task

This process sends out email notifications generated by actions from the Team Drive Clients and Registration Server (e.g. device activation or Space invitation messages, license expiry reminders), which are queued in the Registration Server's internal email queue.

13.2 "Delete Old Messages" Task

Messages not retrieved by Clients will be deleted from the Registration Server's internal message queues after the period defined in the Registration Server settings `<InvitationStoragePeriod>` (e.g. invitations and other client messages, store-forward invitations) and `<InvitationStoragePeriodFD>` (invitations for future devices).

See registration server settings and teamdrive client-server interaction/messages, invitations & invitation types for details on these settings and the various message types.

13.3 "Delete Client IPs" Task

For privacy/data protection reasons, this task removes the Client IP addresses from the Devices table according to the value of `<StoreRegistrationDeviceIPinSeconds>` as described in registration server settings.

13.4 "Update RegServer-List" Task

If TDNS access is active, this task will poll the TeamDrive Master Registration Server to retrieve a list of all Registration Servers within the TDNS network.

Users registered on your Registration Server can only invite users from white listed Registration Servers to their Spaces.

By default, this task will be performed every 12 hours.

The automatic white listing of servers depends on the setting `<TDNSAutoWhiteList>`.

- If set to `True`, new Registration Servers will be automatically white listed.
- If set to `False` you have to enabled each Registration Server manually, using the **Manage Servers** page of the Administration Console.

13.5 "CleanUp" Task

If API logging is enabled (the Provider setting `API_REQUEST_LOGGING` is set to `True`), each API request is logged in a database table. On a busy server, these log entries can significantly increase the size of the Registration Server's database over time.

Enabling this task will remove entries from the API log table, if they are older than 30 days. This task is disabled by default.

This task will also clean up the Client log-files, if they are older than 90 days (See *Client Log Files* (page 75)).

13.6 "CSV Import" Task

Enable this task if you want to manage your users by importing the usernames and other details from an external source via a CSV file. This auto task will perform the import on a periodic basis.

The task can create new users, update existing users and disable users as required. See *Importing Users via CSV Files* (page 39) for details on how to accomplish this.

13.7 "Deactivate/Activate Devices" Task

This task is new in Registration Server 4.5.0. It uses the `MAXIMUM_DEVICES_PER_USER` (see maximum_devices_per_user) provider setting to determine whether user devices must be enabled or disabled in order to ensure that only the required number of devices per user is active.

The task always disables the least recently used (or accessed) devices. This means that a disabled device can be re-enabled by starting the TeamDrive client, which updates the device access time.

The activation of devices changes, on average, every 3 hours according to their usage. If devices are enabled or disabled the task sends an email to the affected user, using the **devices-disabled** email template.

13.8 "Delete Providers" Task

This task deletes Providers that have been marked for deletion.

13.9 "Expire Licenses" Task

If you issue licenses with an expiration date (e.g. by issuing trial licenses or by entering a date in the **Valid until** field manually), this task takes care of sending out reminder emails to the license's user(s), informing them about the upcoming expiration. Once the expiration date has been reached, the license will be invalidated and the user's TeamDrive clients will fall back to their default license.

13.10 "License Report" Task

Create a report on licenses and usage.

13.11 “Remove Auto Created Users” Task

Remove users that have been automatically created by an invitation if they are not activated within `AUTO_CREATED_USER_TIMEOUT` days.

13.12 “Send Notifications” Task

Send notifications of user change events that could not be sent synchronously.

Further details are provided in the chapter user_change_notifications.

CHAPTER

FOURTEEN

CLIENT LOG FILES

TeamDrive clients can upload support requests / bug reports to the Registration Server. To configure this, install the log upload script included in the TeamDrive Registration Server installation package (see Installing the Registration Server client log upload) and change the `RegServer/LogUploadURL` setting to `http://<your-registration-server>/upload/upload.php`

Note: In case of using https instead of http, SSL/TLS **must** be enabled, as the client will not encrypt the log files that get sent

Uploaded log files and bug reports can then be viewed from the `Manage Clients / Download Client Log Files` page.

To recieve an email upload confirmation for new client logs, set a support email in the provider settings support_email.

The support email template `support-notification` can be modified in the `Manage Email Templates` section see html and email templates/email templates/templates for server administration.

CHAPTER

FIFTEEN

UPGRADING THE TEAMDRIVE REGISTRATION SERVER

15.1 General Upgrade Notes

There are two basic approaches to updating a TeamDrive Registration Server: **in-place**, by replacing the software with a newer version on the live system, or starting a **new instance and migrating the configuration** and data (MySQL Database and configuration files) to the new instance.

For older installations, performing a migration to a freshly installed instance might be the better approach, to get rid of accumulated "cruft" and to start from a clean slate.

In case the current system is still running a 32-bit installation, moving to a 64-bit system is **required**, as newer versions of the Registration Server **no longer support 32-bit environments**.

In case the current system uses a Linux OS other than Red Hat Enterprise Linux 6 or a derivative like Cent OS 6, Oracle Linux 6, Scientific Linux 6 or Amazon Linux, you **must** perform the upgrade by starting a new instance and migrating the configuration as outlined in *Moving an Older Installation to a Newly Installed Instance* (page 84).

Updating requires a brief service interruption, as the Registration Server components (e.g. the Apache HTTP Server) need to be stopped while the update is in progress. Short downtimes usually pass unnoticed by the TeamDrive Clients, they will simply try again after a short waiting period. Local Client operations can continue.

The Registration Server-specific MySQL Databases and local configuration files and templates are the crucial pieces of data that need to be preserved during updates. Take backups prior to performing an update and *verify they worked correctly*. In case of an in-place upgrade, the databases and most configuration files can be taken over "as is". When performing a migration to a new instance, the databases and supporting files need to be copied or moved to the new host.

Updates between different Registration Server major versions (e.g. from 3.0.017 to 3.0.018) may require changes to the MySQL table structures.

These changes need to be applied manually prior to starting the services after updating. Reversing these changes (e.g. reverting to the previous database version) requires going back to the previous backup, there is **no automatic roll-back of changes to the database/table structures**.

Starting with version 3.0.018, updates to a new build (e.g. from 3.0.018.0 to 3.0.018.1) can be performed using yum/RPM. Updating from older major versions (e.g. 3.0.017 or 3.0.015) requires manual intervention, as the installations were performed without automatic package management.

15.2 Using version 4.0 with PHP 7.2 / 7.3

Version 4.0 of the Registration Server Admin Console is compatible with PHP 7.2 / 7.3.

However, after upgrading to PHP 7.2 / 7.3 you will experience problems with the Admin Console if you are using the `$dsn2import` setting (in the `/var/www/html/tdlibs/globals.php` file) in order to connect to MySQL.

This is due to the removal of the "mysql" extension in PHP 7 and later. Instead the "mysqli" must be used by the Admin Console.

There are 2 ways to fix this problem:

- Use the `$mysqlConfigFile` variable as specified in admin_console_config.
- Change `mysql:` to `mysqli:` in the string specified by `$dsn2import`.

We recommend using the first method to avoid a similar problem due to changes in the future.

15.3 Upgrading Version 3.5.0 or Later to a Newer Build

The Registration Server Version 4 needs PHP 7.4, because older PHP version are not longer supported by PHP.

At first update your CentOS 7/8 to the latest version:

```
yum update
```

The following update steps differs between CentOS 7 and CentOS 8. To check your system type in:

```
cat /etc/centos-release
```

For CentOS 7 install these additional repositories:

```
yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm␣
↪-y
yum install https://rpms.remirepo.net/enterprise/remi-release-7.rpm -y
yum install yum-utils -y
yum-config-manager --enable remi-php74
yum update php php-cli php-common php-pear php-mysqlnd php-mbstring
```

For CentOS 8 install these additional repositories:

```
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 update php php-cli php-common php-pear php-mysqlnd php-mbstring
```

The next steps are identical for CentOS 7/8:

```
pear uninstall Auth
yum remove php-pear.noarch -y
yum install php74-php-pear.noarch -y
```

Now, proceed with the registration server update.

Note: To enable the 4.0 TeamDrive Registration Server yum repository, you need to download the updated `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
```

Due to an internal change in the td-regserver repo, the update to version 4.6.2 and later requires a special steps. Remove the existing td-regserver package, clean the yum cache and reinstall the package again:

```
yum remove td-regserver
yum clean metadata
yum install td-regserver
```

The use of RPM packages makes updating within a major version from one build to another (e.g. from 4.0.0 to 4.0.1) a fairly straightforward and automatic process.

Usually, you can simply replace the existing packages while the service is running. The update performs an immediate restart of the services (`httpd` and `td-regserver` automatically):

```
[root\@regserver ~]# yum update td-regserver td-regserver-adminconsole yvva
```

Check the chapter releasenotes-4.0 for the changes introduced in each build.

Now update the database to version 4.0 as described in the steps below *Update the database using the linux shell* (page 82)

15.4 In-place Upgrading from 3.0.018 to 3.5.0 or later

These instructions assume a default installation of the TeamDrive Registration Server (version 3.0.018) on RHEL6 or a derivative distribution like CentOS 6 (64-bit) that was set up based on the Registration Server installation instructions or using the TeamDrive Registration Server Virtual Appliance for VMware. They further assume that the MySQL database and Administration Console run locally as well.

The overall procedure is similar in all cases — we'll remove the old software components while retaining the MySQL databases and configuration files, install the current versions of the Registration Server RPM packages and and manually migrate a few configuration settings by performing the following steps:

- Stop the Apache HTTP Server and PrimeBase processes (PBAC)
- Perform a backup of the Registration Server's MySQL Databases and support files
- Remove the PrimeBase Application Environment and related files
- Remove old Apache modules
- Install the new Registration Server RPM package `td-regserver`
- Review/update the configuration files, remove backup configuration files after merging the settings
- Perform necessary conversions of the MySQL table structures
- Review/update the email templates
- Start the TeamDrive Registration Server background service and Apache http Server, check the log files for any errors
- Test the new setup with a local test client before allowing all user Clients to connect to the new instance again

The following paragraphs explain these steps in more detail.

15.4.1 Stop the TeamDrive Services

As a first step, the currently running TeamDrive Registration Server needs to be shut down. If you have any monitoring services that send out alerts for system outages, you might want to disable these beforehand. If your Registration Server is behind a load balancer or firewall, it might make sense to block incoming Client connections from there, too. This prevents unwanted accesses while you are still working on bringing up the updated instance.

Start by stopping the Apache HTTP Server:

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

Next, stop the Registration Server background tasks:

```
[root@regserver ~]# pbctl stop
```

Use `pbctl status` to check that the services have been stopped (their `Status` needs to be `Stopped`) and `ps` or `pstree` to double check that there are no stray `httpd`, `pbeas`, `ase`, `pbas`, `pbac` or `smm` processes running. Use `kill <pid>` or `pkill <name>` to terminate these, if they don't disappear shortly after you issued the stop commands.

15.4.2 Create a MySQL Backup

After all TeamDrive Services have been stopped, you should now create a backup of the MySQL databases, e.g. using `mysqldump`:

```
[root@regserver ~]# mysqldump -u root -p --force \
--max_allowed_packet=64M --databases td2apilog td2reg \
| gzip > td-regserver-mysql-$(date +%Y-%m-%d_%H.%M).sql.gz
```

15.4.3 Backup the old Installation and Configuration Files

Next, create a backup the old PrimeBase Application Environment, Apache Modules and config files, if you don't have a full system backup already (e.g. a VM snapshot) that you could revert to in case of issues.

Note that some of these files might not exist on your local installation. The following sample shell script will skip these and add all existing ones to a backup tar archive named `td-regserver-backup-YYYY-MM-DD.tar.gz` in the current directory:

```
#!/bin/sh
BACKUP="td-regserver-backup-$(date +%Y-%m-%d).tar"

FILES="
/etc/httpd/conf.d/adminconsole.conf
/etc/httpd/conf.d/fastcgi.conf
/etc/httpd/conf.d/pbt.conf
/etc/httpd/conf.d/ssl.conf
/etc/httpd/conf/httpd.conf
/etc/httpd/modules/mod_pbt*.so
/etc/httpd/myssl
/etc/init.d/primebase.boot
/etc/logrotate.d/teamdrive
/etc/php.ini
/etc/php-fpm.d/www.conf
/etc/primebase
/etc/profile.d/custom.csh
/etc/profile.d/custom.sh
/etc/profile.d/primebase.sh
/etc/profile.d/teamdrive.sh
/etc/sysconfig/httpd
/usr/local/lib
/usr/local/lib64
/usr/local/primebase
/var/www/html/activation
/var/www/html/adminconsole"
for a in $FILES
do
  if [ -e $a ]
  then
      tar rvf $BACKUP $a
  fi
done
gzip $BACKUP
```

15.4.4 Review and save values from Configuration File

Before starting the upgrade, please copy a few existing settings. They are stored in a binary file and could not be extracted later on after the old Primebase components are removed. As described in the documentation for version 3.0.018:

http://docs.teamdrive.net/RegServer/3.0.018.8/html/TeamDrive-Registration-Server-Admin-Guide-en/Upgrading.html#review-configuration-files

use the tool `pbee` (PrimeBase Environment File Editor) to review and copy the values from the following settings to store them later on in the admin console after the update:

```
240    Mail Server Address              <SMTP Server hostname>
243    Email Sender Address             <you@yourdomain.com>
244    Host Name                        <reg server hostname>
```

Leave the tool `pbee` with the command `quit`

15.4.5 Install the new Registration Server Software

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.

Note: Please just follow the steps that describe the software installation! The MySQL user and databases have been created already, so there is no need to perform these steps again.

To enable the 3.5 TeamDrive Registration Server yum repository, you need to download the updated `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
```

Now you can simply update the installed packages by entering:

```
[root@regserver ~]# yum update td-regserver td-regserver-adminconsole \
PrimeBase_TD
```

The update removes the old primebase components in `/usr/local/primebase` but will keep your mail templates in the path `/usr/local/primebase/setup/scripts/template/`. They will be imported to the database later on in the update process.

Removing the old primebase components might require additional changes in the configuration. If you used the default teamdrive user and the default password for the mysql database connection, the update will automatically create a new connection definition using the default values. Please change the mysql user and password in the file `/etc/td-regserver.my.cnf` in case that you dont use the defaults.

As described in the Release Notes 3.5 the Apache HTTP Server no longer requires 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. Please remove the FastCGI module with:

```
[root@regserver ~]# yum remove php-fpm
```

Please disable using the "worker" MPM in the file:

```
/etc/sysconfig/httpd
```

and comment out the line:

```
HTTPD=/usr/sbin/httpd.worker
```

to:

```
#HTTPD=/usr/sbin/httpd.worker
```

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

```
#   Per-Server Logging:
#   The home of a custom SSL log file. Use this when you want a
#   compact non-error SSL logfile on a virtual host basis.
CustomLog logs/ssl_request_log \
          "%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%r\" %b"

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

Please update the existing PHP version 5.3 to the latest supported PHP 5.6 version. Follow the steps in chapter configure-php to download the necessary yum repository and activating the PHP 5.6 version.

Proceed with the database update step. This could be done using a web browser or the linux shell. For using a web browser you have to start the apache server again, but you have to make sure, that no client could connect during the database update.

15.4.6 Update the database using the linux shell

To view the databases changes and start the database update use the command:

```
[root@regserver ~]# yvva
```

The upgrade commands will be listed:

```
UPGRADE COMMANDS:
-----------------
To upgrade from the command line, execute:
yvva --call=upgrade_now --config-file="/etc/yvva.conf"

print_changes;;
Print a list of changes will be performed when you run 'upgrade_now'.

upgrade_now;;
Perform upgrade changes to the database (this command cannot be undone).
```

Type in print_changes;; to view the list of changes and start the update with upgrade_now;;.

You will get the output:

```
Upgrade in progress...

Upgrade completed successfully.
```

Exit yvva with quit.

15.4.7 Review Configuration Files

During installation, RPM may detect that some local configuration files differ from the ones to be installed. Instead of overwriting these, RPM will create the distribution's default configuration files as <filename>.rpmnew. Carefully review the differences and manually migrate any relevant changes to the new files before renaming them to their original file names, which will overwrite the previous versions.

15.4.8 Update the MySQL Configuration

Review the content of the /etc/my.cnf configuration file. In particular, make sure that the option max_allowed_packet and max_connections is included in the [mysqld] option group and is set to:

```
[mysqld]
max_allowed_packet=32M
max_connections=512
```

The max_allowed_packet value is necessary for Registration Server Version 3.6.3 and later, to support the upload of client log files.

The max_connections=512 is the minimum value. It might be necessary to increase the value on your system depending on how many clients are connected to your server.

15.4.9 Start the Registration Server Components

Now start the TeamDrive Registration Server background service:

```
[root@regserver ~]# service td-regserver start
Starting TeamDrive Registration Server Auto Tasks:          [  OK  ]
```

Check the log file for any errors:

```
[root@regserver ~]# less /var/log/td-regserver.log
```

Next, start the Apache HTTP Server if not already done above:

```
[root@regserver ~]# service httpd start
Starting httpd:                                            [  OK  ]
```

Check the log files for any errors:

```
[root@regserver ~]# less /var/log/httpd/error_log
```

In case of any errors, check the chapter troubleshooting for guidance.

15.4.10 Log into the Administration Console

Clear your browser cache before accessing the admin console. Set the above email configuration values in the admin console in **Server –> Server Settings –> Email**. The old values must be stored in these new fields:

```
Mail Server Address --> SMTPServer
Mail Server Timeout --> SMTPServerTimeOut
Email Sender Address --> MailSenderEmail
Host Name --> MailSenderHost
```

Check other new values in the provider settings section. Former global server settings are now provider specific settings. A full list of all settings could be found in the chapter settingsChapter

15.4.11 Mail templates

The name of the mail templates beginning with "`td3-`" in the file name changed:

"`td3-privacyinvited-email-utf8`" to "`inv-email-invited`"

"`td3-privacyinvitedsecure-email-utf8`" to "`inv-email-invited-passwd`"

"`td3-privacyinvited-user-utf8`" to "`inv-user-invited`"

"`td3-privacyinvitedsecure-user-utf8`" to "`inv-user-invited-passwd`"

Please check the imported mail templates in the admin console in Manage Templates and customize new email templates for your provider(s).

Please delete the old path `/usr/local/primebase/` using the command:

```
[root@regserver ~]# /bin/rm /usr/local/primebase/ -R
```

15.4.12 Enable the TeamDrive Registration Server at System Boot

If the update was successful and the service is up and running, make sure they get started automatically when the system reboots:

```
[root@regserver ~]# chkconfig | grep td-regserver
td-regserver          0:off   1:off   2:off   3:off   4:off   5:off   6:off
[root@regserver ~]# chkconfig td-regserver on
[root@regserver ~]# chkconfig | grep td-regserver
td-regserver          0:off   1:off   2:on    3:on    4:on    5:on    6:off
[root@regserver ~]# chkconfig | grep httpd
httpd                 0:off   1:off   2:on    3:on    4:on    5:on    6:off
```

15.5 Moving an Older Installation to a Newly Installed Instance

Please contact TeamDrive Systems for further information.

CHAPTER

SIXTEEN

TROUBLESHOOTING

16.1 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.

16.2 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

16.3 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
```

16.4 Common errors

16.4.1 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.

16.4.2 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.

16.4.3 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.

16.4.4 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**.

16.4.5 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.

CHAPTER

SEVENTEEN

RELEASE NOTES - VERSION 4.6

17.1 4.6.4 (2022-11-04)

- Fixed a bug which allowed the use of domains reserved by an account to be use by a user not in the account, when changing the user's email address (REGSERVER-1722).
- Remove unnecessary newlines (n) in a number of email templates (REGSERVER-1724). We assume that the email clients will wrap long lines in text emails as required.
- Admin Console: the confirm deletion dialog for depots was not working.
- Admin Console: account managers can now add the Depots owned by user's of the account to the account.
- Admin Console: it is no longer possible to remove a Depot from an account if the Depot is set to the account default (REGSERVER-1714). The API will also prevent a Depot that does not belong to the account to be set to default.
- It is no longer possible to set the default account license to a license that does not belong to the account, or remove a license from an account that is set to the default (REGSERVER-1713).
- Admin Console: when creating a new account user, the account default license is first assigned to the user, and then the selected license, if any (REGSERVER-1712). This prevents a user from being created if there is an error with the account default license (such as the license is disabled or does not have sufficient users), even if a valid license was selected during creation.
- On login to the Admin Console the Registration Server was sometimes sending a blank OTP in the email.
- Added email templates: **removeuser-request** and **removeuser-confirmed**, and the HTML template: **removeuser-confirmed** (REGSERVER-1705). This is to support the "Rmove User Account" function in the TeamDrive client. Using this function the user can request deletion of their user account and Space data.

  Upon request a **removeuser-request** email is sent to the user containing a link which can be used to confirm deletion. If clicked the **removeuser-confirmed** email is sent to the Provider of the user requesting manual deletion of the user.
- Added the `AdminConsoleURL` global setting, which specifies the URL of the Admin Console if it is different to `RegServerURL` (REGSERVER-1710).
- Admin Console: hitting the ENTER key in a text field in edit forms not longer activates the HTML defined default submit button (REGSERVER-1704). This prevent unwanted actions.
- An unregistered user with a registered domain associated with a external authentication service was not always redirect to the authentication service on login (REGSERVER-1709). This was only working if Provider settings was referencing the same external authentication service.
- Updating license features when no feature bits were previously set was not working (REGSERVER-1702).
- The getregserverlist() API call now returns a specific registration server that must be named.
- The getaccountdata() API call now returns the account flags relating to local encryption, 2FA, and the Super PIN repository.

- The Provider code in the TeamDrive client DISTRIBUTOR file was not used during registration (REGSERVER-1692). Users were incorrectly assigned to the default Provider.
- Fixed a bug when using a TeamDrive Clients version 4.6 or earlier, that caused the error "The specified user is registered on a different Registration Server", during login (REGSERVER-1695).
- The Registration Server will now check for the error: 454 Temporary authentication failure, when sending emails (REGSERVER-1693). This error is handled as if the SMTP server is not reachable. This mean the server will retry sending the same email until it succeeds, or a different error occurs.
- Fixed recognition of the following SMTP errors: 550 Invalid dns, 550 Mailbox unavailable, 550 User unknown. These error cause the email address to be "blacklisted", which means emails are no longer sent to this address. The email address is marked as "bounced" in the user account. This status can be viewed and modified in the Admin Console.
- Added `EnforceHttps` Registration Server setting (REGSERVER-1696). This setting is `True` by default (see enforcehttps for further details).
- Added the Provider settings: `REG_SERVER_PROTOCOL` and `HOST_SERVER_PROTOCOL` (REGSERVER-1696). Possible values for these settings are `https`, `http` and `default`. They are set to `https` by default, which forces clients to use HTTPS for all communications.
- Depot storage and traffic limit notifications via email are now sent to Provider administrators (this includes users with "PROVIDER-MANAGER" rights) as well as the account managers, and depot owner (REGSERVER-1698).

  In the Admin Console, a checkbox is available so that users can opt-out of receiving these emails.
- Added `TEMP_PASSWORD_LENGTH` provider setting which determines the length of a temporary password. Default is 6 characters long.
- The portal page login provided by the Registration Server now supports Email OTP (REGSERVER-1689). Added the **portal-login-otp** HTML template page which is used to submit the OTP and complete login.
- Added support for registering Outlook Plugins for users that use external authentication (REGSERVER-1700). The email template: **device-otp**, was added which is used to send a one-time password used to complete registration.

17.2 4.6.3 (2022-03-24)

- It is now possible to retrieve previously deleted private keys from the Key Repository. This can help to regain access to Space Keys if a mistake is made when upgrading the external authentication encryption, or when disabling the Super PIN.
- 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 an 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.

- Added loginfailed() API call which is used by the Web Portal to count the number of invalid logins.
- The `RedirectorProtocol` setting is now "https" by default. In addition, if "http" is specified then this protocol will only be used if a redirect URL is not explicity set to the HTTPS protocol.
- When deleting a user that is the owner of a depot, the Registration Server now correctly removes the reference to the user from the depot. In particular in the case where the depot also belongs to an account (REGSERVER-1681).
- Login to the Admin Console with an email address used by more than one user will now work correctly (REGSERVER-1679). Which user is selected is unspecified, provided the password is correct.

  After login, check the username of the logged-in user to determine which user has been selected. Use the username of the user rather than the email address in order to login as one of the other users, with the same email address.
- Fixed the search() API function which must return the email address of a provider, when requested to the Host Server.
- Implemented support for OAuth 2.0 and OpenID external authentication (REGSERVER-1691).
- Handle clients that no longer support the Diffie-Hellmann based PBPG 1.0 keys due to incompatibilities in OpenSSL 3.0 (REGSERVER-1688).
- Admin Console: fixed perfomance problem when displaying the user Key Repository statistics (REGSERVER-1690).

17.3 4.6.2 (2011-12-16)

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

- Fixed issue with portal page login that resulted in a "Decryption failed" error.
- 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).

17.4 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.

17.5 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), sent 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**.

  Email based 2FA can be enabled for individual users in the Admin Console on the Edit User page.

  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.

CHAPTER

EIGHTEEN

RELEASE NOTES - VERSION 4.5

18.1 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).

18.2 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.

18.3 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 maxinboxemailperday for details).

18.4 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).

18.4.1 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.

18.5 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 relavent 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.

18.5.1 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).

18.5.2 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).

18.5.3 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

18.5.4 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.

CHAPTER

NINETEEN

RELEASE NOTES - VERSION 4.1

19.1 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).

19.2 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.

19.3 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.

19.4 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 version 4.0 with PHP 7.2 / 7.3* (page 77).
- 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).

19.4.1 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).

19.5 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.

CHAPTER

TWENTY

RELEASE NOTES - VERSION 4.0

20.1 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

20.2 4.0.0 (2018-09-19)

20.2.1 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 *Accounts* (page 21).
- 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).

20.2.2 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>`, `<groupreference>`, `<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).

20.2.3 Administration Console

- Rearranged the menu of the Admin Console and extended the user right levels for viewing, creating, editing and deleting objects (see *User Rights* (page 28)).
- 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.

20.2.4 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 *Encryption Parameters* (page 58)) 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 *Agent/Portal Parameters* (page 57) for details).

CHAPTER

TWENTYONE

RELEASE NOTES - VERSION 3.X

21.1 Change Log - Version 3.6

21.1.1 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.

21.1.2 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.

21.1.3 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).

21.1.4 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).

21.1.5 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).

21.1.6 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.

21.1.7 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.

21.1.8 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)

21.1.9 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 *How to Setup Two-Factor Authentication* (page 69) 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 overwiew 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.

21.2 Change Log - Version 3.5

21.2.1 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.

21.2.2 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.

21.2.3 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 configure-apache-24

- 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)

21.2.4 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)

21.2.5 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

- 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)

21.2.6 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).

21.2.7 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).

21.2.8 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 '|' 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)

21.2.9 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)

21.2.10 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)

21.2.11 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 ususually 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

21.3 Change Log - Version 3.0.019

21.3.1 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)

21.3.2 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)

21.3.3 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)

21.3.4 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

21.3.5 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

21.3.6 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.

21.3.7 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.

21.3.8 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.

21.3.9 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

21.4 Change Log - Version 3.0.018

21.4.1 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)

21.4.2 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)

21.4.3 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)

21.4.4 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)

21.4.5 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

21.4.6 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.

21.4.7 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).

21.4.8 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)

21.4.9 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-4.6.4.0-0.el6.noarch.rpm, files installed in `/usr/local/primebase/setup/scripts`)
  - The PHP-based Administration Console and support files (td-regserver-adminconsole-4.6.4.0-0.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-4.6.4.0-0.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.

21.5 Change Log - Version 3.0.017

21.5.1 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.

21.5.2 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)

21.5.3 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)

21.5.4 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>`)

21.5.5 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"

21.5.6 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`

21.5.7 30017.7 (2014-03-14)

- Fixed nasty typo in `RegServerSetup.xml`

21.5.8 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

21.5.9 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

21.5.10 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

21.5.11 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

21.5.12 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

21.5.13 30017.1 (2014-01-23)

- First build using the scripted build, updated `RegServerSetup.pbt` and included some Admin Console fixes

21.5.14 30017.0 (2013-10-23)

- Not final; Bcrypt is still missing

CHAPTER

TWENTYTWO

APPENDIX

22.1 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.

22.2 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**er**v**er

R