TeamDrive Web Portal Installation and Configuration

Release 3.1.3.0

Paul McCullagh, Eckhard Pruehs

2023

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

INTRODUCTION

The TeamDrive Web Portal provides browser-based access to a TeamDrive user account. Users can login to the Web Portal with their TeamDrive credentials and access the data they have stored in TeamDrive.

In order to provide this service, the Web Portal must have access to the user's data. As a result, it is common practice for companies to setup a Web Portal for their own users.

Due to the obvious security issues, access for the users of a particular Provider to a Web Portal must be explicitly activated on the Registration Server. How to do this is explained in associate_portal_provider.

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

Warning: The TeamDrive Web Portal installation requires a running TeamDrive Registration Server instance. If you are setting up both components on your own premises, please start with setting up the Registration Server as outlined in the TeamDrive Registration Server installation guides. If you are using a Registration Server instance hosted by some other service provider, make sure you can access it and you have performed an initial setup/configuration already.

3.1 Required Skills

When installing the TeamDrive Web Portal, we assume that you have basic knowledge of:

- **VMware:** importing and deploying virtual machines, configuring virtual networking and storage (when using a pre-installed Virtual Appliance)
- **Linux system administration:**
  - Adding/configuring software packages
  - Editing configurations files
  - Starting/stopping services
  - Creating user accounts
  - Assigning file ownerships and privileges
  - Creating and mounting file systems
  - Setting up environment variables
- **Apache Web Server:** installation and configuration, adding and enabling modules, modifying configuration files
- **MySQL Database:** installation and configuration, administration/maintenance, using the MySQL command line client, basic SQL
- Basic knowledge of application server technology.

3.2 Operating System Requirements

We recommend using a recent 64-bit version of **Red Hat Enterprise Linux 8** (RHEL 8) or a derivative distribution like **CentOS 8 Stream**, **Oracle Linux 8** or **Scientific Linux 8** as the operating system platform.

This document is written with this OS environment in mind — the names of packages, configuration files and path names might be different on other Linux distributions. If you have any questions about using other Linux distributions, please contact sales@teamdrive.net .

You will need at least Apache HTTP Server version 2.4 which should be configured using the "prefork" MPM (http://httpd.apache.org/docs/2.4/mod/prefork.html). The prefork option is more scalable under load than the worker option and is usually the default configuration on Linux distributions.

In addition, the TeamDrive Web Portal requires the Yvva Runtime Environment version 1.5.9 or later, and a MySQL Database Server version 8.0 or later.

3.3 Hardware Requirements

The hardware requirements depend on the number of users that will access the Web Portal. Exact sizing will depend on how heavily the portal is used and how many users access the portal concurrently.

To operate a TeamDrive Web Portal you need one or more **64-bit** systems.

CPU usage, RAM, disk storage and network requirements are described below. Since the usage of a Web Portal can differ greatly, our recommendations are only approximate.

Please contact us via sales@teamdrive.net for further assistance.

3.3.1 CPU Requirements

To operate a TeamDrive Web Portal we recommend at least one processor core per 24 users of the portal.

This estimate assumes that only about 10% of all users are actively performing some operation at any given moment. Increase the number of CPU cores if your estimate of the number of active users is higher.

3.3.2 RAM Requirements

The Web Portal starts a TeamDrive Agent for each active user session. Each Agent requires about 100 MB of RAM.

You can assume that the number of agents running is greater than the number of active users (the number of users accessing the portal at any given time). This is because agents running until the user session is closed due to an idle timeout.

3.3.3 Storage Requirements

The main storage requirement is for the Space data that is downloaded from the Hosting Service when a user enters a Space via the TeamDrive Web interface.

The storage requirements are relatively modest because only the "meta-data" (file names and directory structure) of a Space will be stored permanently on the Web Portal.

The rest of the disk space required consists of a file cache which is used for files in transit between the Hosting service and the end-user device. We recommend a cache size of at least 2 GB per Web Portal user plus about 4 MB per Space.

The speed of the storage system used will be decisive for the responsiveness of the Web Portal, in particular when entering a Space. We recommend a system that is capable of at least 100 IOPS per active user of the Portal. As a

rule of thumb we assume that 10% of the users that use a Web Portal are active at any particular time. This means, for example, that if a portal serves 1000 users, then the storage system should be capable of 10000 IOPS.

If a user's account is idle for a certain period of time (for example 1 month), the Web Portal can be instructed to remove the user's data. In this way, the storage can be freed up for other users.

If the user's data is removed from the Web Portal host, the data is not lost, because the Space data is still stored and maintained by the Hosting Server. The only inconvenience for the user is that Spaces will have to be "re-entered" the next time the user logs in to the Web Portal.

3.3.4 Network Requirements

The bandwidth of the Web Portal's network interface plays a vital role in defining the overall performance and responsiveness of the service.

When a user enters a Space, the meta data of the Space will be downloaded to the Web Portal. The speed of this operation will be effected by the speed of inbound connections.

When a user accesses a file in a Space, the file is first downloaded to the Web Portal disk cache for the user, where it is decrypted. The decrypted file is then transferred to the user's device. As a result, the amount of inbound traffic is at least as high as the outgoing traffic.

The system must have IP connectivity, using a fixed IP address and a resolvable fully qualified domain name. This host name is basically the URL that users will use to access the Web Portal. The Web Portal itself needs to be able to properly resolve host names, too.

If the Web Portal is located behind a firewall, please ensure that it is reachable via HTTPS (TCP port 443) by a web browser.

During operation the Web Portal will need to make API calls to an associated TeamDrive Registration Server. For this purpose the Web Portal must be able to establish outgoing HTTPS connections to the Registration Server.

It is possible to use an TeamDrive Authentication Service for the TeamDrive users of the Web Portal, or an external authentication for the administrators of the Web Portal. In this case, the Web Portal must be able to establish HTTP or HTTPS connections (depending on the configuration) with the host running the authentication service.

CHAPTER

FOUR

INTRODUCTION TO THE TEAMDRIVE WEB PORTAL

4.1 TeamDrive Web Portal Overview

The TeamDrive Web Portal consists of a number of components.

Firstly, the TeamDrive Web browser interface (ie. the TeamDrive Client interface that runs in a browser) is served by Apache.

The TeamDrive Web Portal Administration Console and the Web Portal authentication API is served dynamically by the Yvva Apache module `mod_yvva`.

A list of TeamDrive Agents and other administrative information is stored in a Management MySQL Database called `webportal`. This database must be accessible by all components of the Web Portal.

In addition, a Apache-based environment runs the TeamDrive Agents which serve the TeamDrive browser interface. A TeamDrive Agent is a faceless TeamDrive Client which provides a HTTP-based Rest API for the purpose of accessing a TeamDrive user account.

Depending on the scale of an installation, all components: Apache and MySQL may run on one machine or on separate machines.

In the illustration above, you see the Web Portal, which runs as an Apache module, and how it is connected to other components in the TeamDrive system.

4.2 TeamDrive Hosting Basics

Note: The system variables mentioned in this section are set using the Administration Console explained in web_portal_settings.

A TeamDrive Web Portal requires a unique domain name. The domain name is basically the URL for the TeamDrive users that access the Web Portal. This domain name is stored in the `WebPortalDomain` system setting.

The same domain name is also used to access the Administration Console by adding `/admin/` to the base URL:

```
https://webportal.yourdomain.com/admin/
```

Once the TeamDrive Web interface has been served, further calls from the browser will be redirected to the appropriate TeamDrive Agent running in a systemd-sandbox. In order to do this, an Apache rewrite rule is installed which allows Apache to act as a reverse proxy, forwarding calls to the TeamDrive Agent.

As mentioned above, the WebPortal is responsible for running multiple TeamDrive Agent instances in systemd-sandboxes. A systemd-sandbox with an TeamDrive agent is created for each user session.

The users data is not stored in the systemd-sandbox itself. Instead, the TeamDrive agent will be started with the user folder below `\teamdrive`.

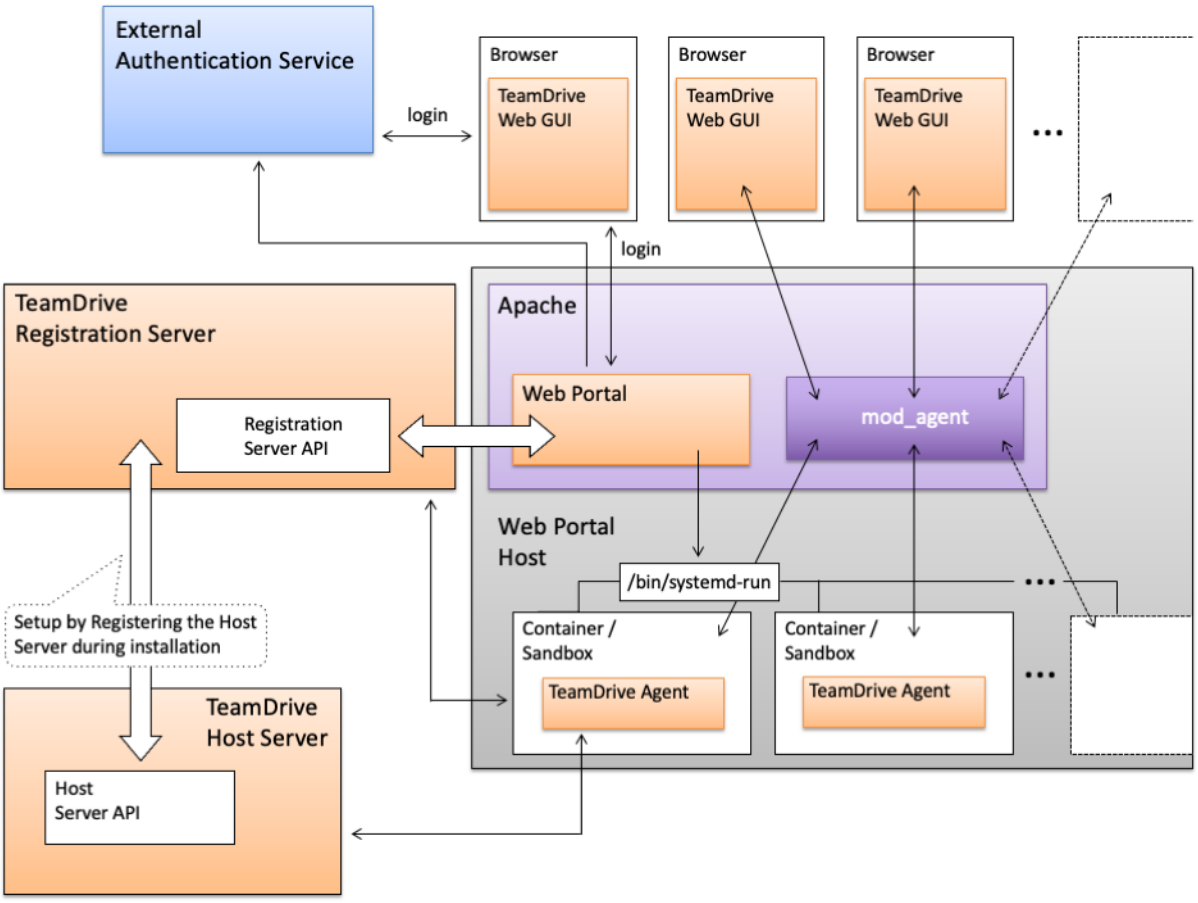

Fig. 4.1: An overview of the Web Portal with connections to other TeamDrive components

The root for a TeamDrive Agent is `\teamdrive\` by default. This path is stored in the `ContainerRoot` system setting. The username of the user is added to this path to produce the user folder for each individual TeamDrive Agent. For example the user for user "td_user_1" is `\teamdrive\td_user_1`.

Under this directory, the TeamDrive agent stores the "meta data" of the Spaces that have been entered, as well as a disk cache for any files in transit.

Note: The Space data stored by the TeamDrive Agents is stored in unencrypted form. For this reason, security of the host system is extremely important.

4.3 Background Tasks Performed by `td-webportal`

The `td-webportal` process is a service that executes background tasks scheduled by the Web Portal.

It uses the Yvva daemon `yvvad` to run the following background tasks at a definable regular interval:

- **Remove Idle Containers**:

  The purpose of this task is to remove Agents that are no longer being used.

  When a user session times out, the TeamDrive Agent exits (stops executing).

  This task removes TeamDrive Agents that are unused for a certain amount of time (indicated by the `RemoveIdleContainerTime`). This period should be longer than the regular user session timeout.

  TeamDrive Agents that are removed are automatically restarted by the Web Portal when the user logs in again.

- **Delete Container Storage**:

  This task removes user data if a the user is inactive for a certain period of time (as specified by the `ContainerStorageTimeout` setting. The user data is the data stored under the `ContainerRoot` directory for a particular user, for example: the directory `\teamdrive\td_user_1` for the user "td_user_1".

  The purpose of the task is to free up unused disk space.

  If a user logs in again after the user data has been deleted the user will find that all of his Spaces have been set to "Inactive". This means that he has to enter a Space again in order to access the data. Since this could inconvenient and time consuming the `ContainerStorageTimeout` should be set to a fairly large period, for example 1 month or more.

- **Remove Old Images**:

  The purpose of this task is to remove TeamDrive Agent version used by the Web Portal which can then be upgraded.

  This task specifically stops running TeamDrive Agents. The current (and new) TeamDrive Agent to be used is specified using the `ContainerImage` setting.

  There are a number of settings which control the behaviour of this task: `RemoveOldImages`, `OldImageTimeout` and `OldImageRemovalTime`.

  `RemoveOldImages` must be to True to enable this task. `OldImageTimeout` is the time, in seconds, that a TeamDrive Agent with an older version must be idle before it is removed. Zero means the TeamDrive Agent is removed, even if it is running. `OldImageRemovalTime` is used to specify when TeamDrive Agent with older version should be removed. You can set it to "now", to remove the TeamDrive Agents immediately, if set to "never", then TeamDrive Agents are only removed if the OldImageTimeout is exceeded. This setting value can also be set to a time (e.g. 03:00, format: hh:mm), or a date (format YYYY-MM-DD hh:mm).

- **Trim Login Log**:

  This task removes login log entries that are older than `MaxLoginLogAge` hours.

- **Swap Containers**:

  If container swapping has been enabled (see container_swapping), then this task moves user data to backup storage, after the TeamDrive Agent has been removed by the "Remove Idle Containers" task.

- **Synchronise Containers**:

  The Web Portal periodically starts containers so that the TeamDrive Agent can sync any changes that may have occurred in space. Currently this is done between 20:00 in the evening and 6:00 in the morning.

  See container_syncing for more details.

CHAPTER

FIVE

OPERATING SYSTEM CONFIGURATION

5.1 Installing a base operating system

Start by performing a minimal OS installation of a recent 64-bit Red Hat Enterprise Linux 8 (8) or derivative Linux distribution (e.g. CentOS 8, Oracle Linux 8), using your preferred installation method (manual install, Kickstart, etc). The details of how to perform this task are out of the scope of this document.

For performing the installation, the system needs to be able to establish outgoing TCP connections (mainly to download additional components).

Boot up the system and log in as the root user, either via the console or via an SSH connection.

Note: CentOS 8 package manager was replaced by DNF. DNF is the next generation version of YUM and intended to be the replacement for YUM in RPM-based systems. YUM is still supported in CentOS 8 but for compatibility reasons with CentOS 7 all subsequent package installations continue to use YUM. On CentOS 8 systems you can replace these calls with DNF.

5.2 Time Synchronization with Chrony NTP Server

We strongly advise that the clocks of all servers in a TeamDrive installation are synchronized using the Network Time Protocol (NTP). For CentOS 8 Chrony will be used and is already installed in general.

For CentOS 8 Chrony will be used instead of NTP. Chrony is already installed in general.

5.3 Disable SELinux

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

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

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

5.4 Firewall configuration

You should configure a local firewall so the server is protected against remote attacks. The only TCP ports that should be reachable from outside are 2021 (SSH, optional for remote administration), 80 (http) and 443 (https).

On a minimal installation please enable access to the following services:

- SSH

- Secure WWW (HTTPS)
- WWW (HTTP)

To configure the firewall, disable the two unnecessary services:

```
firewall-cmd --remove-service=cockpit --permanent
firewall-cmd --remove-service=dhcpv6-client --permanent
```

and enable HTTP (80) and HTTPS (443):

```
firewall-cmd --zone=public --add-service=http --permanent
firewall-cmd --zone=public --add-service=https --permanent
firewall-cmd --reload
```

Enable additional protections based on your local requirements or security policies.

You can check the result with `firewall-cmd --list-all --zone=public`:

```
[root@webportal ~]# firewall-cmd --list-all --zone=public
      public
        target: default
        icmp-block-inversion: no
        interfaces:
        sources:
        services: http https ssh
        ports:
        protocols:
        masquerade: no
        forward-ports:
        source-ports:
        icmp-blocks:
        rich rules:
```

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

- Secure WWW (Port 443 for HTTPS)
- WWW (Port 80 for HTTP)
- DNS Lookup (Port 53 for DNS communication with a public DNS server)

5.5 Installing the Postfix MTA (optional)

If you intend to use the email-based two-factor authentication for accessing the Web Portal Administration Console, or if you want to be notified about Space Volumes running out of disk space via email, the TeamDrive Web Portal needs to be configured to send out these notifications via SMTP.

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

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

We recommend configuring a local Postfix instance to perform this duty. The following packages need to be installed:

```
[root@regserver ~]# dnf install postfix mailx cyrus-sasl-plain
```

The detailed configuration of the local Postfix instance depends heavily on your local environment and how the remote MTA accepts remote submissions and is out of the scope of this document.

See the Postfix SMTP client documentation at http://www.postfix.org/smtp.8.html for details on how to configure Postfix to use a relay server and make sure to test the correct operation by sending local emails using the `mail` command line utility and watching the Postfix log file `/var/log/maillog` for errors.

Once the Postfix service has been configured correctly, ensure that it will be started automatically upon system boot:

```
[root@regserver ~]# chkconfig postfix on
```

CHAPTER

SIX

INSTALLING THE WEB PORTAL COMPONENTS

6.1 Enable the TeamDrive Web Portal `dnf` Repository

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

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

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

This will enable the "TeamDrive Web Portal Version 3.1.3" repository, which you can check by running `dnf repolist` afterwards:

```
[root@webportal ~]# dnf repolist
      repo id                              repo name
      appstream                            CentOS Linux 8 - AppStream
      baseos                               CentOS Linux 8 - BaseOS
      extras                               CentOS Linux 8 - Extras
      td-webportal-3.0                     TeamDrive Web Portal Version 3.1.3 (x86_64)
```

6.2 Download and Install the TeamDrive Web Portal Package

On CentOS 8 please disable the old td-webportal repositories, because they are not available for this version:

```
dnf config-manager --set-disabled td-webportal-1.0
dnf config-manager --set-disabled td-webportal-1.1
dnf config-manager --set-disabled td-webportal-1.2
dnf config-manager --set-disabled td-webportal-2.0
```

Perform the download and installation of the Web Portal installation RPM package using the `dnf` package manager:

```
[root@webportal ~]# dnf install td-webportal
```

The TeamDrive Web Portal depends on the Yvva Runtime Environment version 1.5.9 or later to be installed and configured beside other required software components like the Apache Web Server and Apache SSL module. They will be installed by `dnf` as a dependency on `td-webportal` automatically.

Once the TeamDrive Web Portal software has been installed successfully, you can proceed with the initial configuration.

6.3 Installing the Web Portal HTML Documentation (optional)

The documentation for the Web Portal (in HTML format) can be installed locally, so you can access it directly from the Web Portal (or any other host running an Apache HTTP Server).

To install the HTML Documentation, install the following package via `dnf` from the "TeamDrive Web Portal" repository:

```
[root@webportal ~]# dnf install td-webportal-doc-html
```

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

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

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

CHAPTER

SEVEN

APACHE HTTP SERVER INSTALLATION AND CONFIGURATION

The Apache HTTP server and the `mod_ssl` Apache module should have already been installed as dependencies for the `td-webportal` RPM package. You can verify this with the following command:

```
[root@webportal ~]# dnf install httpd mod_ssl
Setting up Install Process
Package httpd-2.4.37-30.module_el8.3.0+561+97fdbbcc.x86_64 is already installed.
Package mod_ssl-1:2.4.37-30.module_el8.3.0+561+97fdbbcc.x86_64 is already␣
↪installed.
Nothing to do
```

7.1 Update `httpd.conf` and `welcome.conf`

Open the web server configuration file `/etc/httpd/conf/httpd.conf` in a text editor to add the following parameters:

```
Mutex flock
KeepAlive On
KeepAliveTimeout 2
ServerName <Your ServerName>
```

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

```
ServerSignature Off
```

By default, the server version and operating system is also displayed in the `Server` response header field, e.g. `Server: Apache/2.4.37 (CentOS)`. To suppress this output, we suggest updating the `ServerTokens` option as follows:

```
ServerTokens Prod
```

In addition disable the Apache default index page in the configuration file: `/etc/httpd/conf.d/welcome.conf`, by changing: `ErrorDocument 403 /.noindex.html` to `ErrorDocument 403 default`.

7.2 Enable "Prefork" Mode

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

To set the mode, execute:

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

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

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

7.3 Disable Unneeded Apache Modules

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

7.3.1 Apache 2.4

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

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

Re-Enable only the required modules in `/etc/httpd/conf.modules.d/00-proxy.conf`:

```
sed -e '/#LoadModule proxy_module/ s/^#*//' -i /etc/httpd/conf.modules.d/00-proxy.
↪conf
sed -e '/#LoadModule proxy_http_module/ s/^#*//' -i /etc/httpd/conf.modules.d/00-
↪proxy.conf
sed -e '/#LoadModule proxy_wstunnel_module/ s/^#*//' -i /etc/httpd/conf.modules.d/
↪00-proxy.conf
```

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

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

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

7.4 Configure `mod_ssl`

The web-based TeamDrive Web Portal Administration Console should be accessed via an encrypted SSL connection. To facilitate this, add the following to the end of the default `<VirtualHost>` section in `/etc/httpd/conf.d/ssl.conf`:

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

CHAPTER

EIGHT

MYSQL INSTALLATION AND CONFIGURATION

8.1 Installing MySQL Server

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

Alternatively, it's possible to use an external MySQL Server. In this case, you need to make sure that this external MySQL instance is reachable via TCP from the Web Portal (usually via TCP port 3306) and that the `teamdrive` MySQL user account is defined correctly (e.g. the MySQL username in the remote database would become `teamdrive@webportal.yourdomain.com` instead of `teamdrive@localhost`).

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

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

Install the MySQL Client and Server packages from the default repository:

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

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

MySQL 8 requires a charset configuration to work with the Web Portal components. These changes will be done by the mysql_install.sh script later on. The script will update the following config files:

```
/etc/my.cnf.d/client.cnf
/etc/my.cnf.d/mysql-server.cnf
/etc/my.cnf
```

Please start the MySQL server in order to run the secure installation script:

```
[root@webportal ~ ]# service mysqld start
```

Run the secure installation script and follow the recommendations. Make sure to create a password for the MySQL `root` user and take note of it:

```
[root@webportal ~ ]# mysql_secure_installation

      Securing the MySQL server deployment.

      Connecting to MySQL using a blank password.

      VALIDATE PASSWORD COMPONENT can be used to test passwords
      and improve security. It checks the strength of password
      and allows the users to set only those passwords which are
      secure enough. Would you like to setup VALIDATE PASSWORD component?
```

```
      Press y|Y for Yes, any other key for No: No
      Please set the password for root here.

      New password: <mysql_root_pw>

      Re-enter new password: <mysql_root_pw>
      By default, a MySQL installation has an anonymous user,
      allowing anyone to log into MySQL without having to have
      a user account created for them. This is intended only for
      testing, and to make the installation go a bit smoother.
      You should remove them before moving into a production
      environment.

      Remove anonymous users? (Press y|Y for Yes, any other key for No) : Y
      Success.


      Normally, root should only be allowed to connect from
      'localhost'. This ensures that someone cannot guess at
      the root password from the network.

      Disallow root login remotely? (Press y|Y for Yes, any other key for No) : Y
      Success.

      By default, MySQL comes with a database named 'test' that
      anyone can access. This is also intended only for testing,
      and should be removed before moving into a production
      environment.


      Remove test database and access to it? (Press y|Y for Yes, any other key for␣
↪No) : Y
       - Dropping test database...
      Success.

       - Removing privileges on test database...
      Success.

      Reloading the privilege tables will ensure that all changes
      made so far will take effect immediately.

      Reload privilege tables now? (Press y|Y for Yes, any other key for No) : Y
      Success.

      All done!
```

MySQL is now up and running and you can proceed with creating the `teamdrive` user and the MySQL databases required for the TeamDrive Host Server.

8.2 Creating TeamDrive MySQL User and Databases

The TeamDrive Web Portal requires the MySQL databases `webportal`, which will be accessed using a dedicated `teamdrive` MySQL user.

The Web Portal installation package ships with a `mysql_install.sh` script that performs these required configuration steps:

- Modify the local configuration file `/etc/my.cnf`, start and enable MySQL Server at system bootup (only when using a local MySQL Server)
- Create the MySQL user account `teamdrive`, assign the provided password and assign the necessary

database privileges (requires access to the MySQL `root` account)

- Create the required Web Portal MySQL database
- Modify the local Web Portal configuration file `/etc/td-webportal.my.cnf`

The following example demonstrates how to configure the MySQL database using the `mysql_install.sh` script, it assumes that the MySQL database is located on the same system where the TeamDrive Web Portal instance is installed.

You need to have the following information available:

- The password of the MySQL `root` user account you defined while running `mysql_secure_installation`
- The password that you want to assign to the `teamdrive` user

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

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

TeamDrive Web Portal MySQL Database Install Script
--------------------------------------------------

Configuring MySQL database for TeamDrive Web Portal
version |release|

This script will perform the following steps:

 - Modify the local configuration file /etc/my.cnf,
   start and enable MySQL Server
   (only when MySQL Server runs locally)
 - Create the required MySQL user "teamdrive",
   assign the provided password and the required
   database privileges
   (requires access to the MySQL root account)
 - Create and populate the required Web Portal
   MySQL database
 - Modify the local Web Portal configuration file
   /etc/td-webportal.my.cnf

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

mysqld (pid  7490) is running...
Stopping mysqld:                                           [  OK  ]
Changing local MySQL Server configuration...
Backing up existing configuration file /etc/my.cnf...
`/etc/my.cnf' -> `/etc/my.cnf-2015-05-19-17:19.bak'
Starting and enabling MySQL Server...
Starting mysqld:                                           [  OK  ]
Trying to connect to the MySQL server as root...
+---------------+
| MySQL Version |
+---------------+
| 8.0.21        |
+---------------+
Creating teamdrive MySQL user on localhost
Trying to connect to the MySQL server as the teamdrive user...
Creating Web Portal databases...
Updating /etc/td-webportal.my.cnf...
Backing up existing configuration file ...
```

```
`/etc/td-webportal.my.cnf' -> `/etc/td-webportal.my.cnf-2021-03-26-12:44.bak'

Finished!
The MySQL configuration for TeamDrive Web Portal
version |release| is now complete.
```

The MySQL database is now properly configured and populated. As a final test, try logging into the MySQL database from the Web Portal system, using the `teamdrive` user account and the password you defined — you should be able to see and access the TeamDrive Web Portal databases:

```
[root@webportal ~]# mysql -u teamdrive -p<password>
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 51
Server version: 8.0.21 Source distribution

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

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

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

mysql> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| webportal          |
+--------------------+
2 rows in set (0.00 sec)

mysql> QUIT
Bye
```

8.3 CentOS Hardening

We recommend to harden the CentOS system as described in *TeamDrive Server Hardening* (page 27).

The script can be retrieved from TeamDrive Systems.

CHAPTER

NINE

TEAMDRIVE SERVER HARDENING

The server hardening is based on the the CIS Benchmark for CentOS 8 version 2.0.0 which can be downloaded from the Center for Internet Security:

https://www.cisecurity.org/cis-benchmarks/

9.1 CentOS 8 Partition Layout

Partition Layout:

```
- root            42.0 GB
  |               11.0 GB
  |- dev           1.9 GB (tmpfs,noexec,nosuid,nodev)
  |  |- shm        2.0 GB (tmpfs,noexec,nosuid,nodev)
  |- run           2.0 GB (noexec,nosuid,nodev)
  |- sys
  |  |- fs
  |     |- cgroup  2.0 GB (tmpfs)
  |- usr           4.7 GB (nodev)
  |- boot          471 MB (noexec,nosuid,nodev)
  |- opt           950 MB (nosuid,nodev)
  |- proc                 (noexec,nosuid,nodev,hidepid=2*)
  |- home          471 MB (noexec,nosuid,nodev)
  |- tmp           950 MB (tmpfs,noexec,nosuid,nodev)
  |- var                  (noexec,nosuid,nodev)
  |  |- www        471 MB (noexec,nosuid,nodev)
  |  |- ossec      950 MB (nosuid,nodev)
  |  |- spool      471 MB (noexec,nosuid,nodev)
  |  |- tmp        950 MB (noexec,nosuid,nodev)
  |  |- log        4.7 GB (noexec,nosuid,nodev)
  |     |- audit   9.4 GB (noexec,nosuid,nodev)
  |- run                  (tmpfs,noexec,nosuid,nodev)
  |  |- user
  |     |- 0       393 MB (tmpfs)
  |- swap                 (encrypted)

(*) meaning all pids hidden for all users
```

9.2 Service Isolation and Sandboxing

The following services are sandboxed using a 01-sandboxing.conf addin to restrict access to file systems, networks, devices, kernel capabilities and system calls:

- aide: Advanced Intrusion Detection Environment
- auditd: Linux Auditing System (see /etc/audit/rules.d/ for audit rules)

- chkrootkit: Chkrootkit Security Scanner
- chronyd: Network Time Protocol (see /etc/chrony.conf for list of time servers)
- crond: Cronjob
- dbus: inter-process communication
- dnf-automatic-install: synchronizes package metadata
- dnscrypt-proxy: DNS proxy using encrypted DNS
- dnsmasq: DNS-Server
- fail2ban: Fail2ban scans log files and bans IPs that show the malicious signs like too many password failures, seeking for exploits, etc.
- firewalld: Firewall
- haveged: random number generator
- httpd: Apache webserver
- irqbalance: Linux daemon that distributes interrupts over among the processors and cores in your computer system
- mysqld: MySQL database server
- NetworkManager: Program for providing detection and configuration for systems to automatically connect to networks
- php-fpm: Execution of PHP scripts
- polkit: application-level toolkit for defining and handling the policy that allows unprivileged processes to speak to privileged processes
- postfix: mail transport agent
- rkhunter: rootkit scanner
- rsyslog: log processing
- s3d: TeamDrive S3-Daemon (only used on the hosting server)
- sshd: SSH Deamon
- systemd-logind: System service that manages user logins
- systemd-udevd: kernel events processing
- td-hostserver: TeamDrive Hosting Server background task (only used on the hosting server)
- td-regserver: TeamDrive Registration Server background task (only used on the registration server)
- td-webportal: TeamDrive Webportal Server background task (only used on the webportal server)
- tmp.mount: mounting temporary filesystem
- usbguard: USB device watcher

9.3 SSH Authentication, Login and Passwords

- SSL: Disabled TLS 0.9, SSL 3.0, TLS 1.0, TLS 1.1 (see /etc/crypto-policies/back-ends/gnutls.config)
- Booloader: see /etc/default/grub
- SSH Banner: see /etc/issue
- SSH Login on port 2021 instead of 22: Several adjustments in /etc/ssh/sshd_config

- Login parameters in /etc/login.defs: password expiry after 60 days (PASS_MAX_DAYS), set login retries to 5 (LOGIN_RETRIES) with lockouts for failed password attempts, default UMASK set to 022 (/etc/profile, /etc/init.d/functions, /etc/bashrc, /etc/csh.cshrc)
- Password quality, length (min 18 characters, set in /etc/security/pwquality.conf), hashing algorithm, reuse prevention
- OpenSSH Client and Server configured compliant to:

```
- DISA STIG for Red Hat Enterprise Linux 8 V1R7
- CIS Red Hat Enterprise Linux 8 Benchmark for Level 1 - Server
- CIS Red Hat Enterprise Linux 8 Benchmark for Level 2 - Server
- PCI-DSS v3.2.1 Control Baseline for Red Hat Enterprise Linux 8
- Protection Profile for General Purpose Operating Systems (Red Hat Enterprise␣
↪Linux 8)
- Australian Cyber Security Centre (ACSC) ISM Official (Red Hat Enterprise␣
↪Linux 8)
- Health Insurance Portability and Accountability Act (HIPAA) for Red Hat␣
↪Enterprise Linux 8
```

9.4 Kernel adjustments

- Kernel self-protection and exploit mitigation (settings l1tf=”full,force”, mds=”full,nosmt”, nosmt=”force”, spectre_v2=”on”, spectre_v2_user=”on”, spec_store_bypass_disable]=”on”, kvm.nx_huge_pages=”force”, tsx=”off”, tsx_async_abort=”full,nosmt”)
- Restricting access to kernel pointers in the proc filesystem by hiding kernel symbol addresses regardless of privileges
- Disabling of entire ptrace, core dumps (see /etc/sysctl.d/50-coredump.conf, /etc/systemd/coredump.conf) and debugging functionality including debugfs (setting slub_debug=FZ)
- Disabled kexec and kernel module loading
- ASLR with high entropy
- Protected symlinks, hardlinks, fifos and regular files to mitigate TOCTOU (Time-of-check to time-of-use) race conditions and data spoofing attacks
- Prevent use-after-free attacks through poisoning, sanity checks and red zoning of SLUB/SLAB objects
- Randomize kernel stack offset on syscall entry (setting randomize_kstack_offset=on)
- Disabled slab merging, which significantly increases the difficulty of heap exploitation by preventing overwriting objects from merged caches and by making it harder to influence slab cache layout (settings slab_nomerge=””, pti=”on”, vsyscall=”none”, debugfs=”off”, oops=”panic”)
- Mitigate use-after-free vulnerabilities and erase sensitive information in memory by zeroing of memory during allocation and free time (setting init_on_alloc=1, init_on_free=1)
- Randomization of page allocator freelists and the kernel stack offset on each syscal (setting page_alloc.shuffle=1)
- Kernel Page Table Isolation to mitigate Meltdown and prevention of KASLR bypasses
- Disabled vsyscalls to protect against ROP attacks
- Enabling kernel panic mode upon oops to prevent continued operation with compromised reliability
- CPU vulnerability mitigations
- Fully enabled hardening of JIT-compiled BPF to mitigate some types of JIT spraying attacks

9.5 Linux Kernel Runtime Guard

LKRG performs runtime integrity checking of the Linux kernel and detection of security vulnerability exploits against the kernel (https://lkrg.org/). This will be done using the Dynamic Kernel Module Support (DKMS): "An essential feature of DKMS is that it automatically recompiles all DKMS modules if a new kernel version is installed. This allows drivers and devices outside of the mainline kernel to continue working after a Linux kernel upgrade".

A kernel upgrade might need an update of the LKRG package. The current LKRG version is 0.9.6 and supports CentOS 8 kernel versions up to 4.18.0-500. If future kernel releases run into errors, the LKRG package would need to be updated.

9.6 Filesystem

- Adjusted mount options in /etc/fstab
- Encrypted swap device
- Disabled uncommon filesystems

9.7 Network

- Entire IPv6 stack disabled
- IPv4 stack hardening:
- Protection against SYN flood attacks
- Protection against time-wait assassination by dropping RST packets for sockets in the time-wait state
- Protection against IP spoofing through strict mode reverse path filtering
- Protection against Smurf attacks
- Prevent clock fingerprinting through ICMP timestamps
- Prevent man-in-the-middle attacks and minimise information disclosure by disabling ICMP redirect acceptance,sending and echo and also disabling source routing
- Prevent exploits by disabling TCP SACK
- Logging of martian packets
- TCP ISN CPU Information Leak Protection by using the tirdad kernel module
- Disabled uncommon network protocols and (obsolete) services and wireless networking

9.8 Firewall

- Using systemd sandboxed firewalld with nftables backend and "drop" as default zone
- Incoming traffic allowed for: 2021 (SSH), 80 (HTTP) 443 (HTTPS)

9.9 Shell

- Deinstalled unsed shells: tcsh, csh, ash, ksh, zsh, es, rc, esh, dash, screen
- Default shell: tmux with auto-logoff (see /etc/tmux.conf, /etc/profile.d/timeout.sh) (tmux hint: Copy & Paste using mouse by pressing shift-key)

9.10 Disabled services

Ensured that unused services are disabled:

- autofs
- avahi-daemon
- bind9
- bluetooth
- chargen-dgram
- chargen-stream
- chrony-wait
- cups
- cups-browsed
- daytime-dgram
- daytime-stream
- dhcpd
- discard-dgram
- discard-strea
- dovecot
- echo-dgram
- echo-stream
- hidd
- irqbalance
- isc-dhcp-server
- isc-dhcp-server6
- kdump
- lpd.service
- named
- nfs
- nfs-server
- nfslock
- nginx
- nis
- nmb
- ntalk
- ntpd
- ntpdate
- portmap
- proftp
- pure-ftpd

- rexec.socket.service
- rhnsd
- rlogin.socket.service
- rngd
- rpcbind.service
- rpcbind.socket
- rpcgssd
- rpcidmapd
- rpcsvcgssd
- rsh.socket.service
- rsyncd
- samba-ad-dc
- sendmail
- slapd
- smb
- snmpd
- sntp
- squid
- systemd-timesyncd
- tcpmux-server
- telnet.socket.service
- tftp.socket
- time-dgram
- time-stream
- vsftpd
- vsftpd
- xinetd
- ypserv
- systemd-coredump.service
- plymouth-halt.service
- plymouth-poweroff.service
- plymouth-quit-wait.service
- plymouth-reboot.service
- plymouth-switch-root.service
- plymouth-kexec.service
- plymouth-quit.service
- plymouth-read-write.service
- plymouth-start.service

9.11 Package Management and Automatic (Security) Updates

- Enabled gpg check for all repositories and for local packages
- Using dnf-automatic and needrestart for update notification/installation and restart of services, see /etc/dnf/automatic.conf

9.12 Virus check

ClamAV for Linux (see https://www.clamav.net) is installed on the server. The ClamAV service needs 1 GB RAM and will use 1 full CPU core during the scan process. See /etc/clamd.d/scan.conf for scan configuration and parameters.

Signature databases from ClamAV and additional 3rd party signature databases via clamav-unofficial-sigs https://github.com/extremeshok/clamav-unofficial-sigs

9.13 Rootkit Scanner

Two rootkit scanner are installed *chkrootkit* with daily scan interval and after a reboot:

http://www.chkrootkit.org

and *rkhunter* with daily scan interval including the forensic unhide module to detect hidden processes and TCP/UDP ports:

https://rkhunter.sourceforge.net

9.14 RNG and Entropy

- RDRAND distrusted by the kernel as an entropy source
- Using systemd sandboxed haveged (HAVEGE algorithm) as random number generator daemon for high entropy https://github.com/jirka-h/haveged

9.15 Fail2Ban

Fail2ban (see https://www.fail2ban.org) scans log files (e.g. /var/log/apache/error_log) and bans IPs that show the malicious signs – too many password failures, seeking for exploits, etc. Generally Fail2Ban is then used to update firewall rules to reject the IP addresses for a specified amount of time, although any arbitrary other action (e.g. sending an email) could also be configured. Out of the box Fail2Ban comes with filters for various services (apache, courier, ssh, etc).

Whitelist your own IPs in: /etc/fail2ban/jail.local

Fail2ban is activated for Apache, PHP, postfix and SSH with these jails: apache-auth, apache-badbots, apache-noscript, apache-overflows, apache-shellshock, php-url-fopen

To check currently banned IPs:

```
fail2ban-client banned
```

9.16 fapolicyd

Setting and enforcing a policy that either allows or denies application execution based on a rule set efficiently prevents the execution of unknown and potentially malicious software.

Software installed using dnf will be automatically whitelisted.

In case of installing own scripts, you have to whiteliste them, see:

```
https://access.redhat.com/documentation/de-de/red_hat_enterprise_linux/8/html/
↪security_hardening/assembly_blocking-and-allowing-applications-using-fapolicyd_
↪security-hardening#marking-files-as-trusted-using-an-additional-source-of-trust_
↪assembly_blocking-and-allowing-applications-using-fapolicyd
```

9.17 Intrusion Detection (IDS/File Integrity)

Daily AIDE scan and check https://aide.github.io/

The fapolicyd software framework controls the execution of applications based on a user-defined policy. This is one of the most efficient ways to prevent running untrusted and possibly malicious applications on the system ("restrictive" policy rule set defined in /etc/fapolicyd/rules.d/*).

More informations:

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html/security_hardening/assembly_blocking-and-allowing-applications-using-fapolicyd_security-hardening

9.18 DNSCrypt

DNSCrypt with DNSSEC using systemd sandboxed dnscrypt-proxy and dnsmasq as local DNS caching server.

DNSCrypt is a protocol that authenticates communications between a DNS client and a DNS resolver. It prevents DNS spoofing. It uses cryptographic signatures to verify that responses originate from the chosen DNS resolver and haven't been tampered with.

Related conf-Files:

```
- /etc/NetworkManager/NetworkManager.conf --> dns=none
- /etc/resolv.conf --> nameserver 127.0.0.1
- /etc/systemd/resolved.conf --> DNSStubListener=no
- /etc/dnscrypt-proxy/dnscrypt-proxy.toml
```

The DNSCrypt will load and use a DNS server from this list:

https://dnscrypt.info/public-servers

In case you have to use your own DNS server, remove the immutable flag from:

```
chattr -i /etc/resolv.conf
```

and change the nameserver in /etc/resolv.conf to your own value.

9.19 NTP

Secure NTP with NTS (Network Time Security, RFC 8915) via systemd sandboxed chronyd

9.20 Accounting and Auditing

Using comprehensive auditing rules compliant to:

- DISA STIG for Red Hat Enterprise Linux 8 V1R7
- CIS Red Hat Enterprise Linux 8 Benchmark for Level 2 - Server
- PCI-DSS v3.2.1 Control Baseline for Red Hat Enterprise Linux 8
- Protection Profile for General Purpose Operating Systems (Red Hat Enterprise Linux 8)
- Australian Cyber Security Centre (ACSC) ISM Official (Red Hat Enterprise Linux 8)
- Health Insurance Portability and Accountability Act (HIPAA) for Red Hat Enterprise Linux 8

9.21 PHP (only Registration Server)

- Using OWASP recommended security configuration
- Using systemd sandboxed FastCGI Process Manager (FPM)

9.22 CentOS Hardening Check

To check the hardening score, use the Lynis - Security auditing tool and ossec benchmark. Start both checks with:

```
/root/hardening/benchmark.sh
```

Lynis generates a test result after 5 minutes analyzing the system with a green, yellow and red status and calculates a hardening index which should be 97 of 100.

After the Lynis check, the OpenSCAP scanner will be started directly:

https://www.open-scap.org

The OpenSCAP scanner executes the following 8 CIS checks which takes about 25 minutes in total (an overview and further descriptions of the test can be found here https://www.mankier.com/8/scap-security-guide):

```
- CIS Red Hat Enterprise Linux 8 Benchmark for Level 1 - Server
- CIS Red Hat Enterprise Linux 8 Benchmark for Level 2 - Server
- PCI-DSS v3.2.1 Control Baseline for Red Hat Enterprise Linux 8
- ANSSI-BP-028 (enhanced)
- Health Insurance Portability and Accountability Act (HIPAA)
- Australian Cyber Security Centre (ACSC) ISM Official
- Protection Profile for General Purpose Operating Systems
- DISA STIG for Red Hat Enterprise Linux 8
```

Each check will generate a html result file located in:

```
/root/hardening/
```

9.23 Known problems caused by the hardening

An dnf update might fail in case of updating the “setup”-package. To fix the problem:

```
chattr -i /etc/shells
dnf update
chattr +i /etc/shells
```

If dnf update fails with “glibc-devel-2.28-225.el8.i686 has inferior architecture” use:

```
dnf update --allowerasing
```

CHAPTER

TEN

PRE-INSTALLATION TASKS

10.1 Mount the Space Storage Volume

The root directory specified by the `ContainerRoot` setting contains the mount points for all TeamDrive Agents on the host.

The container root (by default `/teamdrive`) is the mount point for a dedicated file system that provides the requirements outlined in chapter storage-requirements.

By default, the directory `/teamdrive` has already been created by the `td-webportal` RPM package.

All data will be written to this directory as belonging to the apache user.

Mount the file system and create the respective mount entry in `/etc/fstab` to enable automatic mounting of the file system at bootup. Please consult your Operating System documentation for details on how to perform this step.

Make sure to set the rights to:

```
chown apache:apache /teamdrive
```

10.2 TeamDrive Agents Sandboxing

The Web Portal use a systemd-sandboxing to run the TeamDrive Agent (see https://www.redhat.com/sysadmin/mastering-systemd for details). A TeamDrive Agent is started for each user that logs into the Web Portal.

The systemd-sandboxing makes sure, that the TeamDrive Agent started for an user has only access to the users folder below /teamdrive.

The sandboxing script will be installed together with the web portal and therefor no additional manual configuration is necessary.

10.3 Installing the TeamDrive Agent

The current version of the TeamDrive Agent used by the Web Portal is stored in the `MinimumAgentVersion` setting. The `ContainerImage` setting stores the name of the Container image currently in use by the Web Webportal. If the version of the Agent in `ContainerImage` is less than `MinimumAgentVersion` it will be automatically updated.

If this required TeamDrive Agent does not exist on the host then it will be automatically download and installed on your host.

To install or update the TeamDrive Agent used by the Web Portal use the upgrade command: start `yvva` and execute `upgrade_now;;`:

```
[root@webportal ~]# yvva
Welcome to yvva shell (version 1.5.13).
Enter "go" or end the line with ';;' to execute submitted code.
For a list of commands enter "help".

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

upgrade_now;;
Upgrade the database structure and agent sandbox container (this command cannot be␣
↪undone).
```

Leave the `yvva` shell by typing `quit`.

10.4 Installing SSL certificates

The default Apache HTTP Server installation ships with self-signed SSL certificates for testing purposes. We strongly recommend to purchase and install proper SSL certificates and keys and to adjust the configuration in file `/etc/httpd/conf.d/ssl.conf` accordingly before moving the server into production.

The exact installation process depends on how you obtain or create the SSL key and certificate, please refer to the respective installation instructions provided by your certificate issuer.

10.5 Starting the Web Portal

After all configuration steps have been performed, we can start the TeamDrive Web Portal to conclude the initial installation/configuration.

10.5.1 Starting `td-webportal`

To activate the `yvvad`-based `td-webportal` background task you have to start the service using the provided init script.

The configuration file `/etc/td-hosting.conf` defines how this process is run. You usually don't have to modify these settings.

To start the `td-webportal` program, use the `service` command as user root:

```
[root@webportal ~]# service td-webportal start
Starting TeamDrive Web Portal:                             [  OK  ]
```

Use the `status` option to the `service` command to verify that the service has started:

```
[root@webportal ~]# service td-webportal status
yvvad (pid  2506) is running...
```

If `td-webportal` does not start (process `yvvad` is not running), check the log file `/var/log/td-webportal.log` for errors. See chapter Troubleshooting for details.

10.5.2 Starting the Apache HTTP Server

Now the Apache HTTP Server can be started, which provides the TeamDrive Web Portal functionality via `mod_yvva`.

You can start the service manually using the following command:

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

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

Check the log file `/var/log/httpd/error_log` and `/var/log/td-webportal.log` for startup messages and possible errors:

```
[notice] Apache/2.4.37 OpenSSL/1.1.1g configured
-- resuming normal operations
[notice] mod_yvva 1.5.4 ((Aug 13 2020 18:27:47) loaded
[notice] Logging (=error) to: /var/log/td-webportal.log
```

Please consult chapter troubleshooting if there is an error when starting the service.

CHAPTER

ELEVEN

INITIAL WEB PORTAL CONFIGURATION

A Web Portal is connected to a single Registration Server. On the other hand, Registration Server may be connected to multiple Web Portals, with each Web Portal responsible for a different Provider.

A single Web Portal can also provide web services for the users of a number of Providers, as long as the Providers are all on the same Registration Server.

A Web Portal that is configured to support an specific external Authentication Service has further restrictions. Such a Web Portal can only support one external Authentication Service. However, this is normally not necessary because the Web Portal will automatically re-direct to the external authentication service associated with the user (see authservice for more details).

11.1 Associating the Web Portal with a Provider

Before you can activate your Web Portal you need to associate your Web Portal with a specific Provider account on the Registration Server. This can be performed via the Registration Server's Admin Console, which you can usually access via the following URL:

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

Please see the Registration Server Manual for details. Note that Registration Server 3.5 is required to run a Web Portal.

Log in with your provider login and click the tab **Server Management** and then click on **Provider Settings**. In the section **Provider Settings**, click the tab labelled **API**.

Select the `API_WEB_PORTAL_IP` setting and click "Set" to activate The setting. Enter the IP address of the Web Portal and click "Save" to apply this change.

As mentioned above, it is possible to associate the use of a single Web Portal with a number of Providers. If this is desirable, then follows the procedure above for the addition Providers.

Only users of the Providers associated in this manner will be able to access the Web Portal.

11.2 Activating the Web Portal

From a desktop system that can connect to the Web Portal via HTTPS, start a web browser like Mozilla Firefox, Google Chrome (or any other modern web browser) and start the configuration process by opening the following URL in your browser:

https://webportal.yourdomain.com/admin/

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

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

Alternatively, you can access the Setup Page via an unencrypted HTTP connection. You will have to uncomment the rewrite rules in the apache config file `/etc/httpd/conf.d/td-webportal.httpd.conf` in order to enabled HTTP access. When you access the setup page using HTTP you will be prompted to proceed using an insecure connection.

When everything is configured correctly, you will see the TeamDrive Host Server Setup page that will guide you through the initial configuration:

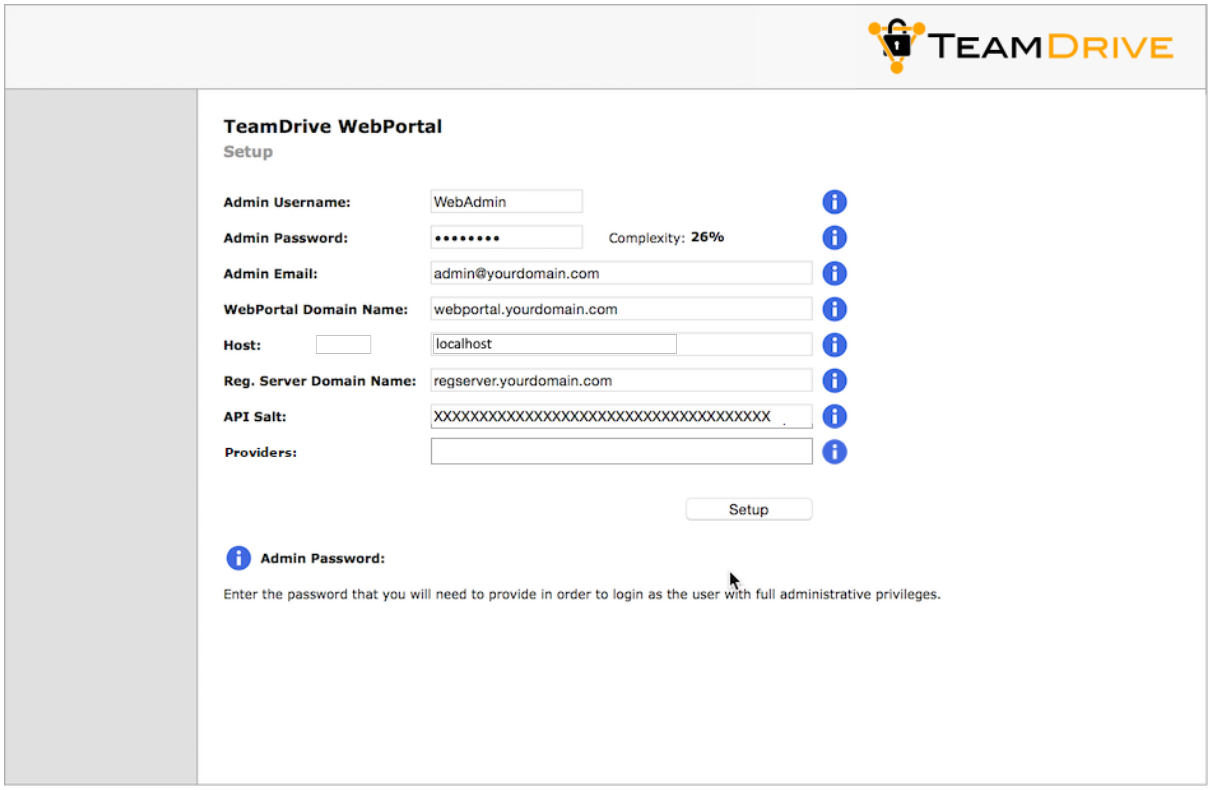

Fig. 11.1: Web Portal Setup Page

Fill out the fields according to your environment and requirements:

Admin Username The name of the user account with full administrative (superuser) privileges.

Admin Password The administrator password that you need to provide to login to the Web Portal Administration Console.

Admin Email The email address of the Administrator. This field is optional. This email address is used for 2-factor authentication (if enabled).

Web Portal Domain Name This is the domain name of the host running the Web Portal. It must be a fully-qualified and resolvable domain name.

Registration Server Name All Web Portals must be registered with a Registration Server. Enter the name of the Registration Server here. This is the value of the `RegServerName` Registration Server global setting.

Please contact TeamDrive Systems for the correct value if you don't manage your own Registration Server.

Registration Server Host Enter the fully qualified domain name of the Registration Server here. **Please contact TeamDrive Systems if you need assistance.**

On the Registration Server, the IP address of the Web Portal must be entered in the appropriate Provider `API_WEB_PORTAL_IP` setting. This will identify the Web Portal when it calls the Registration Server to check user credentials.

Setup will ping this host to ensure that the Registration Server is reachable.

API Key The API Key is a code that allows the Web Portal to validate calls to the Registration Server's API. This value must match the value of the `APIChecksumSalt` setting on the Registration Server to avoid "man in the middle"-attacks. Please consult the Registration Server Documentation on how to obtain it or contact TeamDrive Systems for the correct value if you don't manage your own registration server.

Providers This is a comma separated list of Providers codes. Only users belonging to these Providers will be able to access this Web Portal. If you do not specify any Providers, then all users a the Registration Server will be allowed to login to the Web Portal.

After you have entered all the required details, click **Setup** to initiate the Web Portal configuration and registration process with the Registration Server. An error will occur if the setup process is unable to contact the Registration Server.

This may be due to either network problems or incorrect input, as indicated by the error message.

11.3 Setup and Administration

Upon successful configuration, you will be presented with the Web Portal's Administration Console Login Screen.

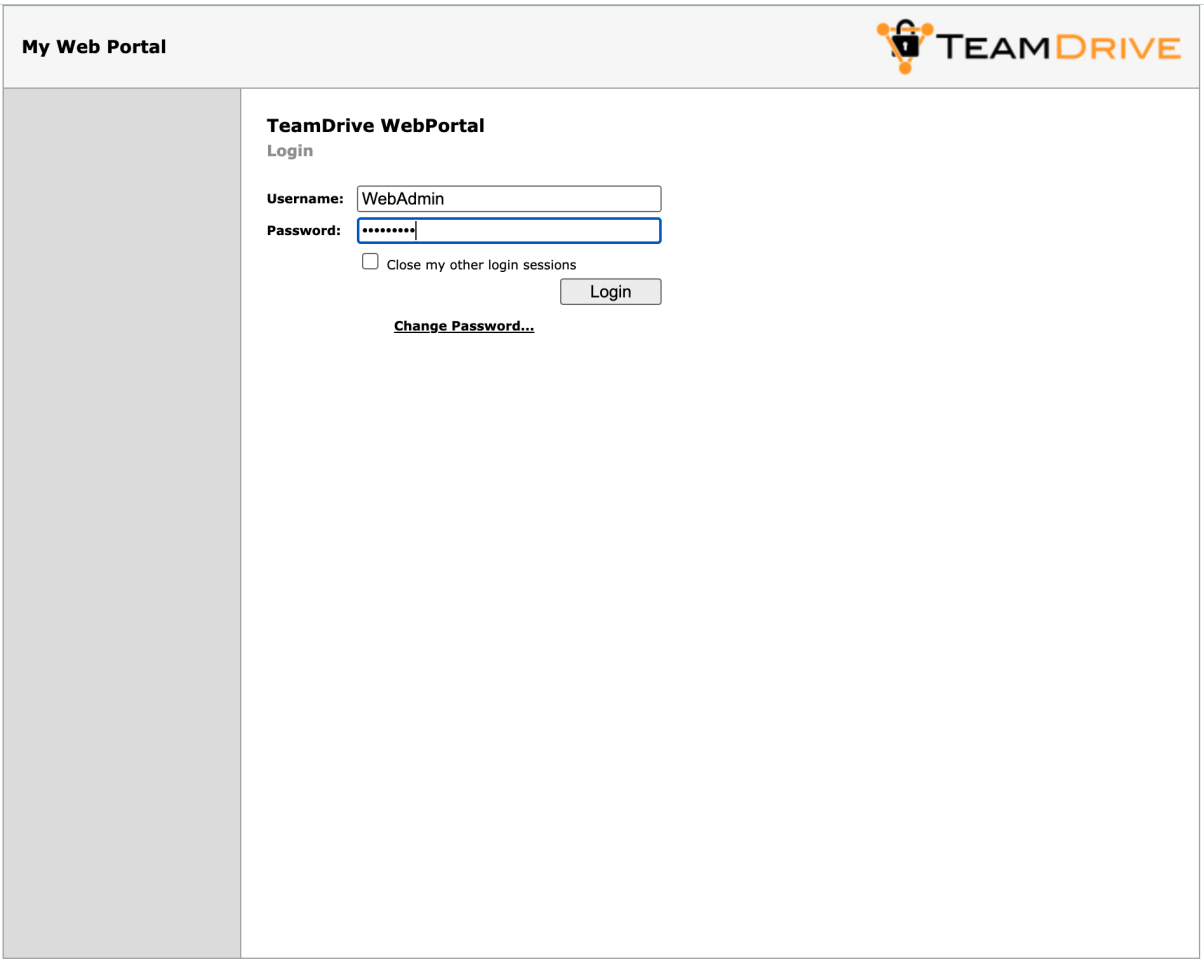

Fig. 11.2: Web Portal Admin Console: Login Screen

Enter the username and password you defined during the initial setup to log in.

After login, you will see the Web Portal's Administration Console Home Screen.

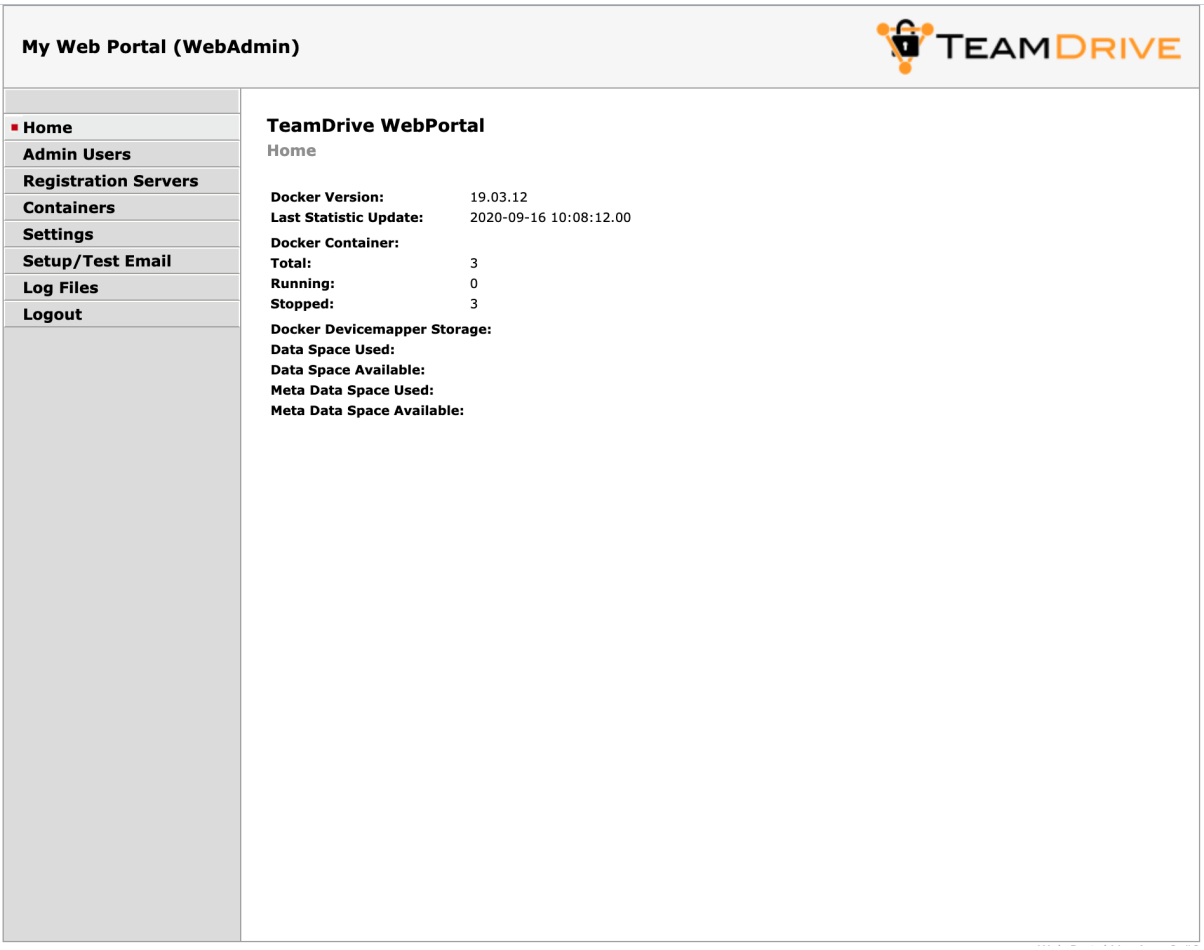

Fig. 11.3: Web Portal Admin Console: Home Screen

At this point, you have concluded the Web Portal's basic configuration and registration. See the *TeamDrive Web Portal Administration Guide* for more details on how to use the Administration Console and how to accomplish other configuration tasks. In case of using a white label version please proceed with the next step otherwise step over to the section *Testing Web Access* below.

11.4 Testing Web Access

The Web Portal has now been set up. To test its functionality, start a web browser and enter the URL of the Web Portal:

https://webportal.yourdomain.com/

Login to a user account belonging to one of the Providers associated with the Web Portal.

If login fails, check your username and password. If this is correct, begin by checking the Web Portals log file for errors.

The log file can be viewed by selecting the **Log Files** menu item and then clicking on **td-webportal.log** in the Web Portal's Administration Console.

CHAPTER

TWELVE

POST-INSTALLATION TASKS

12.1 Startup Sequence / Dependencies

To ensure a proper service start and to minimize error messages during Web access, the following startup sequence of the TeamDrive Web Portal components and services should be observed.

1. Mount the user data volume on the host
2. Start the Web Portal MySQL database service
3. Start the `td-webportal` background service
4. Start the Apache HTTP Server

12.2 Starting the Apache HTTP Server at Boot Time

To ensure that Apache HTTP Server starts up automatically at system bootup time, use the following command to enable it:

```
[root@webportal ~]# chkconfig httpd on
```

Note: It's important, that the MySQL service starts before the Apache will start. On CentOS 8 edit the file:

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

and add at the end of the line starting with `After=` the entry `mysqld.service`. This will ensure, that the Apache will start after the MySQL service.

12.3 Starting TeamDrive Service at Boot Time

To start the TeamDrive Web Portal background service `td-webportal` at boot time, use the following command to enable it:

```
[root@webportal ~]# chkconfig td-webportal on
```

12.4 Next Steps

This concludes the basic installation and configuration of the TeamDrive Web Portal. Please consult the *TeamDrive Web Portal Administration Guide* for additional information on advanced administrative tasks and configuration steps.

CHAPTER

THIRTEEN

TROUBLESHOOTING

13.1 List of relevant configuration files

/etc/httpd/conf.d/td-webportal.httpd.conf: The configuration file that loads and enables the TeamDrive Web Portal Server-specific module for the Apache HTTP Server: `mod_yvva.so`.

`mod_yvva.so` is responsible for providing the web-based Host Server Administration Console as well as an API used for authentication.

The file also contains various Apache "rewrite" rules required by the Web Portal.

Note: The rewrite rules in this file are disable by default. This is because it is assumed that HTTPS is always used to access the Web Portal.

Enable the rewrite rules only if you are certain that HTTP access may be used.

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

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

/etc/td-webportal.my.cnf: This configuration file defines the MySQL credentials used to access the `webportal` MySQL database. It is read by the Apache module `mod_yvva` and the `yvvad` daemon that runs the `td-webportal` background tasks and the `yvva` command line client.

/etc/yvva.conf: This configuration file contains configuration settings specific to the Yvva Runtime Environment that effect the `mod_yyva` Apache module and the `yvva` command line shell.

13.2 List of relevant log files

In order to debug and analyse problems with the Web Portal configuration, there are several log files that you should consult:

/var/log/td-webportal.log: The log file for the Yvva runtime which provides the web-based Administration Console, and the Web Portal authentication API. Errors that are incurred by the Web Portal background tasks are also written to this file.

Consult this log file when the Web Portal has issues in contacting the Registration Server, errors when handling API requests or problems with the Administration Console.

You can increase the amount of logging by changing the Yvva setting `log-level` from `notice` to `trace` or `debug` in the `yvva.conf` file:

```
log-level=trace
```

After changing `yvva.conf` you need to restart the Apache HTTP Server service using `service httpd restart`.

This log file is also used by the `td-webportal` background service. Check the log file to verify that background tasks are being processed without errors.

The log file location can be configured by changing the file name passed to the `log-file` option in the configuration file `/etc/td-webportal.conf`. The log level can be increased by changing the default value `notice` for the `log-level` option to `trace` or `debug`.

Changing these values requires a restart of the `td-webportal` background process using `service td-webportal restart`.

`/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.

13.3 Enable Logging with Syslog

As outlined in list of relevant log files, the TeamDrive Web Portal logs critical errors and other notable events in a log file by default.

It is now possible to redirect the log output of the Yvva runtime components to a local `syslog` instance instead.

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 Yvva component.

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

```
log-file=syslog:webp-bkgr
```

You need to restart the `td-webportal` background service via `service td-webportal 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 11:57:33 localhost webp-bkgr: notice: yvvad startup
Jun 23 11:57:33 localhost webp-bkgr: notice: Using config file:
/etc/td-webportal.conf
Jun 23 11:57:33 localhost webp-bkgr: notice: No listen port
Jun 23 11:57:33 localhost webp-bkgr: notice: yvvad running in repeat 60
(seconds) mode
```

To enable syslog support for the Web Portal API and Administration Console, edit the `/etc/yvva.conf` file as follows:

```
log-file=syslog:webp-httd
```

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 12:06:04 localhost webp-httd: notice: mod_yvva 1.2.1 (May 21 2015
11:00:12) startup OK
```

13.4 Common errors

13.4.1 Web Installation: "500 Internal Server Error"

This error can be triggered by several error conditions. Check the log file `/var/log/td-webportal.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] "startup.yv" (80)
```

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
'webportal.yourdomain.com' (107)
[Error] "startup.yv" (80)
```

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

```
[Error] -12036 (1045): Access denied for user 'teamdrive'@'localhost' (using
password: YES)
[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-webportal.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 'webportal.yourdomain.com' is not allowed to
connect to this MySQL server
[Error] "startup.yv" (80)
```

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

13.4.2 Errors When Accessing the Registration Server

If the Web Portal fails to contact the Registration Server, check the `/var/log/td-webportal.log` log file, as well as `/var/log/td-regserver.log` on the Registration Server for hints.

See the Troubleshooting chapter in the Registration Server Installation Manual for details.

Note: Note that Registration Server version 3.5 or later is required by the Web Portal.

CHAPTER

FOURTEEN

RELEASE NOTES - VERSION 3.1

14.1 3.1.3 (2023-11-13)

- External authentication now redirects to "/external-authentication/finish". This fixes the issue with Web Portal external authentication and 2-Factor authentication.
- Set client version to 5.0.8.3464

14.2 3.1.2 (2023-07-18)

- Set client version to 5.0.6.3386
- The `UseEmbeddedLogin` setting has been removed. This means there is no longer an option to embedded in the TeamDrive Agent GUI (WEBCLIENT-459). This is because most external authentication services do not support embedding in a iFrame for security reasons (for example Microsoft Azure).
- Fixed an update problem (from 3.1.1) which concerned the Container Log table (WEBCLIENT-460).

14.3 3.1.1 (2023-05-23)

- Set client version to 5.0.2.3338
- Added `SyncProviderList` setting (WEBCLIENT-456). This is a comma separated list of Provider codes. All containers of users that belong to these Providers will be periodically synchronised regardless of the `SyncInboxesOnly` setting.
- The Web Portal now writes a "Container Log" which traces the main events and activity regarding a container (WEBCLIENT-457). This includes:
  - `START WEBUI/INBOX` - The container was started because a user logged in.
  - `SYNC WEBUI/INBOX` - The container was started by the auto-sync background task.
  - `STOPPED WEBUI/INBOX` - The container has stopped. Note this log entry is only created when the Web Portal becomes aware that a container is no longer running so this may not be the actual shutdown time.
  - `DELETE FINAL` - The container was completed removed, including all local storage and backups.
  - `SHUTDOWN WEBUI/INBOX` - The container was stopped.
  - `DELETE LOCAL` - The container's local storage was deleted.
  - `ERROR WEBUI/INBOX` - An error occurred when starting the container.
  - `CREATE WEBUI/INBOX` - A new container was created for an inbox or a user that logged in for the first time.
  - `CREATE BACKUP` - The container database and settings have been copied to the Cloud Storage.

  - `RESTORE BACKUP` - The container database and settings have been restored from Cloud Storage.
  - `REMOVE BACKUP` - The Cloud Storage backup of the container has been removed.
  - `ERROR RESTORE` - An error occurred while restoring the container from Cloud Storage.
- Fixed a problem with the flag that indicates that the local databases exists. This can effect the auto-sync function which starts a container regularly so that changes to spaces are applied even if the user does not login.
- Fixed upgrade of the WP_Container table which fails with the error: Invalid default value for 'ActiveTime'.

14.4 3.1.0 (2022-10-25)

- Set client version to 4.8.0.3249
- The setting `BuildBinaryName` is now read-only and is set to the name of the Agent binary executable from the Agent archive on upgrade. The binary is then renamed to "teamdrived.bin" which is the fixed name now used by the Web Portal (WEBCLIENT-451).
- The Web Portal will now periodically start containers so that the TeamDrive Agent can sync any changes that may have occurred in space (WEBCLIENT-454).

  A new auto-task, **Synchronise Containers**, was created to perform this operation.

  The settings: `ContainerRunLimit`, `EnableSyncContainers` and `SyncInboxesOnly`, have been added to control the behaviour of the task.

  See container_syncing for more details..
- Addition template variables for `SandboxCommand`:
  - {RLE=T} Set to non-zero value if the option "require-local-encryption=true" should be set when starting the agent.
  - {IST=N} Set to non-zero if the option "idle-shutdown-timeout" should be set to the given value.
  - {ESE=F} Set to non-zero if the option "enable-shell-extension=false" should be set.
- Prevent the pre-5.0 TeamDrive Agent from starting the shell extension (WEBCLIENT-453).
- If the Web Portal cannot reach the Agent (HTTP connection failed), it will terminate the process (if it is running) and return a session timeout error to the Web GUI (WEBCLIENT-450).
- Hardening sets UMASK to 077 which requires group privileges to be fixed when the TeamDrive Agent package is unzipped (WEBCLIENT-452).
- Login on the Web Portal no longer returns the "Unknown user" error (WEBCLIENT-438). Instead, it will return an error of the form: "Username or password incorrect", when the user enters their password.
- The Admin Console now displays the Auto Task list (WEBCLIENT-434).
- The mod_agent.log can now be viewed in the Admin Console.
- Added a new Sandbox setting: `RequireLocalEncryption`, which allows you to ensure that all containers of the Web Portal use local encryption (WEBCLIENT-399).
- When `UseEmbeddedLogin` is set to `True`, the "embedded" option is no longer set for the `RegistrationURL` link (WEBCLIENT-379). This is because the Web Portal always redirects to this page, and does not embed the page in the Web user interface.
- Improvement to Web Portal redirection:

  The Web Portal will display an information message when the user has been redirected from another Web Portal.

  After a redirect has occurred, the login name (username or email) entered by the user will preserved and display in the appropriate field (WEBCLIENT-363).

If the login email contains a registered domain, then the Web Portal will redirect to the Web Portal belonging to the Provider of the domain, even if the user is not yet a registered user (WEBCLIENT-375).

When multiple Web Portals are used by the same Registration Server, the user will now be redirected to their Provider associated Web Portal, when attempting to login to the incorrect Web Portal. Previously users were only redirected when the required Web Portal was associated with a different Registration Server.

NOTE: The Web Portal associated with Provider is specified by the `WEBPORTAL_API_URL` Provider setting. This value must be set if redirection is required.

CHAPTER

FIFTEEN

RELEASE NOTES - VERSION 3.0

15.1 3.0.4 (2022-09-21)

- Set client version to 4.8.0.3223

15.2 3.0.3 (2022-06-15)

- Set client version to 4.7.5.3196

15.3 3.0.2 (2022-01-10)

This release also includes a number of security improvements. Please follow the instructions in upgrade_to_dockerless to upgrade an existing Web Portal to a docker-less version. Please contact TeamDrive for further details.

- For security reasons, Docker has been replaced by customised TeamDrive Agent containerisation (WEBCLIENT-430).

  The settings `ImageBuildFolder`, `MinDockerDataSpaceAvailable`, `MinDockerMetaDataSpaceAvailable` `RootlessDocker`, `BuildDockerfile`, `ImageBuildCommand`, `DockerEntryPoint`, `BuildWgetCommand`, `ContainerHosts`, `ContainerUserID`, `ContainerGroupID`, `RunAsUser`, `RunAsGroup` and `UseSudo` are no longer used and have been removed.

  Renamed `DockerHost` setting to `ContainerHost`.

- Added a new apache module: mod_agent which is now responsible for routing calls from the browser to the respective TeamDrive Agent.
- Added the `SandboxCommand` setting which specifies the command for the TeamDrive Agent sandbox. If empty, then the agent is not run in a sandbox.

  The following template variables may be used in the setting:

  - {TDBIN} TeamDrive Agent binary, this value should be: "/var/teamdrive/webportal/agent/teamdrived.bin"
  - {APIPORT} API port number
  - {WSPORT} Websocket port
  - {USERNAME} The username of the TeamDrive user
  - {ROOTPATH} TeamDrive root path, this value should be "/teamdrive/" the Agent directories used un this path are "{ROOTPATH}{USERNAME}/system" and ""{ROOTPATH}{USERNAME}/spaces"

  - {DBSPATH} The alternative database path, which is used to store the SQLite database files. If there is no alternative path then {DBSPATH} == {ROOTPATH). The actual directory used by the agent is: "{ROOTPATH}{USERNAME}/system"
  - {INIPATH} The shared directory which contains the "teamdrive.ini" file.

15.4 3.0.1 (2021-10-11)

This is a security update.

- A number of security issue 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).
- Fixed container creation error after user was deleted and recreated (WEBCLIENT-418 and WEBCLIENT-419).

15.5 3.0.0 (2021-08-20)

The 3.0 release includes a several security bug fixes and a number of hardening measures, and is recommended to all users.

Please contact TeamDrive for further details.

Version 3.0 is an in-place upgrade to all previous versions running on CentOS 7.

On CentOS 8 the new version runs with Docker in "rootless mode", see:

https://docs.docker.com/engine/security/rootless/

Because of the added security due to rootless mode, and other CentOS 8 security updates, all users of the Web Portal are requested to transition to this version as soon as possible.

- Initial public release of 3.0.
- Set security headers in Apache configuration (WEBCLIENT-400).
- OS hardening and security update to Apache configuration (WEBCLIENT-385).
- Hardening of TeamDrive Agent (Agent Version >= 4.7.1.3011).
- Support for running Docker in rootless mode (only CentOS 8)

CHAPTER

SIXTEEN

RELEASE NOTES - VERSION 2.0

16.1 2.0.8 (2020-05-10)

- Fixed an access denied error when calling the Registration Server API to get information on a user that belongs to a another provider (i.e. a provider other than the Web Portal's provider).
- Fixed handling of email address change due to user deletion or if two users switch email addresses (WEBCLIENT-372).
- Added support for MySQL 8
- Set client version to 4.7.0.2944

16.2 2.0.7 (2020-12-16)

- If a user logs in with an email address that is not unique, the Web Portal will return an appropriate error (WEBCLIENT-358).
- Login with email will now re-direct to the correct Web Portal if necessary, provider Registration Server version 4.5.4 or later and TDNS version 2.0.2 is use (WEBCLIENT-357).
- Set client version to 4.6.12.2793

16.3 2.0.6 (2020-10-02)

- Login with a temporary password was not working when using an email address (WEBCLIENT-356).
- Fixed bug: the Web GUI not going directly to the external authentication login page when `AuthServiceEnabled` was set to `True` (WEBCLIENT-355).
- Entries separated by a newline in the `ContainerHosts` setting was not working correctly (WEBCLIENT-354).
- Fixed "Array index out of bounds" error when accessing the "Build Image" settings details page.

16.4 2.0.5 (2020-09-15)

- The "White Label" settings have been renamed to "Build Image" settings. In addition, the setting `UseWhiteLabeldDockerImage` has been removed and the `WhiteLabelINIFileSettings` setting has been rename to `ClientSettings` (see below).

  `UseWhiteLabeldDockerImage` is no longer required because all Web Portals now use the image build settings to create a new Docker image on upgrade, if necessary.

The setting `WhiteLabelIdleTimeout` has been renamed to `ContainerIdleTimeout` and is now a "Docker Setting" (see containeridletimeout).

The `IdleContainerTimeout` setting has been renamed to `RemoveIdleContainerTime` to better distinguish this value from `ContainerIdleTimeout`.

- Added `SharedIniPath` and `AgentCommandLineArgs`. Using `SharedIniPath` you can specify a global path for the "teamdrive.ini" file (WEBCLIENT-350).

  `WhiteLabelINIFileSettings` has be rename to `ClientSettings` and is now a "General Setting". Client settings that are set using the `ClientSettings` setting are then written to the `teamdrive.ini` file. If a `SharedIniPath` is specified, then they are read by the all TeamDrive agents, when a container starts. If not, then the client settings are written to the `/etc/teamdrive.ini` file, which is part of the container image.

  The `AgentCommandLineArgs` settings is a read-only variable that specifies the command line arguments that are passed to the TeamDrive agent when the container starts.

  See sharedinipath, agentcommandlineargs and clientsettings for details.

- Added `MaxLoginRate` and `MaxLoginLogAge` settings. These settings are used to detect Denial of Service and other brute force attacks targeting the Web Portal login (WEBCLIENT-344). See maxloginrate and maxloginlogage for details.
- Error messages returned by the Web Portal are now use the translation file provided by the TeamDrive Agent.
- Added `ContainerHosts` setting (see containerhosts). Use this to specify entries for the "/etc/hosts" file of the container (WEBCLIENT-139).
- You can now configure a proxy during setup of the Web Portal (WEBCLIENT-338).
- If `AuthServiceEnabled` is `False` the Web Portal now uses external authentication as required by the user, provided you are using TeamDrive Agent 4.6.11.2656 or later (WEBCLIENT-335).

  As before, if `AuthServiceEnabled` is `True`, then Web Portal uses a specific authentication service (as specified by `AuthLoginPageURL` and `AuthTokenVerifyURL`).

  See authserviceenabled for more details.

- Moved settings `SessionTimeout`, `ForceHTTPSUsage` and `ForceHTTPSUsage` to `Admin Console` settings group.

  Moved `RegistrationEnabled` and `RegistrationURL` to the `Authentication` settings group.

- The Web Portal will now redirect to another Web Portal, if a user attempts to login to the incorrect Web Portal (WEBCLIENT-333). This is done if the provider of the user is not in the list of `AllowedProviders`.

  On the Registration Server of the user, you must set the `WEBPORTAL_API_URL` provider setting. This setting specifies the domain name of the Web Portal used by the provider. In addition, Registration Server version 4.5.4 is required. This version implements the "webportal" redirect required to implement this functionality.

  If any of these conditions is not met, then the user will get the error message: "The provider you are registered to is not enabled for this web portal".

- Set the minimum client Agent version to 4.6.11.2707. This version support the Web Portal redirect, and includes some error message improvements.
- Setting the default distributor code, and language using the `portal/login.html` and `extauth/login.html` pages is not longer supported.

16.5 2.0.4 (2020-05-19)

- Added Multi-Registration Server support.

- Fixed agent download URL.
- All documents and security relavent data stored in containers run by the web portal are now encrypted when using TeamDrive Agent version 4.7 or later.

  Encryption activates the so-called "super PIN" functionality implemented by Registration Server 4.2. When the super PIN is activated for an account the user is required to print out and save a 56-digit super PIN, and recovery URL (in the form of QR code) in a secure place.

  After activation of the super PIN functionality the user can only access their account using their password, or the super PIN, or the recovery code (which can be retrieved using the recovery URL). Changing your password is also only possible using either the super PIN or recovery code.
- Changes made to support local encryption of inboxes. Encryption of inboxes required Registration Server version 4.2 or later, and TeamDrive Agent version 4.7 or later.
- Added `ContainerDatabases` setting (WEBCLIENT-334). This setting allows you to specify an alternative path for the SQLite databases used by the containers. Normally all data is placed in the `ContainerRoot` directory.

  When specified the new location will be mounted in the container under the path: "/teamdrive/dbs". However, this path will only be used if you build a new image using the TeamDrive Agent version 4.6.12.2637 or later.

  This version of the client supports the "–database-path" option which allows you to specify an alternative path for the SQLite database. When `ContainerDatabases` is set, the image build process will automatically add this option to the start parameters of the agent (see `@USEDATABASEPATH` in the `WhiteLabelDockerfile` setting).

16.6 2.0.3 (2020-04-14)

- Changes for yvva 1.5.2 compatibility.
- Fixed a problem removing container data, remove directory was failing when a '$' was in the path name.
- The Web Portal will now correctly use the database specified in the "td-webportal.my.cnf" file (WEBCLIENT-296). Previously the database name was hard-coded to "webportal".
- Fixed: in case of an exception the temporary file created by syscall() is not be deleted (WEBCLIENT-316).
- Fixed: HTML entities conversion problem when editing setting "WhiteLabelDockerfile" (WEBCLIENT-323).
- When the docker image in being updated, the Web GUI will now return the error "Upgrade in progress, please try again shortly", when the user attempts to login.
- Added API functions to enable and disabled a container (WEBCLIENT-324).
- Added support for "prelogin" call in order to support login changes (WEBCLIENT-327).
- Added "sqlite-synchronous=normal" as start parameter for the agents to reduce SQLite flush frequency
- Set client version to 4.6.10.2619

16.7 2.0.2 (2019-07-26)

- Increased MinimumAgentVersion to 4.6.7.2355.

16.8 2.0.1 (2019-06-11)

- Fixed problems the on demand creation and starting of containers that have been deleted (WEBCLIENT-304).

16.9 2.0.0 (2019-04-25)

Note: Please follow the new update process described in chapter upgrade_web_portal. The former separate GUI rpm package is not longer necessary. The standard Web Portal will update the docker Container image from the docker hub during the update step and will extract and update the files necessary for the GUI from this image. A white label Web Portal needs the white label agent .tar.gz to build a white label docker container image.

- Initial release of Web Portal 2.0.

16.9.1 Upgrading from previous versions of the Web Portal

As of version 2.0.4 you must run the `upgrade_now` command from the console after installing a new version of the Web Portal.

This command updates the database structure and the docker image used by the Web Portal. The Admin Console may return errors, and other random errors may occur before the upgrade had been completed.

To update the database structure and docker image start `yvva` and execute `upgrade_now;;`. This command also upgrade the container image used by the Web Portal. See the chapter upgrade_web_portal for details.

16.9.2 Key features and changes

- Increased MinimumAgentVersion to 4.6.7.2328
- External authentication supports both login and registration. This feature can be activated by setting `AuthServiceEnabled` to True. To allow registration set `RegistrationEnabled` to True. If no `AuthLoginPageURL` or `RegistrationURL` page is specified then the Web Portal will use the "portal pages", provided by the Registration Server.
- External authentication can be embedded in the TeamDrive Web GUI, or can the external authentication pages can be used directly. A new setting: `UseEmbeddedLogin`, must be set to `True` in order to use the embedded login form.

  By default, `UseEmbeddedLogin` is set to `False` if you upgrade from a previous version of the Web Portal that was using external authentication. Otherwise, the default is `True`. This is to ensure backwards compatibility, with previous versions that only supported the non-embedded form.

  Accessing the Web Portal domain, for example: `https://webportal.yourdomain.com`, will automatically present the login in the embedded or non-embedded form, as specified by `UseEmbeddedLogin`.
- You can now use "explicit" links to the login page in order to set the default provider code and language, for the login or registration.

  For the non-embedded login form use the following explicit link:

  ```
  https://webportal.yourdomain.com/portal/login.html?dist=CODE&lang=LG
  ```

  and for the embedded login form use the following explicit link:

  ```
  https://webportal.yourdomain.com/extauth/login.html?dist=CODE&lang=LG
  ```

where `CODE` is the provider code, and `LG` is the language code, for example `en` or `de`.

Note that the external authentication service must be able to handle the specified provider code and language.

16.9.3 Administration Console

- Added a Container list page, which can be used to search for containers of a particular user and type. The container details page allows you to stop, start and delete containers.

  Note that deleting a container will remove all the container data as well. This means that Web Portal users will find all spaces deactivated on next login. If the user looses his password he will also loose access to his data, unless he has a TeamDrive installation elsewhere.

CHAPTER

SEVENTEEN

RELEASE NOTES - VERSION 1.2

17.1 1.2.3 (2019-01-15)

- Reset of Admin User's password as described in the documentation (i.e. by setting the password to blank in the database) was not working (WEBCLIENT-259).
- Added a illustrated overview of the Web Portal to the documentation, showing the connection to other components in the TeamDrive system (see *Introduction to the TeamDrive Web Portal* (page 9)).

17.2 1.2.2 (2018-11-06)

- The Web Portal supports now the Docker Community and Enterprise Edition and also still the old Commercially Supported version with the latest version 1.13 (from January 2017). Please notice, that the Docker CS version will be still maintained, but not further developed any more. Check the docker installation chapter for the differences between Docker CS and CE/EE installation.
- The Web Portal will only allow using signed DISTRIBUTOR files like the standard client. The signature will be checked during the creation of the docker image and at each start of the agent. Additional client settings must be moved to the new setting `WhiteLabelINIFileSettings`. If settings are still required in the DISTRIBUTOR file it must be signed by TeamDrive Systems for you.
- The current agent supports now web-sockets to refresh data in the browser without refreshing the page itself. To support web-socket connections, the apache module `proxy_wstunnel_module` must be enabled (See *Apache 2.4* (page 20) for details)
- Increased MinimumAgentVersion to 4.6.4.2183

17.3 1.2.1 (2017-11-29)

- Increased MinimumAgentVersion to 4.5.5.1838
- Upgrade will change the `WhiteLabelAgentDownloadURL` setting from ".../{PRODUCTNAME}_agent_{VERSION}_x86_64.tar.gz" to ".../{PRODUCTNAME}_agent_{VERSION}_el7.x86_64.tar.gz". this is done because the TeamDrive agent is now built in 2 versions: "el6" are built for CentOS 6, and "el7" versions are built for CentOS 7. It is assumed that the Web Portal is run on a CentOS 7 platform. If this is not the case, then you must manually change this setting to ".../{PRODUCTNAME}_agent_{VERSION}_el6.x86_64.tar.gz" (WEBCLIENT-255).
- Updated documentation to include new TeamDrive CI (WEBCLIENT-254).
- The Web Portal external authentication now handles transitioning to a new User Secret generation algorithm as implemented by Registration Server version 3.7.6.
- Bug fix: boolean settings were not correctly pre-selected.

- Several improvements have been made to the upgrade procedure which generates a new Docker image. The setting `WhiteLabelAgentDownloadURL` can now be left blank, of the Agent archive (.tar.gz file) has been placed manually in the build folder (`WhiteLabelDockerBuildFolder`).

  If `ContainerImage` is set to image with a version number higher than the `MinimumAgentVersion`, then the Web Portal will build an image for the version specified by `ContainerImage`.

- Version 1.2.1 requires YVVA runtime version 1.4.4.

17.4 1.2.0 (2017-08-14)

Note: Please follow the new update process described in chapter upgrade_web_portal. The former separate GUI rpm package is not longer necessary. The standard Web Portal will update the docker Container image from the docker hub during the update step and will extract and update the files necessary for the GUI from this image. A white label Web Portal needs the white label agent .tar.gz to build a white label docker container image.

- Initial 1.2 release.

17.4.1 Key features and changes

- Simplified installing and updating the web portal and docker container for standard and white label configuration.
- Increased MinimumAgentVersion to 4.5.2.1775 to support PointInTime-Recovery and Read-Confirmations

CHAPTER

EIGHTEEN

RELEASE NOTES - VERSION 1.1

18.1 1.1.0 (2017-04-10)

Note: When updating from an older version of the Web Portal, remove the DOCKER_HOST setting in the apache config file `/etc/sysconfig/httpd`. It is not longer necessary.

If you update docker to version 1.12.6 the docker service might not start anymore as described in the docker release notes: https://github.com/docker/docker/releases/tag/v1.12.6 Please remove the file `/etc/systemd/system/docker.service.d/web-portal.conf` and add the `--host=tcp://0.0.0.0:2375` instead to the OPTIONS parameters in `/etc/sysconfig/docker` as described in the docker configuration chapter.

- Initial 1.1 release.

18.1.1 Key features and changes

- Added professional license required check (WEBCLIENT-233)
- Added setting to limit currently active users (WEBCLIENT-234)
- Added setting for minimum docker available data and meta data space. If minimum is reached, no more docker container will be created for new users (WEBCLIENT-235)
- Settings are now displayed in groups in the Admin Console (WEBCLIENT-237).
- Increased MinimumAgentVersion to 4.3.2.1681 to support space web access settings (TDCLIENT-2184). The webportal docker agent will be started with an additional setting agent-type=webportal to distinguish a standard and a webportal agent
- Added settings to support a Proxy for outgoing connections: `UseProxy`, `ProxyHost` and `NoProxyList` (WEBCLIENT-242). See outgoing_connections for details.
- Added the `ConnectionTimeout` setting which specifies a timeout for outgoing connections (see outgoing_connections).
- Added support for Docker Swarm. Docker Swarm is a native clustering for Docker. It turns a pool of Docker hosts into a single, virtual Docker host. Please notice, that only the legacy standalone Swarm is supported (https://docs.docker.com/swarm/overview/), because of the different service model in the Docker Engine v1.12.0 using the swarm mode. Change the `DockerHost` Web Portal setting from the standard docker port 2375 to the swarm port 2377 to switch from the standard docker API access to the swarm API access (WEBCLIENT-245).

CHAPTER

NINETEEN

RELEASE NOTES - VERSION 1.0

19.1 1.0.9 (2017-02-10)

- Increased MinimumAgentVersion to 4.3.1.1656 to fix a bug when login with email address and magic usernames.
- Revised chapter Web Portal Virtual Appliance with CentOS 7 and docker direct-lvm storage

19.2 1.0.8 (2017-02-07)

Note: After updating docker to version 1.12.6 the docker service might not start anymore as described in the docker release notes: https://github.com/docker/docker/releases/tag/v1.12.6 Please remove the file `/etc/systemd/system/docker.service.d/web-portal.conf` and add the `--host=tcp://0.0.0.0:2375` instead to the OPTIONS parameters in `/etc/sysconfig/docker` as described in the docker configuration chapter.

- Removed support for CentOS 6
- Fixed docker configuration
- Fixed PDF creation for this documentation
- Fixed download links for VM-Ware images

19.3 1.0.7 (2016-11-10)

- Increased MinimumAgentVersion to 4.2.2.1579 to support email notifications
- Fixed docker configuration
- Fixed apache 2.4 configuration

19.4 1.0.6 (2016-07-11)

Note: Updating the Host 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 *Apache 2.4* (page 20)

- Improved Docker installation documentation (WEBCLIENT-219, WEBCLIENT-223).

- The Web Portal now checks if the user is authorised to access a Web Portal. A user is authorised to access a Web Portal if the Provider setting: `ALLOW_WEB_PORTAL_ACCESS` is set to `permit` or `ALLOW_WEB_PORTAL_ACCESS` is set to `peruser` and the user's "Web Portal Access" capability bit is set (a user-level setting).

  When using external authentication, the same check is done if the Registration Server is version 3.6 or later. When using a Registration Server 3.5 or earlier, the Web Portal will not check the user's Web Portal access permissions (in the case of external authentication).

- Added setting `AllowedProviders` which is a list of Provider codes of the users that are allowed to login to the Web Portal.

  An input field on the setup page allows this variable to set during installation of the Web Portal.

- The URL `https://webportal.yourdomain.com/portal/authservice.html` is now the target URL for external Authentication Services acting on behalf of the Web Portal.

  In other words, in successful authorisation by an external Authentication Service, the user is redirected back to this page.

  The Web Portal will may add certain arguments to `AuthLoginPageURL` and `RegisterURL` pages:

  - "portal=true": This argument is always added to the URL. This is useful, in the case when the same Authentication Service is called by the TeamDrive Client and the Web Portal. The argument can be used to determine whether to redirect on successful login or not.
  - "cookie=?": This argument will be added if the Authentication Service provided a cookie after the last successful login. The cookie is stored by the TeamDrive Agent.
  - "error=?": This argument indicates that the Web Portal encountered an error after successful authorisation by the Authentication Service. It is a base-64 (URL) encoded string containing the error message. The error should be displayed in the login page served by the Authentication Service.

- Support CentOS 7 with Apache 2.4
- Increased MinimumAgentVersion to 4.2.0.1470 to support the space activities
- Added setting `RegistrationEnabled` (default `False`). This value must be set to `True` to allow registration of users directly via the Web Portal.
- Added login and registration pages: All of these pages redirect to the associated pages on the Registration Server. After login, or registration, the Registration Server redirects back to the Web Portal.
  - `https://webportal.yourdomain.com/portal/login.html` This page allows users to login using two-factor authentication, if this has been configured. `/portal/login.html` is now the default for the `AuthLoginPageURL` setting.
  - `https://webportal.yourdomain.com/portal/register.html` Using this page a user can register as a TeamDrive user without installing the TeamDrive Client. After registration the user has access to the Web Portal. `/portal/register.html` is now the default for the `RegisterURL` setting.
  - `https://webportal.yourdomain.com/portal/lost_pwd.html` This page sends a temporary password to the user and allows the user to login and set a new password. The page is linked from `/portal/login.html`.
  - `https://webportal.yourdomain.com/portal/setup-2fa.html` Using this page the user can configure two-factor authentication using the Google Authenticator App.
  - *The default of the ''AuthTokenVerifyURL'* setting is now: `https://webportal.yourdomain.com/portal/ve`

19.5 1.0.5 (2016-02-16)

- Fixed a problem on login with a user registered via the Registration Server API using email address as identification (WEBCLIENT-205).

- Use the -v option when removing containers. This ensures that the container volume is also removed (WEBCLIENT-204).

19.6 1.0.4 (2016-02-09)

- Framework synced with Host- and Reg-Server

19.7 1.0.3 (2016-02-02)

- Added setting `MinimumAgentVersion` which specifies the minimum version of the TeamDrive Agent that will work with the Web Portal. Upgrade to this version of the Agent is forced as soon as the new version of the Web Portal is online (WEBCLIENT-194).
- Updated documentation for Docker version 1.7.1
- Fixed Internet explorer caches API calls. (WEBCLIENT-186)
- Added description about the dependencies between Webportal, Provider and Reg-Server and normal and external Authentication. (WEBCLIENT-176)
- The performExternalAuthentication redirects to http:// instead of https://. (WEBCLIENT-182)
- The getLoginInformation() API call now returns "registerUrl" if the setting RegistrationURL, is set on the Web Portal. (WEBCLIENT-179)
- Redirect to the login page when a request to an agent returns a 503 code. This requires a manual update to the ssl.conf, refer to the documentation on server installation and configuration. (WEBCLIENT-198)

19.8 1.0.2 (2015-12-07)

- Fixed container language settings so that Spaces with non-ascii characters in the name now work.
- Corrected redirect to external login pages under certain circumstances.
- Login with an email address now works.
- The Portal no longer creates containers based on the case of the input username, instead the actual username is used. This prevents the creation of duplicate containers for the same user.
- The Web Portal session will now timeout after 15 minutes idle time. The user is then required to login again.
- Implemented reset password functionality. Login after password has been forgotten now works. The user will receive a temporary password via email which is used to set a new password and login.
- Note, new re-write must be added to `/etc/httpd/conf.d/ssl.conf`:

```
RewriteRule ^/requestResetPassword /yvva/requestResetPassword [PT]
RewriteRule ^/tempPasswordLogin /yvva/tempPasswordLogin [PT]
```

- Fixed loading of favicon

19.9 1.0.1 (2015-10-27)

- OldImageRemovalTime setting was not visible.
- Updated Web Portal GUI to the latest 4.1.x version from the webfrontend branch.

19.10 1.0.0 (2015-10-08)

- Initial public release of the Web Portal.
- Web Portal 1.0 requires TeamDrive Agent version 4.0.12.1292 or later.

CHAPTER

TWENTY

APPENDIX

20.1 Abbreviations

PBT **PBT** is an object oriented language specifically designed for the programming of "server-side" functionality common to intra- and internet Web sites. A large share of the TeamDrive Host, Registration Server and Webportal Server functionality is implemented in PBT. The code is parsed and executed by the Yvva application server components.

TDNS **T**eam **D**rive **N**ame **S**ervice

TDRS **T**eam **D**rive **R**egistration **S**erver

TSHS **T**eam **D**rive **S**calable **H**ost **S**torage.

CHAPTER

TWENTYONE

DOCUMENT HISTORY

Date	Version	Name	Description
2015-08-07	1.0	Paul McCullagh	Start