



Hands-on Lab

Enterprise reporting, distribution, and governed AI - product documentation

Version 6.0.0.0

Contents

Hands-on Lab

About this lab	4
Lab 1: Install Reportworq on Windows	7
Lab 2: First-run setup and licensing	17

Hands-on Lab

About this lab

explanation

administrator

report author

This is a **guided course**, not a reference. You start with an empty Windows server, install Reportworq, and finish with a working reporting system that emails a board pack to the people who need it. Every lab is a task you actually perform, in order, with the reason for each step alongside the step itself.

The worked example throughout is **Timberline Harvest Co.**, Reportworq's own demo company. Its workbooks, planning model, and reports are the material you build with, so the reports you produce are the reports a real finance team would recognize, not toy examples. See [the Timberline Demo System guide](#) for what the company is and what is in it.

Who this is for

Anyone standing Reportworq up for the first time, on their own server or a trial one: the administrator who installs it, and the report author who then builds on it. The labs assume you know Excel well and your planning system reasonably well. They assume nothing about Reportworq.

If you already have a running server and only want to look at what Reportworq can do, the Timberline guide is the faster route. Come here when you want to build it yourself and understand why each piece is set the way it is.

How every lab reads

The labs share one shape, so you always know where you are:

Heading	What it gives you
What you are doing	The task in business terms, the way somebody would ask for it
What this shows you	The Reportworq capability the lab is really about
Before you start	Prerequisites, and what an earlier lab must have left in place
Part A: the source file	What is in the workbook and why it is built that way
Part B: in Reportworq	The job itself, step by step, with the exact control labels
Part C: what you got	Where the output lands and how to read it
Why it works this way	The design decision behind the step, and the trap people fall into
Try this	One deliberate change, so you prove you understood it
Going deeper	Links into the other guides, for the full detail

Install and setup labs use Parts A to C for the stages of the install rather than for a workbook, but the rest of the shape holds.

The labs

Work through them in order. Each one assumes the one before it.

Lab	What you build	Status
Lab 1: Install Reportworq on Windows	A running server and the service account your jobs will use	Available
Lab 2: First-run setup and licensing	The repository, the Timberline demo content, your license, and your first sign-in	Available
Lab 3: Connect your data and your source files	A report provider and a data-source connection	Planned
Lab 4: Your first report, one output	The Financial Statements pack: one workbook, one parameter, one output per market	Planned
Lab 5: One deck, a section per division	The board pack: Excel to PowerPoint, and a page per item	Planned
Lab 6: A file per division	The divisional pack: cascading parameters, and one output per item	Planned
Lab 7: An email people actually read	The dashboard email body, with a chart and a live table	Planned
Lab 8: Depth and AI commentary	The sales pack: several parameters, generated tabs, and AI-written commentary	Planned
Lab 9: Put it on a schedule	Scheduling, holds, and watching a run in Activity	Planned

How this fits with the other guides

The labs are deliberately narrow. They walk one correct path and explain it. When you need the full picture, the other guides carry it:

- **Administrator Guide** is the complete install reference, including Linux, containers, and every option the labs skip.
- **User Guide** is the reference for reports, parameters, bursting, distribution and scheduling.
- **Timberline Demo System** describes the finished reports the labs build toward.

Going deeper. Each lab ends with its own links into those guides, at the exact point where you have just done the thing they describe.

Lab 1: Install Reportworq on Windows

LAB-01

tutorial

administrator

What you are doing. Taking a Windows server from nothing to a running Reportworq 6 installation, and setting the one thing that is far easier to get right now than to discover later: the Windows account the service runs as.

What this shows you. How Reportworq is delivered and what it actually installs. Reportworq is one service that hosts both the web application and the job runtime, so there is no separate scheduler, no separate engine, and nothing to install on anyone's desktop. By the end of this lab the server is ready for the first-run wizard in Lab 2.

How long. About 20 minutes, most of it the file copy.

What you need. A Windows server you have administrator rights on, and a valid Reportworq 6 license key. The key is not optional here: it unlocks the download itself, not just the software.

Before you start

Check the server against these before you download anything. Getting an install most of the way through and then finding the box is too small is a slow way to learn it.

Check	What it needs
Operating system	Windows Server 2016 or later. Earlier versions are not supported.
CPU	4-core, 2 GHz or faster, x64
RAM	32 GB minimum. Allow about 2 GB per concurrent job on top of that.
Disk	10 GB free, and it grows with run history and logs
Browser	A Chromium browser, Chrome or Edge, with a window at least 1280 x 720
Firewall	TCP 8600 and 8601 open on the server
Internet	Outbound access for license activation, updates, hosted AI, and CloudHub

Three of those are worth a sentence of their own, because they are where installs go wrong:

- **Both ports, not one.** 8600 is the web server you and your users connect to. 8601 is the Real-time Event Hub, which is always the web port plus one. Local access on the server works the moment the service starts, but remote access stays blocked until somebody adds an inbound-allow firewall rule for the web port. Open only

8600 in a multi-server setup and it will appear to work while its real-time coordination fails quietly.

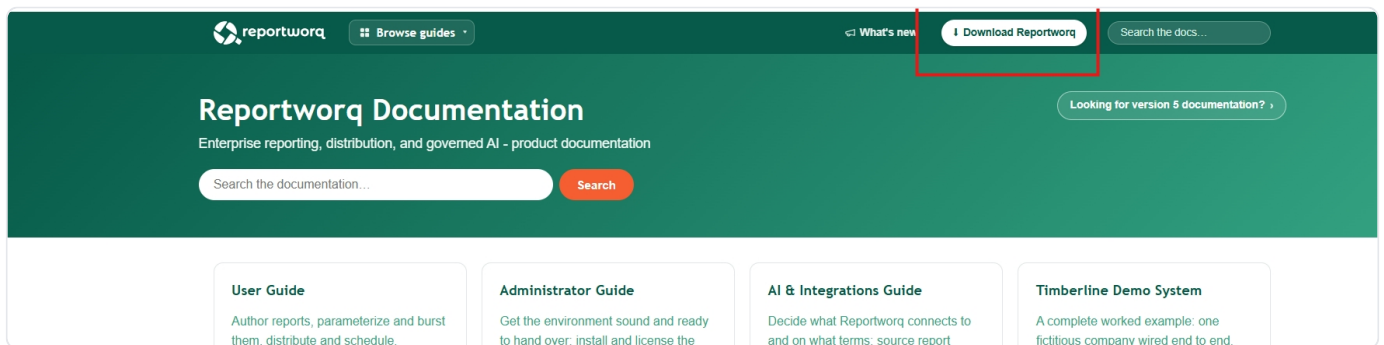
- **8600 is the Windows default.** If you install on Linux or in a container instead, the defaults are **8080** for the web server and **8081** for the Event Hub. It is the same product, so do not carry the Windows numbers across.
- **No outbound access at all?** Then plan for the offline activation path before you start, see [Install Reportworq](#).

The browser window figure is the size of the **window**, not the monitor. Display scaling makes the difference larger than people expect: a 1920 x 1080 screen at 150% scaling gives the browser 1280 pixels of width. See [System requirements](#) for the full list.

Part A: Get the installer

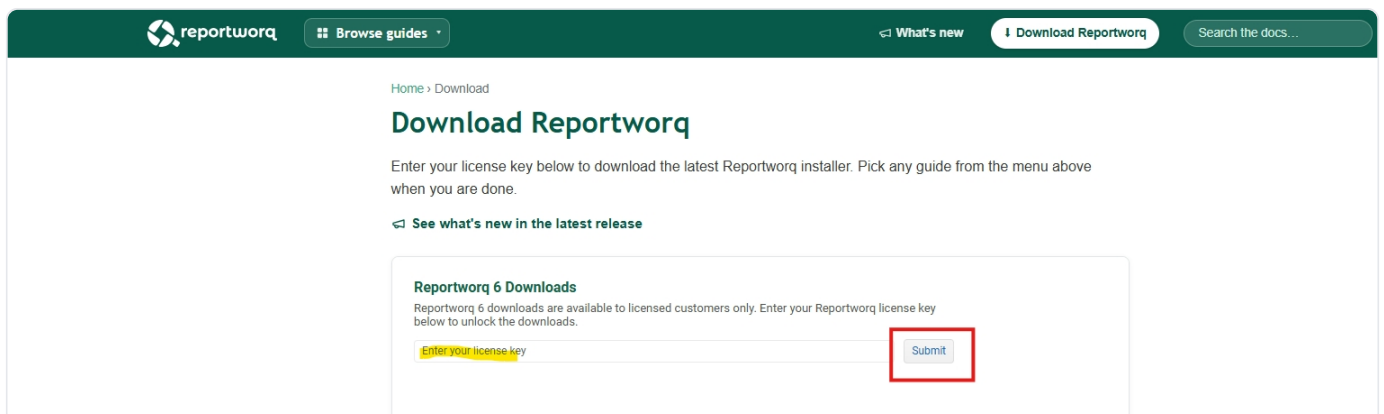
1. Go to the documentation site and start the download

Go to **docs.reportworq.com** and select **Download Reportworq** in the site header.



2. Enter your license key

Downloads are for licensed customers only, so the page asks for your **Reportworq 6 license key** before it shows you anything. Enter it and select **Submit**.



This is the same key you will activate the product with in Lab 2, so keep it to hand rather than closing the message it arrived in.

3. Choose the release and the artifact

The page now lists two releases and four files under each.

Current Reportworq 6 Official Release

OFFICIAL

Version: 6.0.0.2

Posted: 8/5/2026

The Reportworq Installer is necessary for new installations only. To upgrade an existing Reportworq instance, use the application's Administration screen. If the Reportworq server is offline, download the Reportworq Patch file from the link provided above and upload it to the Reportworq Administration Software Versions page.

Installer	ReportWORQ_Installer.exe	(487.38 MB)
Patch	ReportWORQ_Patch_6.0.0.2.zip	(447.76 MB)
Linux Package	reportworq-6.0.0.2-linux-x64.tar.gz	(462.59 MB)
Docker Container	reportworq-image-6.0.0.2.tar.gz	(534.65 MB)

Current Reportworq 6 Hotfix Release

HOTFIX

NEW

Version: 6.0.1.7

Posted: 8/31/2026

The Reportworq Hotfix release provides early access to bug fixes that are slated for the next official release. It is recommended to use the Official release unless otherwise instructed by the support desk.

Installer	ReportWORQ_Installer.exe	(491.24 MB)
Patch	ReportWORQ_Patch_6.0.1.7.zip	(451.64 MB)
Linux Package	reportworq-6.0.1.7-linux-x64.tar.gz	(465.90 MB)
Docker Container	reportworq-image-6.0.1.7.tar.gz	(543.20 MB)

Take the **Installer** from the **Current Reportworq 6 Official Release** panel. For this lab that is the Windows installer, `Reportworq_Installer.exe` . Downloads of 6.0.1.8 and earlier carry the older spelling `ReportWORQ_Installer.exe` , as the picture above shows; it is the same file.

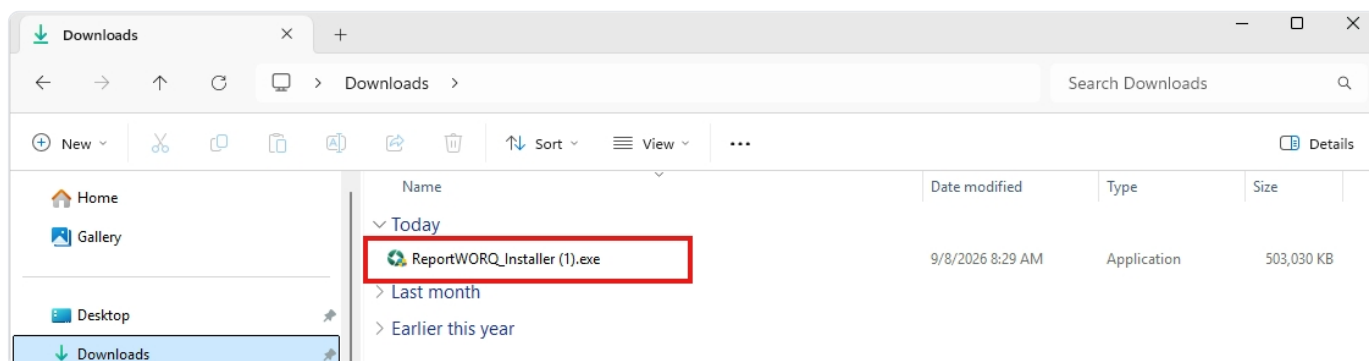
The other three rows matter later, so it is worth knowing what they are now:

File	When you use it
Installer	A new Windows installation. This lab.
Patch	Upgrading a Windows server that is already running Reportworq. You normally apply this from inside the product instead, under Settings > Update Reportworq , and only download it by hand when the server has no outbound access.
Linux Package	Installing on Linux, from a tarball rather than an installer.
Docker Container	Running as a container, on Docker or Azure App Service.

Take the **Official** release rather than the **Hotfix** unless the support desk has told you otherwise. The hotfix panel carries fixes that are scheduled for the next official release, which is useful when you are waiting on one of them and unnecessary when you are not.

4. Find the download

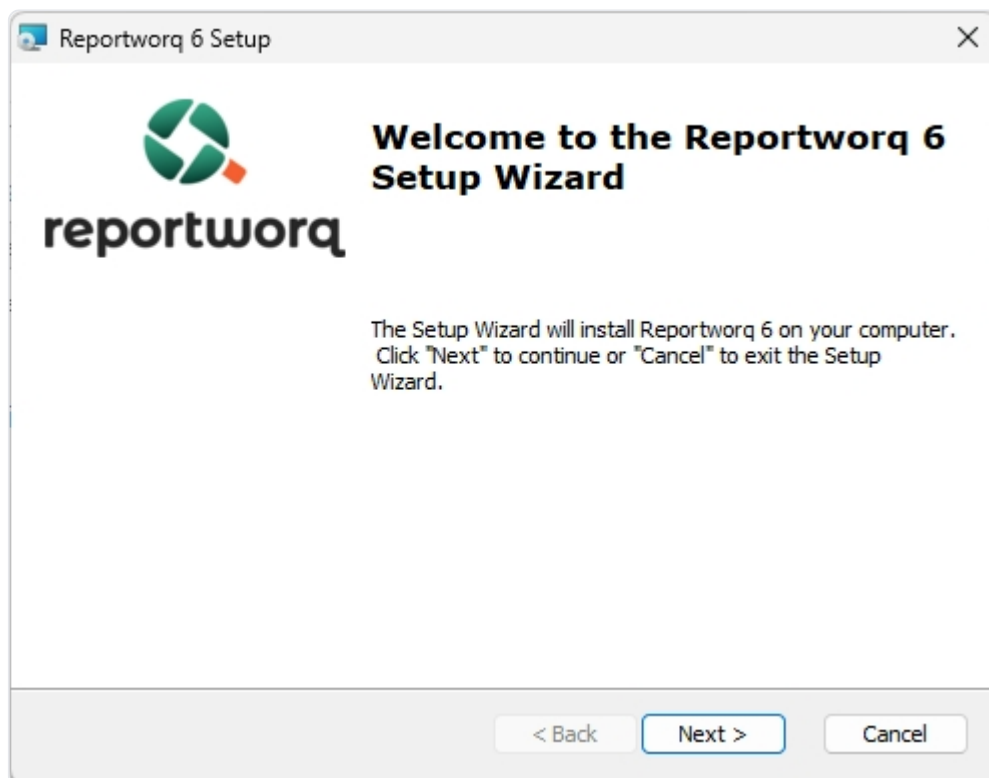
Open your **Downloads** folder in Windows Explorer.



Part B: Install Reportworq

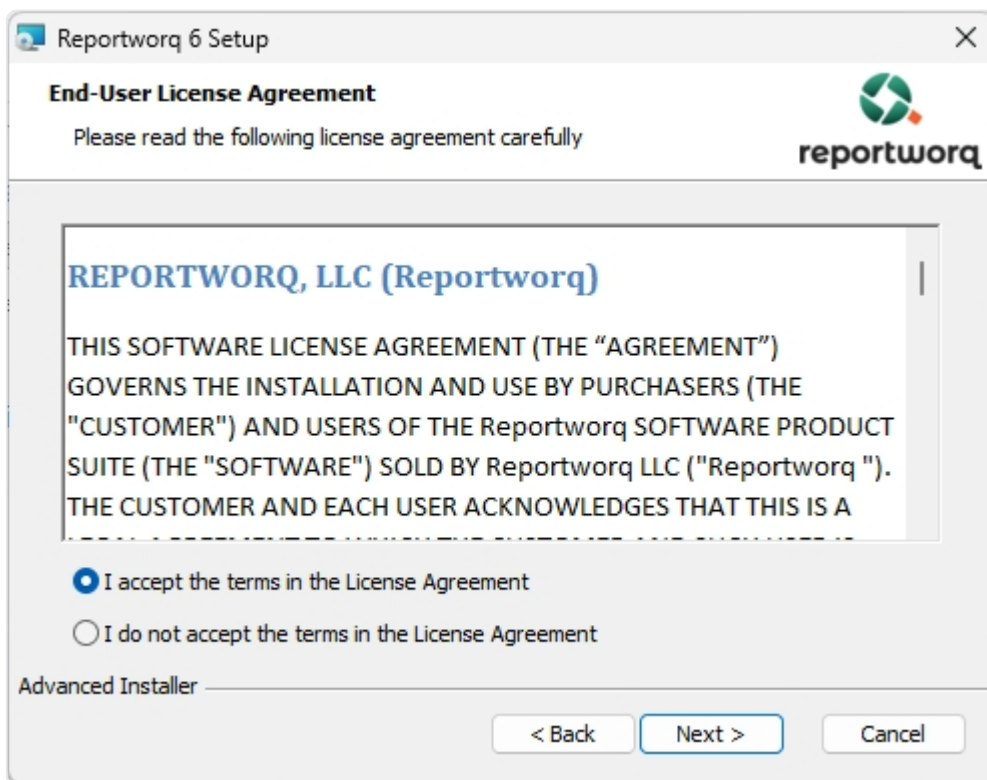
5. Run the installer

Double-click the installer to kick it off, then select **Next**.



6. Accept the license agreement

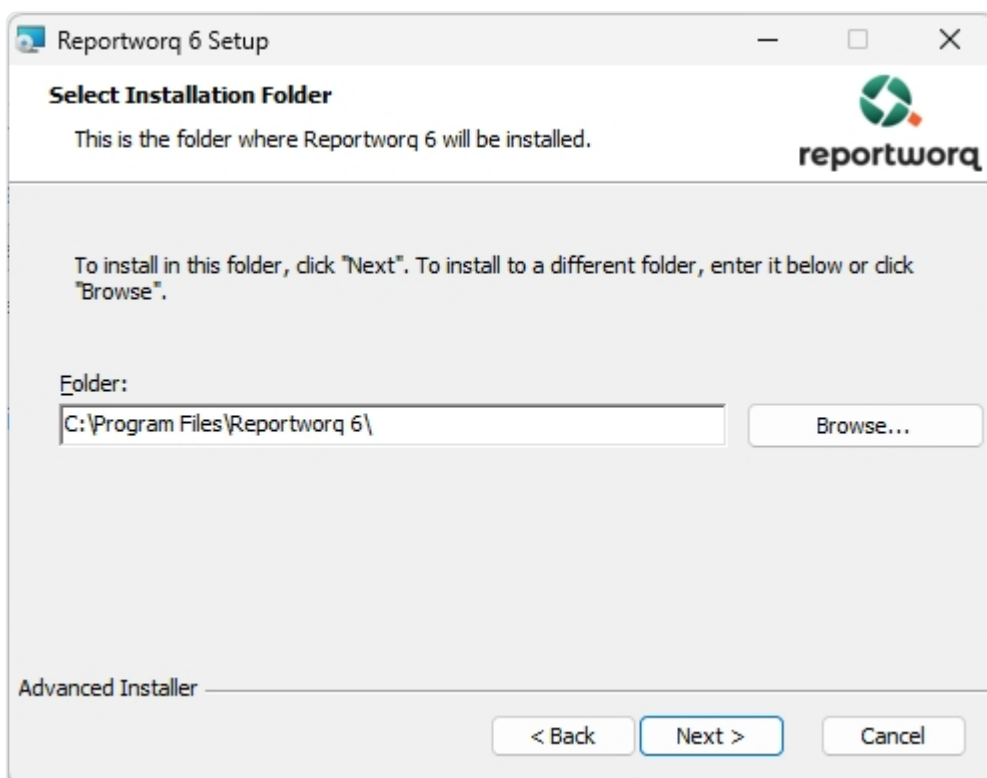
Select **I accept the terms in the license agreement**, then **Next**.



Acceptance is mandatory. The wizard will not move on without it.

7. Keep the default installation directory

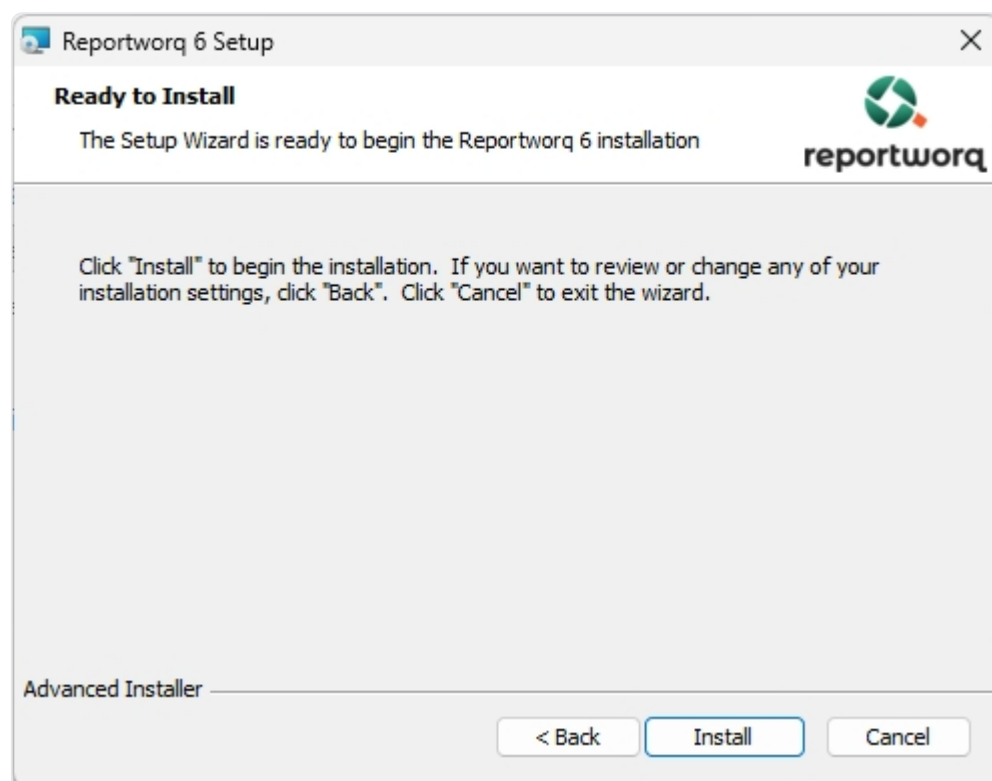
Select **Next** on the installation directory page. The default path is the one shown.



That default is doing real work. Reportworq 6 installs into its own folder, registers its own service, and listens on its own port, which is what lets it run **side by side with an existing Reportworq 5** on the same machine. That is the whole basis of a staged migration: stand version 6 up next to a running version 5, move content across, validate it, and decommission the old one only when you are satisfied. Change the path if your organization requires it, but change it knowing that is what you are stepping around.

8. Install

Select **Install** to begin.



This is the slow step. Nothing is asked of you until it finishes.

9. Finish

Select **Finish** when the installer is done.

Part C: Confirm the service, and set the account it runs as

10. Find the service

On the server, open **Services** from the Start menu, or run `services.msc`. Look for **Reportworq 6 Bootstrap Service**. It should be **Running**, with a startup type of **Automatic**, logging on as **Local System**.

Name	Description	Status	Startup Type	Log On As
Radio Management Service	Radio Management and Airplane Mode ...	Running	Manual	Local Service
Realtek Audio Universal Service	Realtek Audio Universal Service	Running	Automatic	Local Syste...
Recommended Troubleshooting Service	Enables automatic mitigation for known ...		Manual	Local Syste...
ReFS Dedup Service	ReFS data deduplication and compressio...		Manual	Local Syste...
Remote Access Auto Connection Manager	Creates a connection to a remote networ...		Manual	Local Syste...
Remote Access Connection Manager	Manages dial-up and virtual private netw...	Running	Manual	Local Syste...
Remote Desktop Configuration	Remote Desktop Configuration service (...)		Manual	Local Syste...
Remote Desktop Services	Allows users to connect interactively to a...		Manual	Network S...
Remote Desktop Services UserMode Port Redirector	Allows the redirection of Printers/Drives/...		Manual	Local Syste...
Remote Procedure Call (RPC)	The RPCSS service is the Service Control ...	Running	Automatic	Network S...
Remote Procedure Call (RPC) Locator	In Windows 2003 and earlier versions of ...		Manual	Network S...
Remote Registry	Enables remote users to modify registry s...	Disabled		Local Service
Reportworq 6 Bootstrap Service	This service is used to load and manage i...	Running	Automatic	Local Syste...
ReportWORQ Bootstrap Service	This service is used to load and manage i...	Running	Automatic	Local Syste...

This single service is the whole product: it hosts the web application and it runs the jobs.

Look at the row directly below it in the picture. That machine also runs the Reportworq 5 service, untouched and still running. That is the side-by-side install from step 7, as it appears in practice. If this is a clean server you will see only the version 6 row.

11. Decide whether Local System is enough

Local System cannot reach network resources. It is a local identity, so as far as any file server is concerned it is nobody.

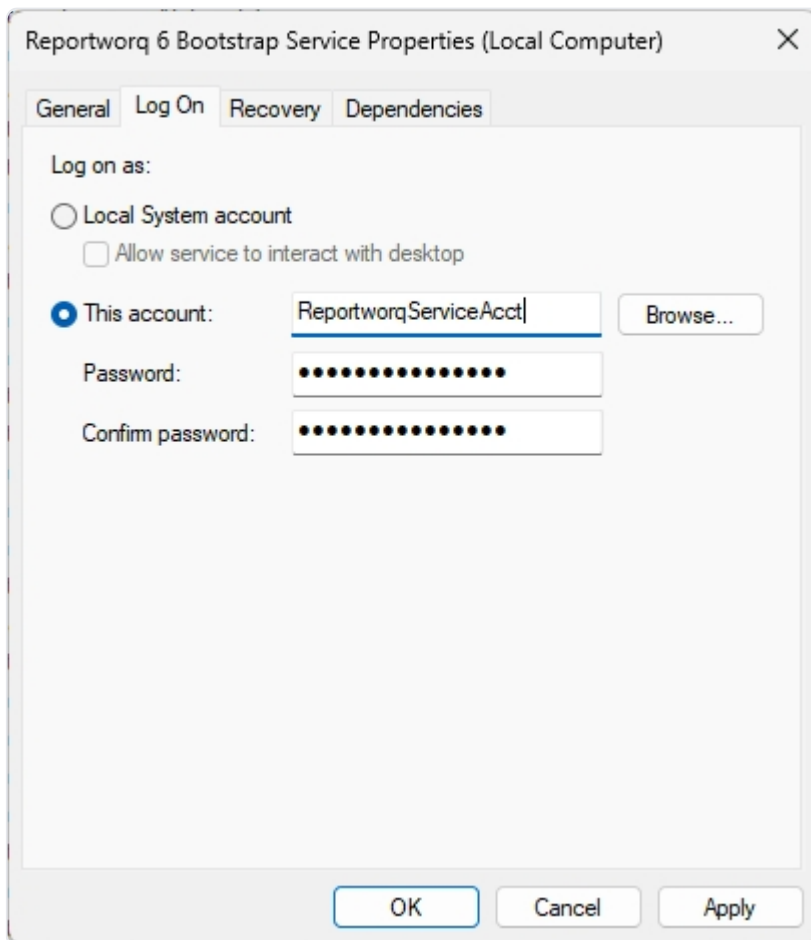
That matters because **the service account is the identity every job runs under**. When a job reads a source workbook or writes output, Windows checks the service account, not you. So:

- If every source file and every output destination is on this server, Local System is fine and you can skip to step 13.
- If **any** report source, output destination, or the Repository lives on a network share or a UNC path, change the service to run as a domain account now.

Most real installations need the domain account, so a **service account** created for the purpose is what we normally suggest. Set it up before you build jobs, not after one fails.

12. Change the service account

1. Stop the **Reportworq 6 Bootstrap Service**.
2. Right-click it, select **Properties**, and open the **Log On** tab.
3. Select **This account**, then enter the service account and its password.
4. Select **Apply**.



Windows grants that account the "Log on as a service" right for you. You still have to grant it, at minimum:

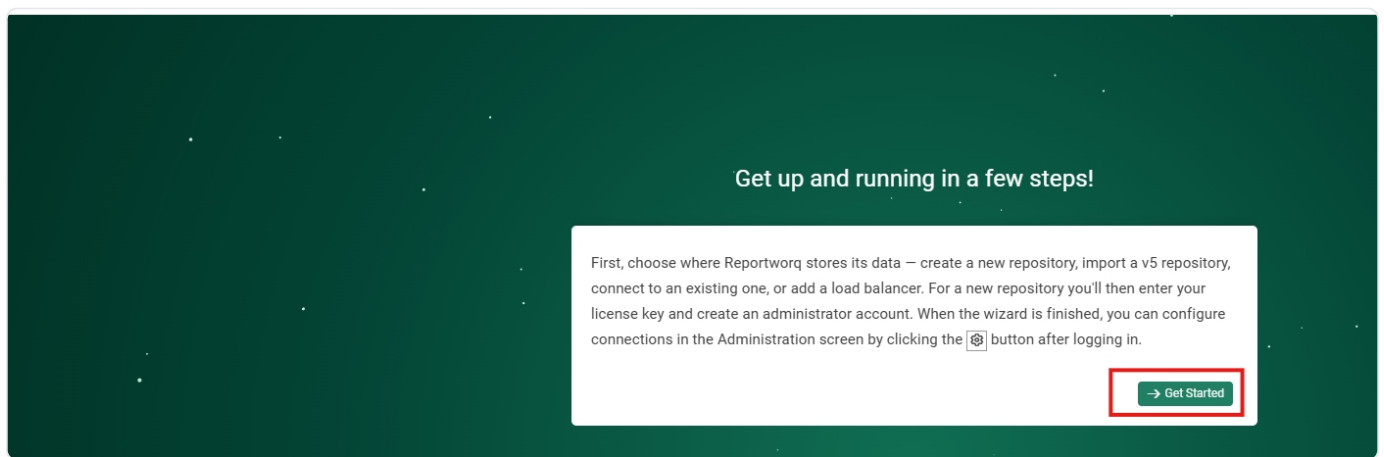
- read and write to the install folder,
- read and write to the Repository folder, if you move it outside the install root,
- read and write to every network output destination,
- read to every network location the report source files come from.

13. Start the service

Right-click **Reportworq 6 Bootstrap Service** and select **Start**. The new identity takes effect when the service starts.

14. Check it answers

On the server, browse to `https://localhost:8600` . What loads is the first-run setup wizard.



Note the **https**. On a fresh Windows install, Reportworq turns HTTPS on by itself at first boot: it generates a self-signed certificate and trusts it on the server, so the first page you ever load is already encrypted. People expect `http` here, get a browser warning or a dead page, and conclude the install failed. It did not.

Two things follow from that certificate:

- **Remote browsers will not trust it** until somebody distributes the trusted-root certificate to them. On the server itself it is already trusted.
- **Replace it for production** with a certificate from a certificate authority you trust.

This automatic step is Windows-only and happens only on a genuinely new install. Linux and container hosts serve HTTP until you configure a certificate or terminate TLS at a reverse proxy, and HTTPS is mandatory for the Excel add-in and every Microsoft 365 feature, so it is not optional there in practice.

What loads is the first-run setup wizard. That is Lab 2.

Why the service account matters more than anything else here

Of everything in this lab, the step people regret skipping is step 12.

The failure it causes does not look like a permissions problem. It looks like Reportworq is broken. You can browse to a network folder in Explorer, you can open the source workbook yourself, and the job still fails to read or write it. The reason is that the job never ran as you. It ran as the service account, and a path you can see is irrelevant if the service account cannot reach it.

It is also the most tedious thing to change later, because by then you have jobs, destinations, and schedules pointing at shares that quietly stop working the moment the identity changes. Ten minutes now.

Try this

Before you move on, prove the two facts this lab rests on:

1. **Prove both ports are listening.** In PowerShell on the server, run `Get-NetTCPConnection -LocalPort 8600,8601 -State Listen`. You should get a row for each. Do not test the Event Hub by browsing to it; it is not a web page. What you are proving is that the pair is live, which is what a remote client needs and what a single firewall rule for 8600 alone would not give you.
2. **Prove the identity.** In the Services console, look at the **Log On As** column for the Reportworq 6 row. Whatever is in that column is the account every job you build from here on will run as. If it still says Local System and your files are on a share, go back to step 12.

Notes and limits

- **This lab is Windows.** Linux installs from a tarball with an `install.sh` that creates a service user and a systemd unit, and containers are configured entirely from environment variables. There is no installer UI and no Windows service in either case, and the default web port is 8080. Linux is at feature parity with Windows, so this is a difference in delivery, not in capability. See [Install Reportworq](#) and [Deploy on an Azure Linux VM](#).
- **The hardware figures are requirements, not enforcement.** Reportworq will start on a smaller box and then struggle under real jobs.
- **Do not uninstall without releasing the license first**, or the seat stays consumed. The release control is on the License screen.

Going deeper. For every install option this lab walked past, including unattended installs, the uninstall path and air-gapped activation, see [Install Reportworq](#). For the full requirements list, see [System requirements](#).

Next

Lab 2: First-run setup and licensing. The wizard now waiting at `https://localhost:8600` creates the Repository, loads the Timberline demo content this course builds on, and applies your license key.

Lab 2: First-run setup and licensing

LAB-02

tutorial

administrator

What you are doing. Taking the installed-but-empty server from Lab 1 and making it a working Reportworq instance: choosing where it stores its data, loading the Timberline Harvest Co. content this course builds on, applying your license, and signing in as the administrator.

What this shows you. That a Reportworq install is really two decisions: *where the Repository lives*, and *what license this instance holds*. Everything else the wizard does is bookkeeping around them. It also introduces the Repository, the thing every later lab reads from and writes to.

How long. About 15 minutes, most of it the restart.

What you need. The server from [Lab 1](#), with the Reportworq 6 Bootstrap Service running, and your **Reportworq 6 license key**.

Before you start

The wizard is already waiting for you. It is what loads at `https://localhost:8600` on a server whose Repository is empty, in place of the sign-in screen.

Two things are worth knowing before you start clicking, because they decide which of the two routes below you take.

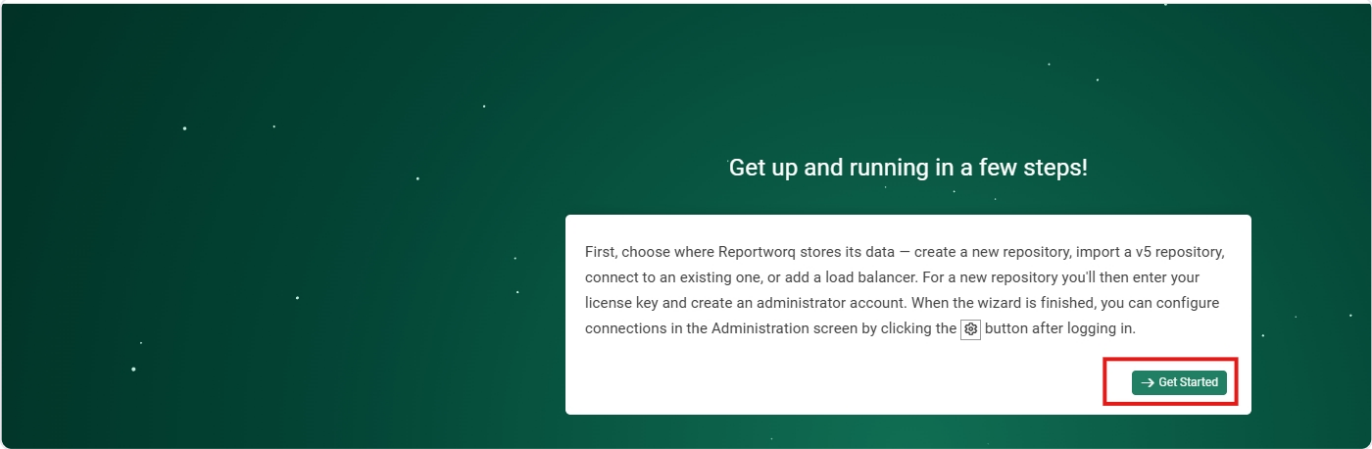
The Repository is a folder, and it is the whole system. Workspaces, jobs, reports, schedules, settings, audit history and run history all live in it. Reportworq does not use a database you have to provision; the Repository is a directory on disk, and backing Reportworq up means backing that directory up. Where you put it is the one decision in this lab that is genuinely hard to change later, which is why the wizard asks first.

You are going to install the demo content, not build from an empty repository. Labs 3 to 9 build on the Timberline Harvest Co. workbooks and reports, so this lab takes the **Set up the Timberline demo system** route, which extracts the whole demo store for you in one click. Part C covers what the **Create a new repository** route does instead, because that is the route you will take on a real production install, and it has two steps the demo route does not.

Part A: Open the wizard

1. Browse to the server

On the server, go to `https://localhost:8600` . The wizard opens on its Welcome step. Select **Get Started**.



If you get the sign-in screen instead of this, the Repository is not empty, which means somebody has already run the wizard on this server. See **Notes and limits** at the end for how to reopen it.

2. Choose where Reportworq stores its data

The next step is the one that matters. It offers five routes:

Option	What it does
Create a new repository	Starts fresh at an empty path. The normal choice for a real install.
Add a load balancer to an existing web app	Runs this install as an extra execution node sharing another instance's Repository.
Connect to an existing repository	Points this install at a folder that already holds Reportworq v6 data.
Import a Reportworq v5 repository	Builds a new v6 Repository from a copy of a v5 one, leaving the v5 original untouched.
Set up the Timberline demo system	Extracts the ready-made Timberline Harvest Co. demo store into a new Repository.

Select **Set up the Timberline demo system**, leave **Repository path** blank so it takes the default shown, and select **Set up Timberline**.

Where should Reportworq store its data?

Tell Reportworq how to set up its repository — the folder that stores all of your workspaces, jobs, settings and history. Leave the path blank to use the default location (`C:\Program Files\Reportworq 6\Repository`).

- ☐  Create a new repository
Start fresh. The path must be empty or unused.
- ☐  Add a load balancer to an existing web app
Run this install as an additional execution node that shares an existing repository.
- ☐  Connect to an existing repository
Use a repository that already contains Reportworq data.
- ☐  Import a Reportworq v5 repository
Create a new v6 repository from a copy of a v5 repository.
- ☒  **Set up the Timberline demo system**
One click to a ready-to-run demo — the static Timberline Harvest Co. store, no data sources to connect.

Extracts the ready-made **Timberline Harvest Co.** demo — reports, jobs, campaigns, schedules and sample data — into a new repository and points Reportworq at it. This is the **static** demo: it needs no Planning Analytics server or any other external data source.

Sign in after setup

Username `admin@rw.com` · Password `Admin_1234`



Read the Timberline documentation

Repository path

`C:\Program Files\Reportworq 6\TimberlineRepo`

 Browse...

Leave blank to install to `C:\Program Files\Reportworq 6\TimberlineRepo`, alongside where a normal repository folder would be created. The folder must be empty or unused.

[Back](#)

[→ Set up Timberline](#)

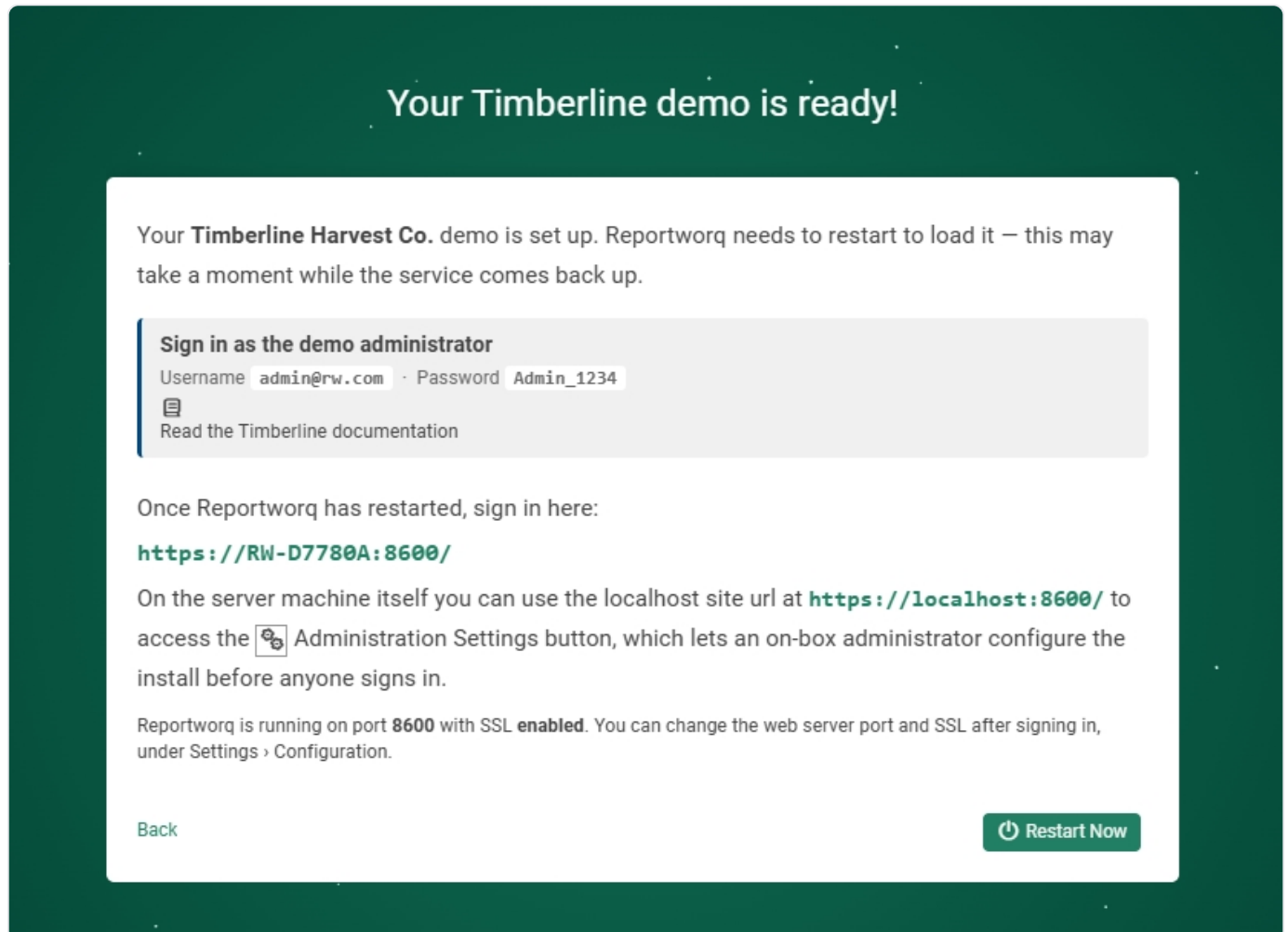
That default path is deliberate. The demo goes to `C:\Program Files\Reportworq 6\TimberlineRepo`, *alongside* where a normal `Repository` folder would go rather than in it, so the demo is obviously the demo and a real Repository can sit next to it later. The path you give must be empty or not exist; the wizard refuses a folder that already holds a repository.

Note what the option says about itself: this is the **static** Timberline store. It carries reports, jobs, campaigns, schedules and sample data, and it needs no external data source to run. That is what makes it usable as a course.

Part B: Set up the Timberline demo system

3. Restart to load it

Extraction takes a few seconds, and then Reportworq tells you it needs to restart to bind the new Repository. Select **Restart Now**.



Write down what this screen is telling you before you leave it:

- **The demo administrator is admin@rw.com , password Admin_1234 .** That is public on purpose. Timberline is a fictitious company on a demo store, so the wizard prints the credential on screen rather than making you hunt for it. It is not a credential that means anything outside the demo.
- **Two URLs, and they are not interchangeable.** The machine URL (https://<your-server>:8600/) is what everyone else uses; the localhost URL works only on the server itself.
- **Do not go looking for the Administration Settings button yet.** This screen mentions it, but that cog is a **lockout-recovery** surface and it is not on the sign-in screen of a normal install. It appears only after an administrator has deliberately opened recovery, by creating a lockout-recovery.txt file in the installation

directory or starting the service with `RW_CONFIG_ACCESS=1`. Being at the server console is not enough by itself. See [Sign-in and SSO](#) for when you would actually need it.

- **The port and SSL line is live, not boilerplate.** It reports the binding this install actually has. On a fresh Windows install that reads port **8600** with SSL **enabled**, because Reportworq generated and trusted a self-signed certificate at first boot (see Lab 1). On Linux or in a container it will read 8080 and SSL disabled.

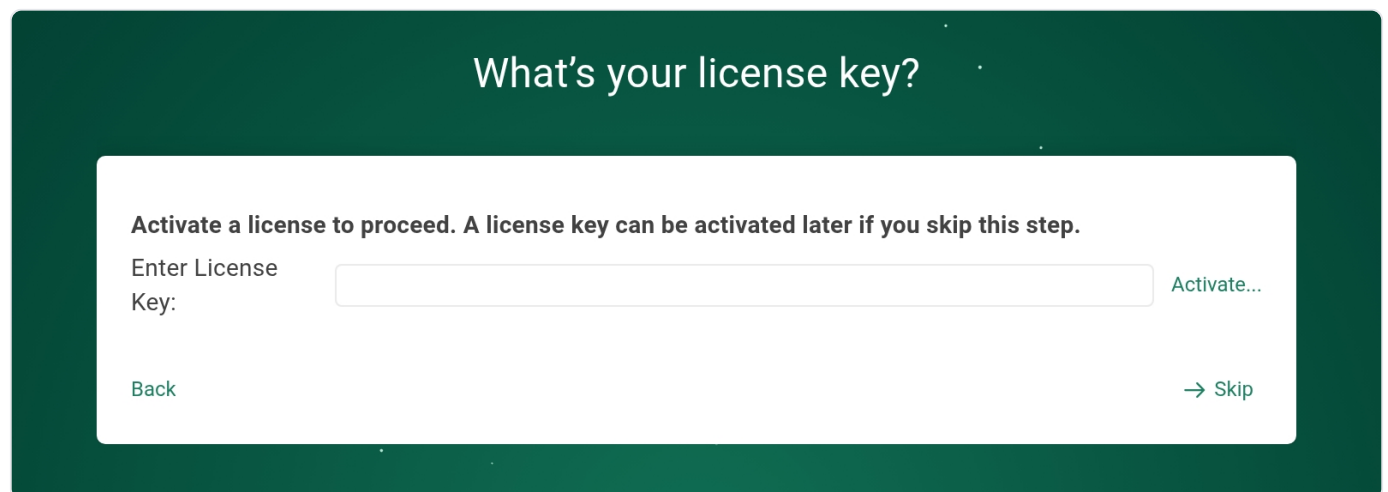
The restart can take a minute while the service comes back up. If the page seems to hang, refresh it.

This is the end of the wizard on the Timberline route. The demo store brings its own accounts, so when Reportworq comes back it has no reason to ask you who the administrator should be. It goes straight to the sign-in screen. Skip to [Part D](#).

Part C: What a new repository does instead

If you had chosen **Create a new repository** in step 2, Reportworq would also have restarted to bind it, and then the wizard would have **resumed**, not restarted, at two more steps that the demo route has no need for. This is the route a real install takes, so it is worth seeing even though this course does not take it.

The License step



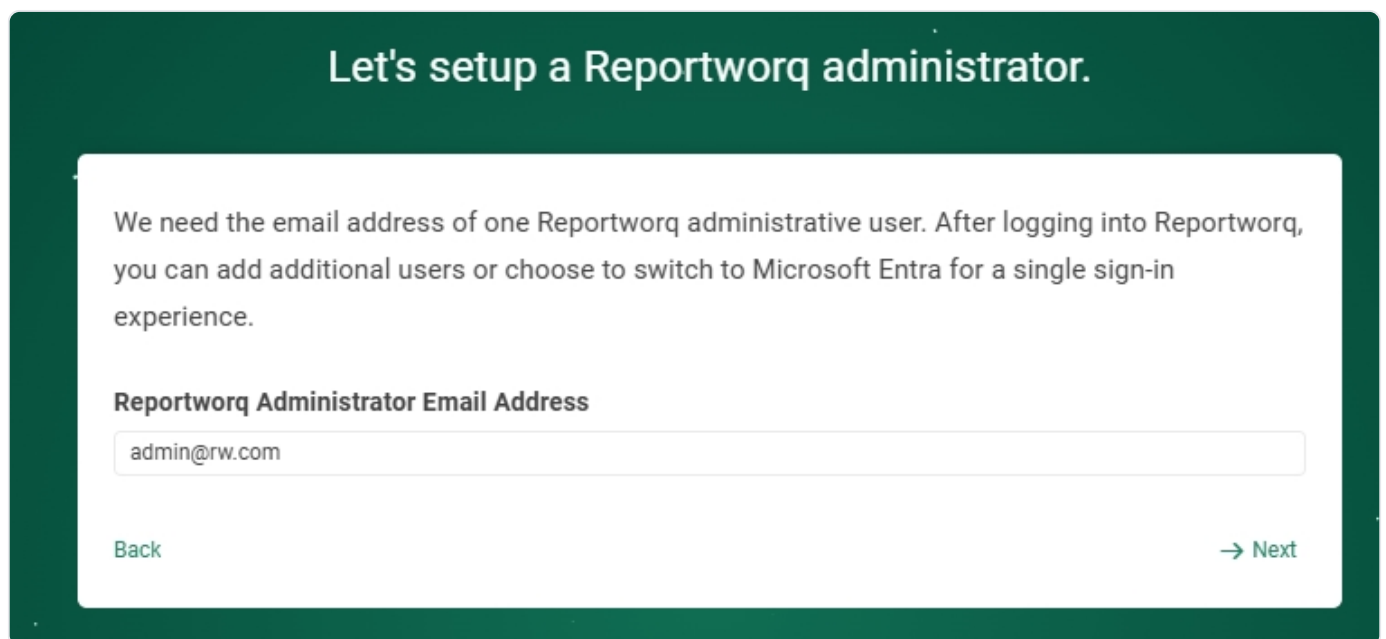
The screenshot shows a dark green background with a white rectangular box in the center. At the top of the box, the text "What's your license key?" is displayed in a light green font. Below this, a message reads: "Activate a license to proceed. A license key can be activated later if you skip this step." Underneath the message, there is a label "Enter License Key:" followed by a text input field. To the right of the input field is a button labeled "Activate...". At the bottom left of the box is a link labeled "Back", and at the bottom right is a link labeled "→ Skip".

This is where a license key is entered during setup. Paste the key in and select **Activate**. The instance contacts the licensing service, applies your entitlements, and the warning above the box is replaced by a line confirming what you hold: the subscription's user count and its end date, or the days left if it is a trial. The bottom-right button changes from **Skip** to **Next**.

Three things about this step catch people out:

- **Skipping is allowed, and the consequence is not obvious.** The copy says a key can be activated later, and that is true, but an unlicensed Reportworq is unavailable to everyone until a key is applied. Skipping does not get you a limited instance; it gets you a locked one. Part D shows exactly what that looks like.
- **Type the key and the button changes.** With a key in the box the button reads **Activate & Continue**, so a typed key is never thrown away by a stray **Skip**.
- **No outbound access?** Use **Manual Activation** rather than retrying. Activation needs to reach the licensing service, and the manual path exists for servers that cannot.

The administrator step



The screenshot shows a dark green header with the text "Let's setup a Reportworq administrator." in white. Below this is a white box containing the following text: "We need the email address of one Reportworq administrative user. After logging into Reportworq, you can add additional users or choose to switch to Microsoft Entra for a single sign-in experience." Below the text is a label "Reportworq Administrator Email Address" and a text input field containing "admin@rw.com". At the bottom left of the white box is a "Back" link, and at the bottom right is a "→ Next" link.

Enter the email address of the person who will administer this instance and select **Next**. One account is all the wizard wants; you add the rest, or switch the instance to Microsoft Entra single sign-on, after you are in.

Congratulations, you're all set

Congratulations, you're all set!

You're all set. Sign in to Reportworq here:

<https://RW-D7780A:8600/>

On the server machine itself you can use the localhost site url at **<https://localhost:8600/>** to access the  Administration Settings button, which lets an on-box administrator configure the install before anyone signs in.

Reportworq is running on port **8600** with SSL **enabled**. You can change the web server port and SSL after signing in, under Settings › Configuration.

The Reportworq Forum is the best way to stay informed about new releases and other updates. Please sign-up for an account at the link below.

 <https://forum.reportworq.com>

[Back](#)

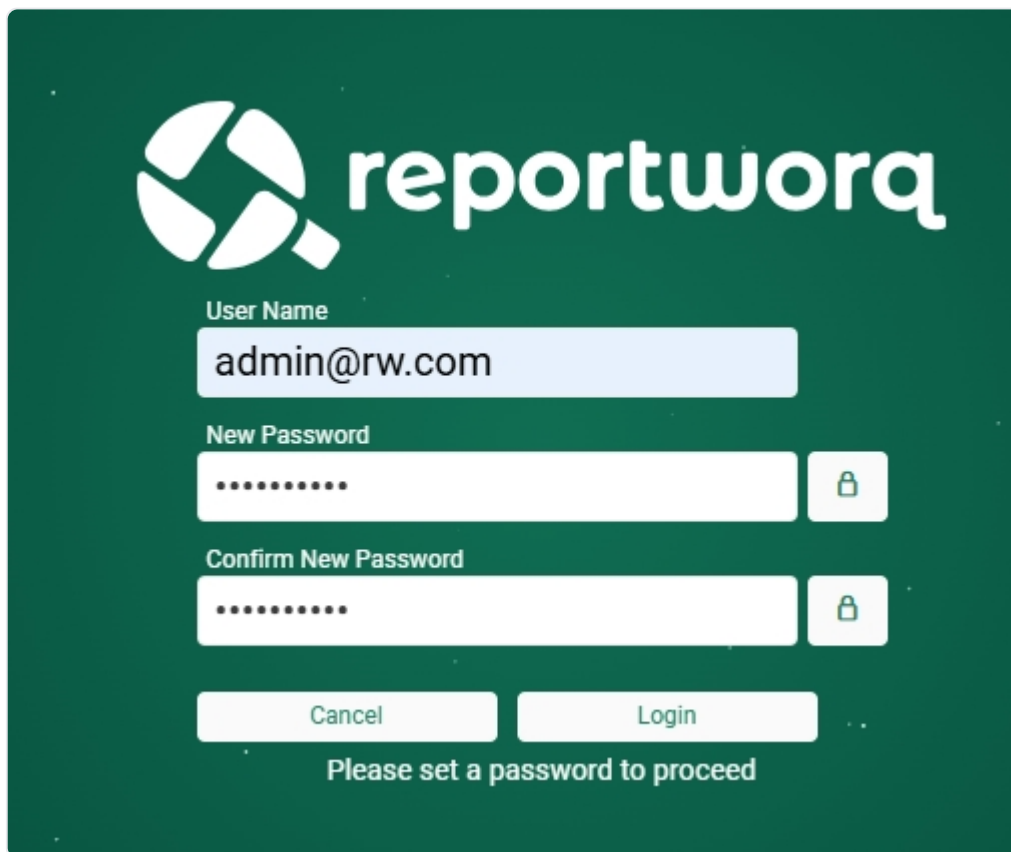
[→ Login](#)

The wizard is done. It repeats the two URLs and the port/SSL line from step 3, and selecting **Login** closes the wizard for good and reloads into the normal application.

Two things happened quietly when you did that, and both matter on a real install:

- **A workspace named `Default` was created**, so you land somewhere usable rather than in an empty shell.
- **The instance was opened wide.** The built-in **All Users** group was granted the System Administrator entitlement and membership of that workspace, so every account that signs in is an administrator. That is a bootstrap posture, not a recommendation. If more than one person will use this environment, tighten it before you add accounts, see [Accounts, groups, and entitlements](#).

On this route the account you named has no password yet, so the first sign-in asks you to set one.

The image shows a login screen for Reportworq. At the top left is the Reportworq logo, which consists of a stylized circular icon made of four segments and the word "reportworq" in a lowercase, sans-serif font. Below the logo, there are three input fields. The first is labeled "User Name" and contains the text "admin@rw.com". The second is labeled "New Password" and contains a series of dots, with a small lock icon to its right. The third is labeled "Confirm New Password" and also contains a series of dots, with a small lock icon to its right. At the bottom of the form are two buttons: "Cancel" on the left and "Login" on the right. Below these buttons, centered, is the text "Please set a password to proceed".

 **reportworq**

User Name

New Password
 

Confirm New Password
 

Please set a password to proceed

Part D: Sign in, and license the instance

Back on the Timberline route. Reportworq has restarted and is showing the normal sign-in screen.

4. Sign in as the demo administrator

Enter `admin@rw.com` and `Admin_1234` and select **Login**. The demo administrator already has a password, so there is nothing to set.

5. Apply your license key

You do not land in the application. You land here:





 **CONFIGURATION REQUIRED**

No valid license is set. You are in the configuration area because Reportworq has no valid license. The application is unavailable to all users until this is resolved – set a valid license key on the **License** tab now to continue.

Settings > Configuration > **License**

Close

Save Changes

Restart

Configuration

Server, email, execution, logging, API, data collection and license.

Web Server

Email

Performance

Logging

Authentication

License

License Information

License Status

Reportworq requires a subscription.

License Key



Activate

Release

[Manual Activation...](#)

License Entitlements

☐ Distribution

☐ Contribution (includes Data Collection)

☐ API (includes Script Runner)

Administrators

1 user(s)

Authors

Not licensed

Users

Not licensed

Load Balancer

Not licensed

Data Connections

Unlimited

Workspaces

Unlimited

The Timberline demo ships unlicensed, deliberately. An activated subscription is never packaged into the demo, and the build refuses to produce a demo package that contains one, so the first thing the demo asks you for is your own key. This is the answer to "where does the license key go?" on this route: not in the wizard, but here, on the first sign-in.

Paste your key into **License Key**, select **Activate**, and then **Restart** when it asks. On the way back up the entitlement list fills in: Distribution, Contribution, API, and the account counts your subscription carries.

This is the same screen you will use for the rest of the instance's life: **Settings ► Configuration ► License** is where you check what you hold, release a seat before decommissioning a server, or activate a server with no outbound access via **Manual Activation**.

6. Accept the license agreement

Reportworq asks you to accept the End User License Agreement. Tick **I have read and accept the End User License Agreement** and select **Accept & Continue**.



End User License Agreement

Reportworq was recently updated to version **6.0.1.7**. Please review and accept the End User License Agreement before continuing.

 [Reportworq Software Subscription & Services Agreement](#)

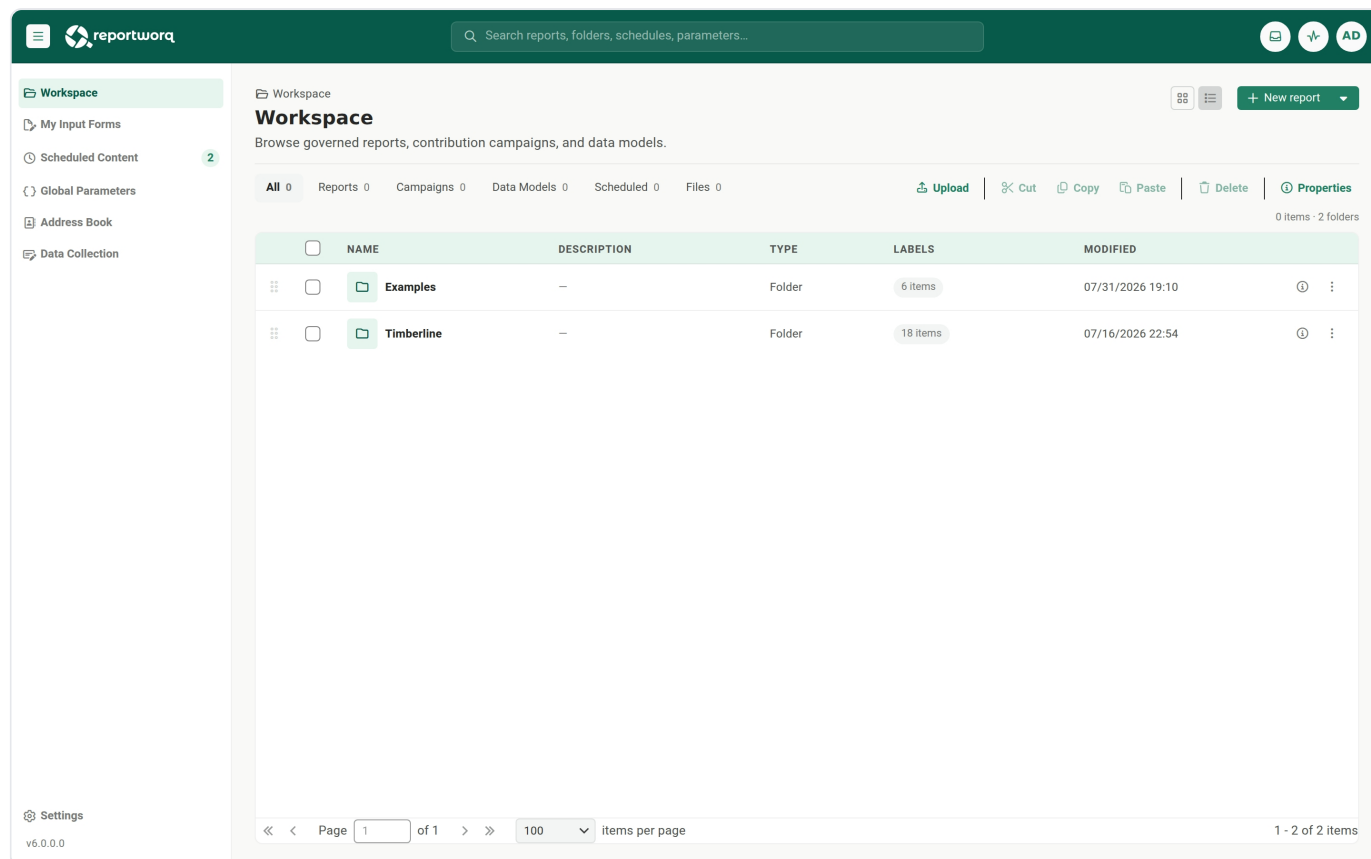
☒ I have read and accept the End User License Agreement

[Decline & Logout](#)

[Accept & Continue](#)

Acceptance is recorded in the Repository against the version that was accepted and the account that accepted it, so you see it again after an upgrade rather than on every sign-in. That also means you may not see it here at all: if the demo store you just installed already carries an acceptance for the exact version you are running, Reportworq signs you straight through to step 7.

7. Land in the Workspace



Reportworq is installed, licensed, and carrying the demo content. The two folders are the material for the rest of this course:

- **Timberline** is Timberline Harvest Co.'s own content, under a **Financial Reports** folder: the Financial Statements pack, the Board Packet with Division Slides, the Divisional Packet, the Sales Rep Scorecard and a contribution campaign. Those are the jobs Labs 4 to 9 build, one at a time.
- **Examples** is a small set of chart reports and the workbooks behind them, for looking up how one feature is wired without opening a whole pack.

The left rail is the whole application: **Workspace** to browse and build, **Scheduled Content** for what runs on its own, **Global Parameters**, the **Address Book** of distribution recipients, and **Data Collection**.

Why the order is what it is

The wizard asks for the Repository before the license, and it is worth understanding why, because the sequence is not arbitrary.

The license is stored in the Repository. So there is nothing to store a license *in* until the Repository exists and the service has restarted to bind it. That is also why both

routes restart in the middle: the bind is a process-level thing, not a setting. On the new-repository route the wizard remembers where it was and resumes at the License step; on the Timberline route the extracted store already has accounts, so the wizard has nothing left to ask and stands down.

It also explains the one thing that surprises people about the demo route: you get an instance that is fully populated and completely unusable until you license it. The content came from the package; the license can only come from you.

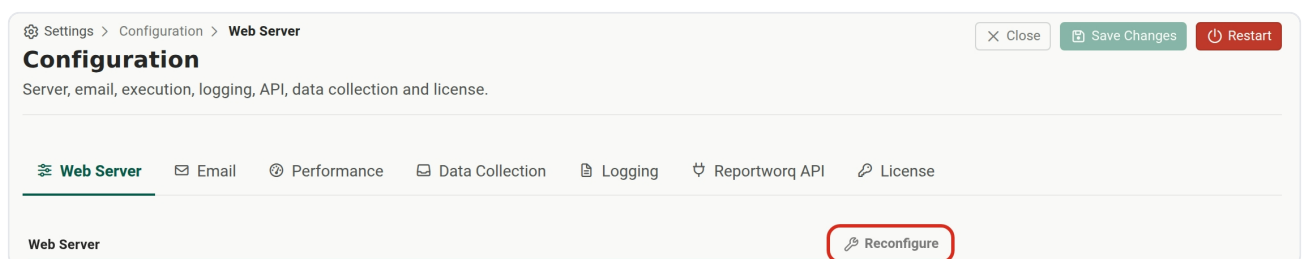
Try this

Two checks, both of which prove something the later labs depend on.

1. **Find the Repository on disk.** Open `C:\Program Files\Reportworq 6\TimberlineRepo` in Explorer. Everything you just signed in to is in there, and in `v6\data` you will find the folders behind the screens you have seen, including `license-artifacts`, where the key you activated now lives. This is what "back up Reportworq" means.
2. **Find the jobs you are going to build.** In the Workspace, open **Timberline**, then **Financial Reports**. The Financial Statements pack, the Board Packet with Division Slides, the Divisional Packet and the Sales Rep Scorecard are all sitting there, finished. Labs 4 to 9 build each of them from nothing, so treat this folder as the answer key: open one now if you want to see where you are heading.

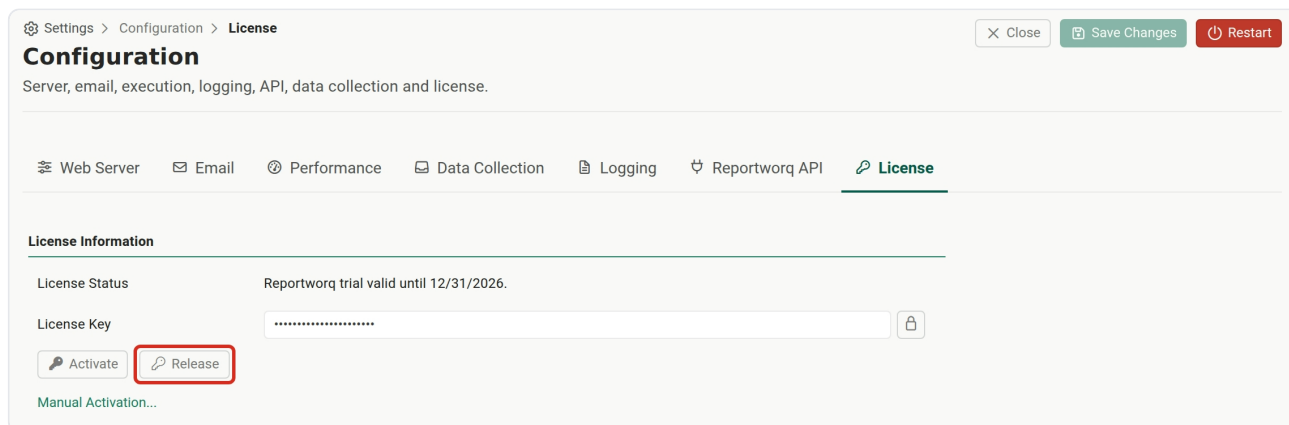
Notes and limits

- **The wizard runs once, on an empty Repository.** A server that already has a configured Repository, or any accounts, will not show it again by itself. To change the Repository path, connect or import a repository, or convert the install into a load-balancer node afterwards, go to **Settings ► Configuration ► Web Server** and select **Reconfigure**, at the right of the Web Server section heading.



- **The demo option only appears if the demo shipped with your install.** It is packaged alongside the binaries; a build without it shows four repository options rather than five.

- **The static demo is deliberately data-source-free.** It needs no external data source at all, which is what makes it safe to hand to anyone. Lab 3 is where you connect a real data source and a report provider of your own.
- **Do not uninstall without releasing the license first**, or the seat stays consumed. **Release** is on the same License screen you used in step 5, next to **Activate**.



- **Different defaults off Windows.** On Linux or in a container the wizard is the same, but the web server defaults to port 8080 with SSL disabled, and an immutable deployment (Docker, Azure App Service) takes its Repository from the environment, so the repository step is skipped entirely and the wizard opens on the License step.

Going deeper. For every repository option in full, including the v5 import path and the load-balancer node, see [First-run setup](#). For the license model, entitlements and the manual activation path, see [Licensing and entitlements](#). For what is inside the demo store, see [What is Timberline](#).

Next

Lab 3: Connect your data and your source files. The instance is running and licensed, but everything it holds came out of a box. Lab 3 points Reportworq at report source files and a data source of your own.