Compass Hub

Installation guide for IT

version 0.9.2 · October 2026

Loonar Mariusz Głowacki
ul. Gombrowicza 12, 73-110 Stargard
loonar.eu · mariusz@loonar.eu

Contents

1. What you install
2. Before you start
3. Step 1 – Files and folder
4. Step 2 – hub.json and the models
5. Step 3 – Secrets
6. Step 4 – The Windows service and the first sign-in
7. Step 5 – HTTPS with IIS (URL Rewrite + ARR)
8. Step 6 – Company sign-in (AD FS or Entra ID)
9. Step 7 – AI keys, Jira, Confluence and the team policy
10. Step 8 – The workstations (Group Policy)
11. Step 9 – Acceptance checklist
12. Operation: update, backup, logs, removal
13. Troubleshooting
14. Linux and Docker

1. What you install

Compass Hub is the team service of Loonar Compass. It is one program, `compass-hub.exe`, without an installer and without Enterprise Architect. It runs as a Windows service and keeps:

- the team's EA models (read only): `.qea` / `.qeax` files or a PostgreSQL, SQL Server, MySQL or Oracle repository;
- the team policy, which the Compass apps follow: the minimum approval mode, web access, the AI only through the hub, and so on;
- the audit trail of the changes from all workstations, the change sets for review, and a web view for people without EA;
- the AI gateway: the Claude API keys stay on the server, with a key and limits per person or role;
- Jira and Confluence through one service account, a read-only GraphQL API, tables for Power BI and an MCP server.

Where	What	Network
Hub server	Service `CompassHub` (compass-hub.exe), listening only on `127.0.0.1:8730`	Local only
Hub server	IIS site `https://hub.company.com` with URL Rewrite + ARR in front of the service	Inbound 443 from the workstations
Identity provider	AD FS or Microsoft Entra ID: the "Compass Hub" application (OpenID Connect)	The browser of the user
Hub server → internet	`api.anthropic.com` (AI gateway), the Atlassian site (Jira/Confluence)	Outbound 443, only for the parts you use
Workstations	Loonar Compass (desktop and add-in in EA), settings forced by Group Policy	Outbound 443 to the hub

> This is the setup tested end to end on Windows Server 2022 with AD, AD CS, AD FS and IIS. Each step ends with a check, so a problem shows up at the step that caused it.

2. Before you start

✓	Item	Notes
☐	Windows Server 2019 or 2022, x64	2 vCPU, 4 GB RAM and 2 GB disk are enough for a team; more disk for large model files. No Enterprise Architect.
☐	DNS name, e.g. `hub.company.com`	Resolvable from the workstations.
☐	TLS certificate for that name	From the company CA or a public CA. **The workstations must trust its root** (Compass checks it against the Windows certificate store). The hub name must be in the certificate's SAN; from AD CS, the account that requests it needs Enroll on the Web Server template.
☐	IIS with the URL Rewrite and Application Request Routing (ARR) modules	Downloads from Microsoft (iis.net).
☐	AD FS (Windows Server) or Microsoft Entra ID	Only for company sign-in, which is recommended. Without it the hub has its own users and passwords.
☐	AD groups: `Compass-Admins`, `Compass-Architects`, `Compass-Viewers`	The group gives the role in the hub. Other names are possible (step 6).
☐	The models	.qea files copied to the server, or read access to the EA repository database (a read-only account is enough).
☐	Secrets ready, held by the person who installs	Claude API key(s), Atlassian API token, OIDC client secret, DB password. They are never put into hub.json.
☐	Outbound HTTPS from the server	`api.anthropic.com` (AI gateway), your Atlassian site (Jira/Confluence). Through a proxy: set `HTTPS_PROXY` for the service (step 3).

Files from the release: `CompassHub-0.9.0-windows-x64.zip` (with `compass-hub.exe`, `hub.example.json`, `README.md`, the licences), and for the workstations `LoonarCompass-Setup-0.9.exe` plus the `policies` folder (ADMX/ADML).

3. Step 1 – Files and folder

1. Check the SHA-256 of the download against the value published with the release:

   ```
   Get-FileHash .\CompassHub-0.9.0-windows-x64.zip -Algorithm SHA256
   ```

2. Unpack it to `C:\CompassHub` and create the subfolders `models` and `secrets`.
3. Restrict the folder to the service and the administrators. The hub writes its data here, and the secrets are here:

   ```
   icacls C:\CompassHub /inheritance:r /grant:r "SYSTEM:(OI)(CI)F" "BUILTIN\Administrators:(OI)
   (CI)F"
   ```

   On a Windows in another language, use the SID `*S-1-5-32-544` instead of the name of the Administrators group.

Check: `C:\CompassHub\compass-hub.exe --version` prints `0.9.0`.

4. Step 2 – hub.json and the models

Copy `hub.example.json` to `C:\CompassHub\hub.json`. A minimal start:

```
{
  "listen": "127.0.0.1:8730",
  "sources": [
    { "id": "architecture", "name": "Enterprise architecture", "properties":
"models/architecture.properties" }
  ],
  "policy": { "min_approval": "ask" }
}
```

- `listen` stays on `127.0.0.1`: only IIS on the same server talks to the hub.
- `sources`: one entry per model. `id` is short and stable, because the users and Power BI use it. `properties` is relative to hub.json.
- Leave out `users`: the hub then creates the account `admin` (step 4) and writes users back itself, as hashes only.

The model file, e.g. `C:\CompassHub\models\architecture.properties`:

```
# An EA project file (read only)
model.source=file
model.file=C:/CompassHub/models/Architecture.qea

# ... or a repository database (the password is the secret DB_PASSWORD):
# model.source=sqlserver          (or postgres, mysql, oracle)
# db.host=sql.company.com
# db.port=1433
# db.name=ea_repository
# db.user=compass_reader
# db.sslmode=verify-full
```

SQL Server, MySQL and Oracle are read through the vendor's 64-bit ODBC driver, which must be installed on the server. For SQL Server, "ODBC Driver 17 for SQL Server" (it comes with SQL Server 2022) or 18 is enough; the installed drivers: `Get-OdbcDriver -Name '*SQL Server*' -Platform 64-bit`. PostgreSQL and files need nothing more. Every read is rolled back, so a read-only database account is enough. SQL Server without the secret `DB_PASSWORD` uses Windows authentication: the hub service (LocalSystem) signs in as the computer account (`DOMAIN\SERVER$`), which then needs the `db_datareader` role in the model database.

- **Named instance** (e.g. SQL Server Express): `db.host=SERVER\SQLEXPRESS` without a port, with the SQL Server Browser service running; or a fixed TCP port (SQL Server Configuration Manager → TCP/IP → IPAll: TcpPort 1433, TcpDynamicPorts empty) and `db.host` / `db.port`.
- **Your own connection string:** `db.odbc=Driver={ODBC Driver 17 for SQL Server};Server=sql.company.com,1433;Database=ea_repository;Trusted_Connection=Yes` replaces host, port, name and user (without a password; `DB_PASSWORD`, if set, is added).
- **db.sslmode** for SQL Server: `prefer` = encryption when the server and the driver use it (no certificate check); `require` = always encrypted, the certificate is not checked; `verify-full` = encrypted and the certificate must be trusted and match the server name (not with a self-signed SQL Server certificate).

> Keep the hub's model files separate from the files people edit in EA. To refresh a file model, replace the file. The hub opens it read-only for each request, so no restart is needed.

5. Step 3 – Secrets

Each secret is a text file in `C:\CompassHub\secrets` that contains only the value. The service learns the path from an environment variable `LOONAR_SECRET_<NAME>_FILE`. The person who knows the value creates the file, e.g. in Notepad or with:

```
Read-Host -AsSecureString "Value" | ConvertFrom-SecureString -AsPlainText | Set-Content
C:\CompassHub\secrets\claude_key_team -NoNewline
```

(`-AsPlainText` needs PowerShell 7; in Windows PowerShell 5.1, Notepad is simpler.)

File (example)	Variable for the service	Needed for
`claude_key_team`	`LOONAR_SECRET_CLAUDE_KEY_TEAM_FILE`	The AI gateway: one variable per key named in hub.json (step 7)
`claude_api_key`	`LOONAR_SECRET_CLAUDE_API_KEY_FILE`	"Ask" in the web view
`jira_token`	`LOONAR_SECRET_JIRA_TOKEN_FILE`	Jira
`confluence_token`	`LOONAR_SECRET_CONFLUENCE_TOKEN_FILE`	Confluence
`oidc_client_secret`	`LOONAR_SECRET_OIDC_CLIENT_SECRET_FILE`	Company sign-in
`db_password`	`LOONAR_SECRET_DB_PASSWORD_FILE`	A model database

The name in the variable is the secret name in capitals, with other characters as `_`: the hub.json secret `claude_key_anna` is `LOONAR_SECRET_CLAUDE_KEY_ANNA_FILE`.

The variables are set on **the service only**, not for the whole computer, after the service is installed (step 4). This takes effect at the next start of the service, with no reboot:

```
$vars = @(
  'LOONAR_SECRET_CLAUDE_KEY_TEAM_FILE=C:\CompassHub\secrets\claude_key_team',
  'LOONAR_SECRET_JIRA_TOKEN_FILE=C:\CompassHub\secrets\jira_token',
  'LOONAR_SECRET_CONFLUENCE_TOKEN_FILE=C:\CompassHub\secrets\confluence_token',
  'LOONAR_SECRET_OIDC_CLIENT_SECRET_FILE=C:\CompassHub\secrets\oidc_client_secret'
  # 'HTTPS_PROXY=http://proxy.company.com:8080'   only if the server goes out through a proxy
)
Set-ItemProperty HKLM:\SYSTEM\CurrentControlSet\Services\CompassHub -Name Environment -Type
MultiString -Value $vars
Restart-Service CompassHub
```

The hub reads a secret when it first needs it. A missing one is reported where it is used, e.g. "secret 'jira_token' is not set: give it as LOONAR_SECRET_JIRA_TOKEN (or a file named in LOONAR_SECRET_JIRA_TOKEN_FILE)". The values never appear in hub.json, the logs or the web view. A byte order mark at the start and spaces or a new line at the end of a secret file (Notepad) are ignored.

6. Step 4 – The Windows service and the first sign-in

1. In PowerShell as administrator:

   ```
   C:\CompassHub\compass-hub.exe --service-install C:\CompassHub\hub.json
   ```

   This creates the service `CompassHub` ("Compass Hub (Loonar Compass team service)"), with automatic start under LocalSystem, and starts it.
2. Now set the secrets of the service (step 3) and run `Restart-Service CompassHub`.
3. On the server, open `http://127.0.0.1:8730/` and sign in as `admin` / `passw0rd` (with a zero). The hub asks for a new password at once and allows nothing else until then. The password needs at least 10 characters. Do this **before** step 5 makes the hub reachable.

Check:

- `Get-Service CompassHub` shows Running;
- `C:\CompassHub\hub.log` shows the start and your sign-in;
- the web view lists the model from step 2.

7. Step 5 – HTTPS with IIS (URL Rewrite + ARR)

The hub speaks plain HTTP on 127.0.0.1. IIS gives it the company name and certificate.

1. Install IIS with URL Rewrite and ARR. Create a site, e.g. **CompassHub**, with a binding for `https`, port 443, the host name `hub.company.com` and the certificate. The physical path is an empty folder, e.g. `C:\inetpub\compasshub`.
2. Turn on the proxy and switch off rewriting the host in redirects. This is **required**; otherwise ARR rewrites the redirect to AD FS to the hub's own address, and the company sign-in fails. Set the timeout for long AI answers as well:

   ```
   Import-Module WebAdministration
   $p = 'MACHINE/WEBROOT/APPHOST'
   Set-WebConfigurationProperty -PSPath $p -Filter system.webServer/proxy -Name enabled -Value True
   Set-WebConfigurationProperty -PSPath $p -Filter system.webServer/proxy -Name
   reverseRewriteHostInResponseHeaders -Value False
   Set-WebConfigurationProperty -PSPath $p -Filter system.webServer/proxy -Name timeout -Value
   '00:05:00'
   ```

3. Put this `web.config` in the site folder. It forwards everything to the hub and allows request bodies up to 32 MB, because questions with model diagrams are large:

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <system.webServer>
    <rewrite>
      <rules>
        <rule name="CompassHub" stopProcessing="true">
          <match url="(.*)" />
          <action type="Rewrite" url="http://127.0.0.1:8730/{R:1}" />
        </rule>
      </rules>
    </rewrite>
    <security>
      <requestFiltering>
        <requestLimits maxAllowedContentLength="33554432" />
      </requestFiltering>
    </security>
  </system.webServer>
</configuration>
```

4. Firewall: allow inbound 443 only. Port 8730 is never opened, because the hub listens on 127.0.0.1.
5. AD FS on the same server: AD FS also uses port 443 (HTTP.sys). Tick **Require Server Name Indication** on the IIS binding (`New-WebBinding … -SslFlags 1`), or IIS and AD FS get in each other's way.
6. The hub does not need `preserveHostHeader` or `X-Forwarded-Proto`: its addresses come from `redirect_uri` in hub.json.

The hub trusts `X-Forwarded-For` (which ARR adds) only from a proxy on the same computer. The audit trail and the lockout of wrong tokens then see the real client address.

Check: from a workstation, `https://hub.company.com/` shows the sign-in page with no certificate warning.

8. Step 6 – Company sign-in (AD FS or Entra ID)

AD FS

1. AD FS Management → Application Groups → Add, then the template **Server application accessing a web API**, name "Compass Hub".
2. Server application:
   - note the **client identifier**;
   - redirect URI: `https://hub.company.com/auth/callback`;
   - credentials: **Generate a shared secret**. Save it right away into `C:\CompassHub\secrets\oidc_client_secret`; AD FS shows it only once.
3. Web API:
   - identifier: the client identifier from the previous step;
   - access control policy: the people who may use the hub, e.g. "Permit everyone" (roles still come from the groups);
   - permitted scopes: `openid`, `profile`, `email`, `allatclaims`.
4. Web API → Issuance Transform Rules → "Send LDAP Attributes as Claims" (Active Directory):
   - *User-Principal-Name* → `upn`;
   - *E-Mail-Addresses* → `email`;
   - *Display-Name* → `name`;
   - *Token-Groups - Unqualified Names* → `groups`.

   (Type the outgoing claim names exactly like that.)

In `hub.json` (the hub adds `allatclaims` for AD FS itself, so the claims reach the ID token):

```
"oidc": {
  "issuer": "https://adfs.company.com/adfs",
  "client_id": "<client identifier>",
  "redirect_uri": "https://hub.company.com/auth/callback",
  "groups_claim": "groups",
  "roles": { "admin": ["Compass-Admins"], "architect": ["Compass-Architects"], "viewer": ["Compass-
Viewers"] },
  "password_login": true
}
```

Microsoft Entra ID

The setup in Entra ID:

- **App registration:** platform *Web*, redirect URI `https://hub.company.com/auth/callback`, a client secret (into the secret file), and the ID token with the optional claims `upn` and `email`.
- **Issuer:** `https://login.microsoftonline.com/<tenant-id>/v2.0`.
- **Groups:** Entra sends them as object IDs by default. Either put those IDs in `roles`, or, for groups synchronised from AD, choose "sAMAccountName" in the token configuration.

Rules the hub applies

- The role comes from the groups at every sign-in, and the highest one wins. A person in none of the groups gets no access.
- A company account with the same name as a local hub user is refused until the local user is renamed or removed.
- The hub checks the ID token itself: the RS256 signature, issuer, audience, expiry and nonce, with PKCE and a one-time state.
- When everything works, set `"password_login": false`: only the company button remains. API tokens for MCP clients still work. Keep one local admin password in your vault, for the case that the identity provider is down.

Check: after `Restart-Service CompassHub`, "Sign in with your company account" leads to AD FS and back. hub.log shows "… signed in with a company account as architect", and the **Users** tab lists the company account with its role.

9. Step 7 – AI keys, Jira, Confluence and the team policy

Add the sections to `hub.json`. Every secret named here also needs its `…_FILE` variable (step 3). Then run `Restart-Service CompassHub`; the signed-in people stay signed in.

```
"ai": {
  "keys":   [ { "for": "role:architect", "secret": "claude_key_team" },
              { "for": "anna@company.com", "secret": "claude_key_anna" } ],
  "limits": [ { "for": "*", "tokens_per_day": 2000000, "tokens_per_month": 30000000 } ],
  "models": [ "claude-haiku-5-5", "claude-sonnet-5-5" ]
},
"integrations": {
  "jira":       { "url": "https://company.atlassian.net", "user": "svc-compass@company.com",
"project": "ARCH", "issue_type": "Task" },
  "confluence": { "url": "https://company.atlassian.net/wiki", "user": "svc-compass@company.com",
"space": "ARCH" }
},
"policy": {
  "min_approval": "ask",
  "ai_via_hub": true,
  "jira_via_hub": true,
  "confluence_via_hub": true,
  "web_access": false,
  "update_check": false
}
```

Setting	Meaning
`ai.keys`	Which Claude key a person uses. Order: the user name, then `role:<role>`, then `*`. No match means no AI for that person.
`ai.limits`	Tokens per day and per month. Above the limit, the hub refuses with a clear message.
`ai.models`	The allowed models; other models are refused.
`policy.ai_via_hub`	Compass asks the AI only through the hub: no API key on the workstations.
`policy.jira_via_hub`, `confluence_via_hub`	Jira/Confluence through the service account of the hub. A viewer may only read; every change is audited.
`policy.min_approval`	The weakest approval mode allowed in Compass for writes to the model (`ask` is the strictest).
`policy.web_access`, `web_allowed_domains`	Whether the assistant may search the web or read pages, and from which domains.
`policy.update_check`	Whether Compass checks GitHub every hour for a new version.

Admins change the policy later in the **Team policy** tab of the web view, and the usage per person is in **Users → AI usage**.

10. Step 8 – The workstations (Group Policy)

1. **Trust the certificate.** If the hub certificate comes from the company CA, deploy its root to the workstations (GPO: Computer Configuration → Policies → Windows Settings → Security Settings → Public Key Policies → Trusted Root Certification Authorities). Without it, Compass reports an untrusted certificate (UnknownIssuer).
2. **Templates.** Copy `LoonarCompass.admx` to `\\<domain>\SYSVOL\<domain>\Policies\PolicyDefinitions`, and `en-US\LoonarCompass.adml` (and `pl-PL` if needed) to the language subfolders. Without a central store, copy them to `C:\Windows\PolicyDefinitions` on the computer where you edit the GPO.
3. **GPO** for the workstation OU: Computer Configuration → Administrative Templates → Loonar → Loonar Compass.

Policy	Registry value (HKLM\SOFTWARE\Policies\Loonar\Compass)	Effect
Hub address	`HubUrl` (text)	The hub address is fixed in Compass and cannot be changed.
Require the hub	`RequireHub` = 1	The assistant works only while the person is signed in to the hub.
Web access	`WebAccess` = 0	No web search and no reading of pages; the 🌐 button is disabled.
Update check	`UpdateCheck` = 0	No daily check for a new version.
Minimum approval mode	`MinApproval` = ask / plan / auto	A weaker mode cannot be chosen.
Allowed AI providers	`AllowedProviders`, e.g. `anthropic`	Other providers (e.g. OpenAI) are refused.

Without a domain, the same values can be imported with `LoonarCompass-company-policy.reg` (and removed with `LoonarCompass-remove-policy.reg`). The computer policy is stronger than the hub policy and the user's settings; it can only make Compass stricter.

Installing Compass. `LoonarCompass-Setup-0.9.exe` installs for the current user (`%LOCALAPPDATA%\Programs\Loonar Compass`) or for all users (`Program Files`), and registers the add-in in Enterprise Architect. For deployment tools (Intune, Configuration Manager, a startup script) it installs without a window:

```
LoonarCompass-Setup-0.9.exe --quiet --all-users
```

Option	Effect
`--quiet`	No window and no questions. Exit code 0 = installed, 1 = error, 740 = `--all-users` without administrator rights. The result is in `%TEMP%\LoonarCompass-Setup.log` (for SYSTEM: `C:\Windows\Temp`).
`--all-users`	For everybody: Program Files, the common Start menu, the add-in in EA for all users (HKLM). Run it as SYSTEM or as an administrator.
`--dir <folder>`	Another program folder.
`--lang en|pl|de|fr`	The language of the licence and changelog files.
`--no-desktop`, `--no-start-menu`	Without these shortcuts.
`--no-ea-addin`	Without the add-in in Enterprise Architect.

The same command updates an existing installation. Close Compass and EA on the computer first. The user settings are not touched; with the computer policy (hub address, required hub) nobody needs to configure anything. Quiet removal: `"C:\Program Files\Loonar Compass\uninstall.exe" --uninstall --quiet`.

The user's first sign-in. In Compass, open Settings → Team (Compass Hub) → **Sign in with your company account**. The browser opens AD FS and returns to Compass. The sign-in lasts 12 hours and survives a restart of the hub.

11. Step 9 – Acceptance checklist

✓	Test	Expected
☐	Open `https://hub.company.com/` from a workstation	The sign-in page, no certificate warning
☐	Company sign-in as a member of Compass-Architects	Role architect; hub.log "signed in with a company account as architect"
☐	A person in none of the Compass groups	Refused
☐	Compass with the GPO, before the sign-in: send a question	"Your organisation requires the team hub: sign in to it in Settings → Team (Compass Hub)..."
☐	The same after the sign-in	An answer, with no API key on the workstation; a row in Users → AI usage
☐	A viewer asks a question (no key for viewers)	"the hub has no AI key for you"
☐	The 🌐 button in Compass	Disabled: "switched off by your organisation"
☐	`Restart-Service CompassHub`, then ask again	Works without signing in again
☐	Hub stopped, ask	A short message that the hub does not answer; EA keeps working
☐	Jira: "Create a Jira issue: test" (architect)	The issue in the project; an entry in the hub audit trail with the person's name

12. Operation: update, backup, logs, removal

Update

```
Stop-Service CompassHub
Copy-Item .\compass-hub.exe C:\CompassHub\compass-hub.exe -Force   # from the new zip, after checking
its SHA-256
Start-Service CompassHub
```

hub.json, the secrets and the sign-ins stay as they are. People remain signed in.

What to back up (C:\CompassHub)

File / folder	Content
`hub.json`	Configuration, local users (password and token hashes only), policy
`audit.jsonl`	The audit trail of changes and sign-ins
`changes\`, `notes\`	Change sets for review and the team notes
`ai_usage.jsonl`	AI usage per person (the base of the limits)
`company_accounts.json`, `sessions.json`	Company accounts seen and current sign-ins (hashes); can be lost without harm: people sign in again
`secrets\`	Only if your backup is encrypted. Otherwise keep the values in the password vault.
`models\`	The model files, if they are not kept elsewhere

Logs

- `hub.log`: start, sign-ins, refused requests, errors. An "internal error (id ...)" shown to a user has its details here under the same id.
- `audit.jsonl` and `hub.log` are renamed with a date when they grow large.

Removal

```
C:\CompassHub\compass-hub.exe --service-uninstall
```

Then remove the IIS site, the AD FS application group and the GPO, and delete `C:\CompassHub` after the backup.

13. Troubleshooting

Symptom	Cause and fix
Compass: certificate not trusted / UnknownIssuer; the browser shows "Not secure"	The workstation does not trust the root of the hub certificate. Deploy the root CA (step 8), then restart the browser and Compass.
Company sign-in ends at `https://hub.company.com/adfs/…` or with 404	ARR rewrites the redirect. Set `reverseRewriteHostInResponseHeaders` = False (step 5).
"<name> is not in any group of the hub (…): ask the administrator for access"	The person is in none of the role groups, or the `groups` claim is missing. Check the transform rule "Token-Groups - Unqualified Names → groups" and the `allatclaims` scope.
"A local user of the hub has the same name"	Rename or remove that local user in the Users tab.
Compass: "the team hub or the server in front of it does not answer properly (502)"	The service is stopped or starting: `Get-Service CompassHub`, then hub.log.
Compass: "your sign-in to the team hub has expired"	12 hours have passed, or sessions.json was lost. Sign in again in Settings → Team.
"the hub has no AI key for you"	No `ai.keys` entry for the person, their role or `*`.
"your AI limit for today is used up"	Raise `ai.limits` for the person or role, or wait for the next day.
"secret '…' is not set: give it as LOONAR_SECRET_…"	The `…_FILE` variable is missing in the service's `Environment`, or the file is empty or not readable by SYSTEM.
hub.log: "<address>: 20 wrong tokens, refused for 10 minutes"	20 **different** wrong tokens came from one address (an old token repeated counts once). Valid tokens still work. Find the client in hub.log.
Large questions fail with 413 or 404.13	`maxAllowedContentLength` in web.config (step 5).
Long answers end with 502.3 after 2 minutes	The ARR `timeout` (step 5).
AD FS: MSIS7628 from `Grant-AdfsApplicationPermission` right after installing AD FS	The farm is still starting: repeat the command a few minutes later.
hub.log / AD FS: MSIS9622 "Client authentication failed"	The secret file does not match AD FS. Make a new secret: `Set-AdfsServerApplication -TargetIdentifier <client id> -ResetClientSecret`, write it to the file, `Restart-Service CompassHub`.

Compass on a Windows Server (e.g. Remote Desktop Services): PDF export fails, or the first Office sign-in is blocked	Compass makes PDFs with Word and PowerPoint on the computer where it runs. On Windows Server, IE Enhanced Security Configuration can block the first Office sign-in (`odc.officeapps.live.com`): add the Office sign-in sites to the trusted sites, or turn the setting off for users in Server Manager.

14. Linux and Docker

The same hub runs in a container. The data folder (hub.json with `"listen": "0.0.0.0:8730"`, the models and the hub's files) is mounted as `/data` and must be writable by user 10001. The secrets come from `LOONAR_SECRET_*` variables or from Docker secrets through `LOONAR_SECRET_*_FILE`:

```
docker run -d --name compass-hub -p 127.0.0.1:8730:8730 -v /srv/compass-hub:/data \
  -e LOONAR_SECRET_CLAUDE_KEY_TEAM_FILE=/run/secrets/claude_key_team \
  ghcr.io/loonar-eu/compass-hub:0.9.0
```

Put nginx or another HTTPS proxy in front of it, as in step 5. The proxy must keep the `Location` header of redirects unchanged, pass `X-Forwarded-For`, allow 32 MB bodies and a 5-minute timeout. For SQL Server, MySQL or Oracle models, build your own image on this one and add the vendor's Linux ODBC driver.