Neuropacs™ Agent Usage Guide

Authored By: Kerrick Cavanaugh

Document Version History

Date Updated	Version	Notes
12/01/2025	v1.0.0	Initial release
03/02/2026	v1.0.1	Added “Quality Control” section to Settings.
03/12/2026	v1.0.2	Added Docker standalone installation method and Swagger page
04/01/2026	v1.0.3	Added Windows and virtualization support
04/08/2026	v1.0.4	Added builds for clinical use
04/14/2026	v1.0.5	Deprecate Linux script-based install
04/16/2026	v1.0.6	Added de-identification information and updated screenshots, UI auth, and password reset
05/12/2026	v1.0.7	Added SSO SAML authentication and configurable TLS certificates + RBAC

Table of Contents

Neuropacs Agent Description

The Neuropacs Agent is an on-premise software service designed to securely receive, validate, and transmit medical imaging studies from a PACS to the Neuropacs Cloud platform. It supports automated DICOM routing, quality control, manual study uploads, and optional local preprocessing, while maintaining HIPAA-compliant handling and transport of all imaging data. The Neuropacs Agent also provides an intuitive, modern interface for managing studies and overseeing the Neuropacs workflow.

The agent can be configured to operate in one of two modes:

Cloud Processing
Ingested imaging studies are validated on-premises through an initial quality control check and then securely transmitted to the Neuropacs Cloud platform for full processing.

Local Processing
Imaging studies are validated and preprocessed on-premises, enabling stricter control over the data transmitted outside the host facility. Full imaging datasets remain within the local data center, while only derived non-PHI features are securely forwarded to the cloud for further analysis.

System Requirements

Supported Environments

The platform is supported in environments that can run Linux-based x86_64 / AMD64 containers either natively or within a supported Linux virtual machine. This provides flexible deployment across Linux, Windows, macOS, on-premise virtualization platforms, and certain cloud environments while maintaining a consistent Linux-based runtime.

Supported Runtime Environments

The following deployment environments are supported:

- **Native Linux hosts**
  - Modern Linux distributions with support for Docker Engine and Docker Compose
  - Ubuntu 22.04 or later is recommended
- **Windows hosts**
  - Windows 10/11 Pro, Enterprise, Education, and Server with Hyper-V/VMware virtualization capabilities
- **macOS hosts**
  - Modern macOS versions with Docker Desktop
  - macOS systems using VMware Fusion

Supported Virtualization Platforms

The platform may also be deployed inside a supported Linux virtual machine on the following platforms:

- **VMware**
  - VMware Workstation (Windows/Linux)
  - VMware Fusion (macOS)
  - VMware vSphere / ESXi
- **Hyper-V**
  - Windows Server Hyper-V
  - Windows 10/11 Pro, Enterprise, and Education with Hyper-V enabled

Supported Cloud Deployment Targets

Virtual machine images may be imported for deployment in supported cloud environments, including:

- AWS
- Microsoft Azure
- Google Cloud Platform (GCP)

Supported CPU Architecture

The platform currently supports:

- **x86_64 / AMD64**

Other CPU architectures may be compatible only if they provide a supported virtualization or emulation mechanism capable of running x86_64 Linux guest environments or containers.

Software Requirements

- Docker Engine / Docker Compose
- Docker Desktop where applicable
- Docker version 2+

Important Compatibility Notes

- The application runtime is Linux-based in all supported deployment models.
- Windows and macOS are supported as host environments only and do not run the application as native Windows or macOS workloads.
- Native Windows-only container deployments are not supported.
- Linux container execution must occur either:
    - directly on a supported Linux host, or
    - within a supported Linux virtual machine
- Custom prebuilt or converted virtual machine images may be provided for:
    - VMware-compatible environments
    - Hyper-V-compatible environments

Hardware Requirements (Cloud Preprocessing)

- CPU: 2+ vCPUs
- RAM: 4+ GB
- Storage: 50-200+ GB recommended
    - Storage recommendations depend heavily on order volume. Contact Neuropacs support for assistance calculating recommended storage requirements.

Hardware Requirements (Local Preprocessing)

- Application
    - CPU: 2+ vCPUs
    - RAM: 4+ GB
- Processing pipeline
    - CPU: 6+ vCPUs / container
    - RAM: 8+ GB / container

- Storage: 50-200+ GB
    - Storage recommendations depend heavily on order volume. Contact Neuropacs support for assistance calculating recommended storage requirements.

It is highly recommended to use **high-performance SSD storage** for local processing installations, preferably on a dedicated processing volume. Avoid standard HDD or other lower-I/O storage, as image processing tasks perform many repeated reads and writes of large imaging files and intermediate outputs. Fast SSD storage helps reduce I/O bottlenecks and improves overall processing speed and consistency.

Agent Application Releases

Release Date	Version	Notes
12/01/2025	v0.1.x	Initial production release

Installation

Note: To install the Neuropacs Agent, you must first obtain an API key and service user credentials. Contact Neuropacs support for more information.

Linux/MacOS Installation

Prerequisites

Before installation, ensure the following are available:

- A supported **Ubuntu 20.04+ Linux host**
- Sufficient CPU, memory, and disk resources for the Docker containers
- Docker version 2+ installed and running
- Network settings and any site-specific configuration values required by your environment

Installation

1. **Authenticate with GitHub Container Registry (required a GHCR token)**

```
echo "$GITHUB_TOKEN" | [sudo] docker login ghcr.io -u neuropacman
--password-stdin
```

When prompted, enter your token provided by Neuropacs.

2. **Pull the Neuropacs Agent Docker image**

```
[sudo] docker pull ghcr.io/neuropacs/neuropacs-agent:latest
```

3. **Basic Usage**

```
[sudo] docker run -it --name neuropacs-app \
 --restart always
 --cpu-shares=2048
 -p <UI_PORT>:3001 -p <DICOM_PORT>:104 \
 -v /var/run/docker.sock:/var/run/docker.sock \
 -v <path_or_volume_name>:/data \
 -v <path_or_volume_name>:/usr/src/app/data \
ghcr.io/neuropacs/neuropacs-agent:latest
```

4. Options and Flags

Option	Description	Required	Default
`-v neuropacs-container-workspaces:/usr/src/app/processing`	Docker managed volume for local processing I/O. Must be passed exactly as defined in the Option column.	Yes **(in local-processing mode)**	N/A
`-v /absolute/path/to/certs:/data/certs:ro`	Docker bind mount for TLS certificates. Within this directory must include a **cert.pem** and **key.pem** file.	Yes **(in tls-enabled mode)**	N/A

5. Full Example

```
[sudo] docker run -it --name neuropacs-app \
 --restart always \
 --cpu-shares=2048 \
 -p 3001:3001 -p 104:104 \
 -v /home/ubuntu/neuropacs/certs:/data/certs:ro \
 -v /var/run/docker.sock:/var/run/docker.sock \
 -v neuropacs-config:/data \
 -v neuropacs-agent-data:/usr/src/app/data \
 -v neuropacs-container-workspaces:/usr/src/app/processing \
ghcr.io/neuropacs/neuropacs-agent:latest
```

6. Verify Installation

Confirm that:

- the required containers are running
- the application is reachable on the expected interface
- the environment-specific configuration has been applied successfully

Windows Installation

Prerequisites

Before installation, ensure the following are available:

- A supported **x86_64 / AMD64 Windows host**

- Sufficient CPU, memory, and disk resources for the virtual machine
- One of the following:
    - **VMware Workstation**
    - **Hyper-V**
    - **VMware vSphere/ESXi**
- The provided prebuilt virtual machine files
- Network settings and any site-specific configuration values required by your environment

Installation

1. **Open or Import the Virtual Machine**

Using VMware: Open your VMware environment and import or open the provided virtual machine files. Available formats include: **OVA**, **OFV**, and **VMDK**.

Using Hyper-V: Open Hyper-V Manager on the Windows host and import the provided **VHDX** file.

For local workstation deployments, use **VMware Workstation**. For managed server or enterprise virtualization environments, deploy the VM to **VMware vSphere / ESXi**.

2. **Review Virtual Machine Settings**

The provided virtual machine files are preconfigured with recommended resource allocation and networking configuration.

Before starting the VM, review and adjust the following as needed for your environment:

- **CPU allocation**
- **Memory allocation**
- **Disk location or datastore**
- **Network adapter / port group configuration**
- Any other VMware-specific settings required by your organization

Contact Neuropacs support for additional information on virtual machine configurations. Improper configurations can lead to unpredictable application behavior.

3. **Start the Virtual Machine**

Power on the virtual machine and confirm Linux guest operating system starts successfully.

4. **Access the Guest System**

Log into the guest operating system using the credentials and access method provided with the deployment package.

5. **Apply Environment-Specific Configuration**

Update any required configuration values for your environment, such as:

- hostnames
- network settings
- mounted storage paths
- application configuration values
- credentials or secrets, if applicable

To enable TLS, mount a path to your certificates and follow the installation procedure.

6. **Start the Application**

The virtual machine environment automatically provisions a **`neuropacs`** user account with the required permissions preconfigured. The user's home directory, **`/home/neuropacs`**, contains the following files:

1. **`README.md`** – comprehensive documentation covering application management and troubleshooting
2. **`init.sh`** – the initialization script used to begin the application setup process

From the neuropacs home directory, run:

```
./init.sh
```

Follow the on-screen prompts to configure the application. Once setup is complete, the application will start automatically.

7. **Verify Installation**

Confirm that:

- the virtual machine is running
- the required containers are running
- the application is reachable on the expected interface
- the environment-specific configuration has been applied successfully

Application Management

Docker-based Installation

Management is performed via Docker commands, allowing administrators to start, stop, restart, inspect, and monitor the application using standard Docker and Docker Compose workflows.

Example usage:

```
[sudo] docker start neuropacs-app                                  # Start all services

[sudo] docker stop neuropacs-app neuropacs-db                      # Stop all services

[sudo] docker restart neuropacs-app                                # Restart services

[sudo] docker ps -a \                                              # Display service status
    --filter name=neuropacs-app
    --filter name=neuropacs-db

[sudo] docker logs -f neuropacs-app                                # View service logs

[sudo] docker ps \                                                 # Show service Docker images
    --filter name=neuropacs-app --filter name=neuropacs-db \
    --format 'table {{.Names}}\t{{.Image}}\t{{.Status}}'

[sudo] docker rm -f neuropacs-app neuropacs-db && \                # Uninstall
    docker volume rm \
    neuropacs-data neuropacs-agent-data \
    neuropacs-db-data neuropacs-container-workspaces

[sudo] docker run --rm neuropacs/neuropacs-agent help              # Show help message
```

Data Collection

Overview

The Neuropacs Agent continuously collects operational telemetry to ensure the secure, reliable, and compliant operation of the system. These metrics are used exclusively for system health monitoring, performance optimization, fault detection, and regulatory compliance.

Collected Metrics

The following categories of operational metrics are collected by the Neuropacs Agent:

Runtime Metrics

Metric	**Description**
Agent ID	Logical Agent identifier (UUID)
Runtime Type	Runtime of the Agent (i.e., Docker, Singularity, Native)
Application Version	Running agent version

System Metrics

Metric	**Description**
CPU Core Count	Number of logical CPU cores available
CPU Load	Current system CPU utilization
Total Memory	Total physical memory available
Available Memory	Unused system memory available
Disk Free Space	Available disk storage on the working volume
Disk I/O Activity	Aggregate read/write activity indicators

Container and Service Health

Metric	**Description**
Container Status	Runtime state of each Neuropacs service container

Container Uptime	Duration each service has been continuously running
Service Availability	Health status of critical processing and API services

Processing and Workflow Metrics

Metric	**Description**
Order Status	Status of executed orders
Processing Queue	Local processing queue metrics (if applicable)

Metrics are strictly limited to infrastructure-level operational information and do not contain patient identifiers, clinical imaging data, diagnostic results, or host system information. All telemetry communications are encrypted in transit and authenticated.

De-identification and PHI Handling

The Agent implements HIPAA Safe Harbor de-identification applied to DICOM files before they leave the local environment. NIfTI datasets are not deidentified and are processed as is.

Architecture and Data Flow

1. **Workflow initiation**
   The de-identification process begins when a study enters the processing workflow. This applies to both the standard processing path and any alternate workflow path that has been explicitly permitted by system configuration.
2. **Input and output segregation**
   Source imaging is read from the study's ingestion location and written to a separate de-identification output location. This separation ensures that original received data and de-identified output remain logically distinct throughout processing.
3. **Controlled concurrency**
   De-identification is performed in parallel across multiple files to improve throughput. Concurrency is limited by system configuration to maintain predictable resource usage and operational stability.
4. **Write safety and consistency**
   Output files are written using a safe file-handling approach designed to prevent incomplete or partially written files from appearing in the final output set. If a processing error occurs, incomplete de-identified output for the study is removed so that no inconsistent dataset is retained.
5. **Failure handling policy**
   De-identification is enforced at the study level as a strict pass/fail operation. Failure to successfully de-identify any required file causes the study to fail this stage of processing, and the study does not proceed further in the workflow.
6. **Post-processing validation**
   After de-identification completes, the output dataset is validated against the expected source dataset to confirm completeness and integrity. This validation ensures that the de-identified output contains the required files and does not contain missing, unexpected, or invalid files.
7. **Processing record generation**
   Upon successful completion, the system may generate a manifest or comparable processing record to document the resulting de-identified file set and support downstream traceability, auditing, and operational review.

Scrubbed Fields by HIPAA Safe Harbor Category

Category 1 – Names (deleted)

- (0010,0010) Patient Name
- (0010,1001) Other Patient Names
- (0010,1005) Patient Birth Name
- (0010,2297) Responsible Person
- (0010,2298) Responsible Person Role
- (0008,0090) Referring Physician Name
- (0008,1048) Physician of Record
- (0008,1050) Performing Physician Name
- (0008,1060) Name of Physician Reading Study
- (0008,1070) Operator Name
- (0032,1032) Requesting Physician
- (0040,0006) Scheduled Performing Physician Name

Category 2 – Geographic Subdivisions (deleted or masked)

- (0010,1040) Patient Address - masked if maskZipCodes is enabled (first 3 ZIP digits kept, last 2 zeroed); otherwise deleted
- (0010,1080) Military Rank
- (0010,1081) Branch of Service
- (0010,1090) Medical Record Locator
- (0008,0080) Institution Name
- (0008,0081) Institution Address
- (0008,1010) Station Name
- (0008,1040) Institutional Department Name
- (0032,1060) Requested Procedure Location
- (0008,0092) Referring Physician Address

Category 3 – Dates (shifted or deleted)

When shiftDates is enabled (default), all DA/DT-type fields are shifted by a random offset (±dateShiftRangeDays, default ±365 days). TM-only (time) fields are preserved when date shifting is active. When shifting is disabled, all date and time fields are deleted.

Date-shiftable tags:

- (0008,0020) Study Date, (0008,0021) Series Date, (0008,0022) Acquisition Date, (0008,0023) Content Date
- (0008,002A) Acquisition DateTime
- (0010,0030) Patient Birth Date
- (0032,1000/1010) Scheduled Study Start/Stop Date
- (0040,0002/0004) Scheduled Procedure Step Start/End Date
- (0040,0244/0250) Performed Procedure Step Start/End Date

Time-only tags (deleted only when shiftDates is off):

- (0010,0032) Patient Birth Time
- (0008,0030-0033) Study/Series/Acquisition/Content Time
- (0032,1001/1011) Scheduled Study Start/Stop Time
- (0040,0003/0005/0245/0251) Procedure Step Start/End Times

Categories 4-6 – Contact Information (deleted)
- (0010,2154) Patient Telephone Numbers
- (0010,2155) Patient Telecom Information
- (0008,0094) Referring Physician Telephone Numbers

Categories 7-13 – Identifiers & Numbers (deleted)
- (0010,0020) Patient ID
- (0010,1000) Other Patient IDs
- (0010,1002) Other Patient IDs Sequence
- (0010,0021) Issuer of Patient ID
- (0020,0010) Study ID
- (0018,1000) Device Serial Number
- (0040,2016) Placer Order Number
- (0040,2017) Filler Order Number
- (0008,0050) Accession Number – conditionally deleted based on sendAccessionNumber config (deleted by default)
- (0018,1020) Software Versions – conditionally deleted based on removeSoftwareVersions config (preserved by default for downstream converter compatibility)

Categories 14-15 – URLs & IP Addresses (regex-scrubbed from text fields)
Detected and replaced with [URL_REMOVED] / [IP_ADDRESS_REMOVED] via regex in free-text fields.

Categories 16-18 – Biometric/Photo/Other Identifiers
Private DICOM tags removal is coded but currently commented out (ds.remove_private_tags() is disabled). Free-text fields are regex-scrubbed for embedded identifiers.

UID Replacement
All instance UIDs are regenerated:
- StudyInstanceUID: One new UID generated per study (shared across all files)
- SeriesInstanceUID: Mapped 1:1 from original → new UIDs, preserving series grouping
- SOPInstanceUID: Unique new UID per file
- MediaStorageSOPInstanceUID: Updated to match new SOP UID

Regex Scrubbing of Free-Text Fields
The following DICOM text fields are scanned for embedded identifiers:

- (0020,4000) Image Comments
- (0032,4000) Study Comments
- (4008,4000) Results Comments
- (0008,103E) Series Description
- (0008,1030) Study Description
- (0040,2001) Reason for Imaging Service Request

Patterns detected
SSN, phone numbers, email addresses, URLs, IP addresses, medical record numbers (MR/MRN/RN-NNNN), account numbers (ACCT/ACC/ACCOUNT-NNNN). Matches are replaced with [PATTERN_REMOVED] placeholders.

Preserved Fields
• Patient Age and Patient Sex – retained for clinical analysis accuracy
• Transfer Syntax – preserved (or inferred from encoding flags if missing) to maintain DICOM encoding integrity
• Pixel Data – untouched

See de-identification configurations in the **Configuring the Agent** section below.

Usage

Configuring the Agent

To configure the Neuropacs Agent to align with your custom workflow, access the integrated web portal available on port **3001**. When accessing the portal locally on the host system, navigate to:

```
http(s)://localhost:3001/
```

If accessing the portal from another machine on the network, you must have network access to the host system and use its IP address, for example:

```
http(s)://<host_ip_address>:3001/
```

If the installation was successful, you should now see the following screen:

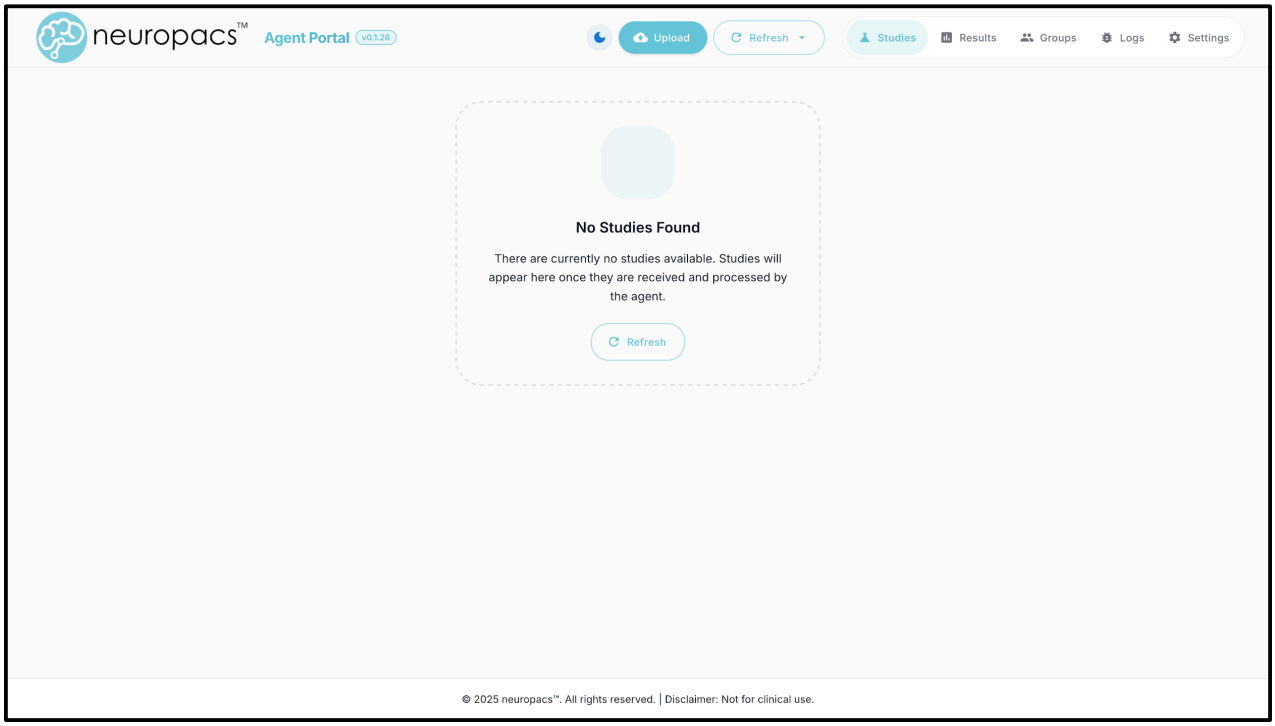

Figure 1: Integrated UI Studies page

Navigate to the **Settings** tab within the user interface.

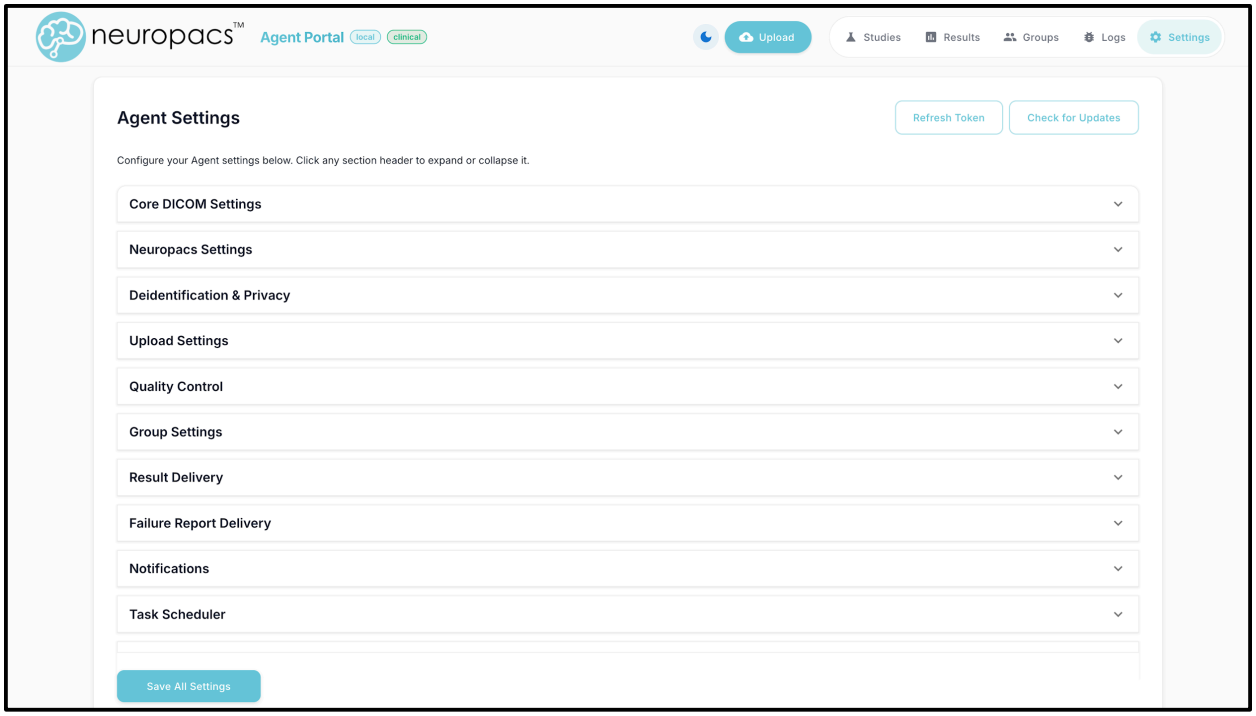

Figure 2: Settings page

Action Buttons:

- **Refresh Token:** Retrieves a new API access token using the current credentials. Refreshed the token cache with a new token.
- **Check for Updates:** Check if a Neuropacs Agent update is available. Updates handle application updates as well as associated Docker image updates.

Core DICOM Settings:

The Neuropacs Agent uses the configured DICOM settings (AE Title, Host, and Port) to securely receive imaging studies from PACS and other modalities. These values must match the corresponding routing configuration on the sending system.

- **AE Title:** A unique identifier used to register the Neuropacs Agent within the DICOM network for communication and routing.
    - **Recommended value: NEUROPACS_SCP**
- **DICOM Host:** The hostname or IP address where the Neuropacs Agent will receive DICOM connections. This value should reference the system on which the Agent is installed. This value cannot be modified in the application for Docker-native builds.
    - **Recommended value: localhost**
- **DICOM Port:** The TCP port on which the Neuropacs Agent listens for incoming DICOM connections. This port must be accessible to connected modalities or

PACS systems. This value cannot be modified in the application for Docker-native builds.
    - **Recommended value: 104**
- **Select Product(s):** Specify one or more Neuropacs processing modules to be executed by the Agent for DICOM-routed workloads.
- **DICOM TLS:** Enable TLS (1.2+) encryption for incoming DICOM connections. Sending PACS must be configured to use DICOM TLS.
    - Requires a TLS certificate and TLS private key with an optional CA certificate
    - When DICOM TLS is enabled, sending PACS systems must be configured to connect using TLS. Non-TLS connections will be rejected.

HTTPS / TLS:

The Neuropacs Agent supports TLS encryption for the integrated web UI. This section allows you to rotate or update the TLS keys and certificates used to secure browser sessions with the Agent's local web interface.

- **TLS Certificate:** A public certificate used to identify the server and enable encrypted HTTPS/TLS communication.
- **TLS Private Key:** The private key paired with the TLS certificate. It must be kept secure and is used to prove the server owns the certificate.
- **CA Certificate:** A certificate from the Certificate Authority that validates or chains trust to the TLS certificate.

Neuropacs Settings:

The Neuropacs settings define how the Agent connects to the Neuropacs Cloud platform, including API access and authentication. These values must be configured correctly to enable secure data transmission and processing.

- **Neuropacs API URL:** The endpoint URL used by the Neuropacs Agent to communicate with the Neuropacs Cloud platform. This value must be reachable from the host system for successful data transfer and processing.
- **Service User:** The username of the service account provisioned for authentication to the Neuropacs Cloud. This account is used exclusively for system-to-system communication.
- **Service Password:** The password associated with the provisioned service account used to authenticate the Neuropacs Agent with the Neuropacs Cloud.

To reset the service user password, access the Neuropacs Web Portal at https://d1nxuh43hp41jj.cloudfront.net/web-portal/**.** On the sign-in page, select **Sign In Securely**, then choose **Forgot your password?** and follow the on-screen

instructions. A verification code will be sent to the administrator email address associated with the account.

Password requirements

- Minimum of 15 characters
- At least one lowercase letter
- At least one uppercase letter
- At least one number
- At least one special character

Once your password has been successfully reset, replace the **Service Password** in **Neuropacs Settings** to finalize the password reset process.

SAML SSO + RBAC:

Configure SAML 2.0 Single Sign-On to allow users to authenticate via your organization's identity provider (e.g. Microsoft Entra ID, ADFS). When disabled, only the service user password is used.

- **Enable SAML SSO:** Master toggle. When off, SAML is completely disabled and only local password authentication is used.
- **Authentication Mode:** Controls which login methods are shown on the login screen. "Local Password Only" disables SSO. "SAML SSO + Local Password" shows both the SSO button and the password form. "SAML SSO Only" hides the password form entirely.

Identity Provider (IdP) Settings

- **IdP Entity ID:** The unique identifier (Issuer) of your identity provider. In Entra ID this is the "Azure AD Identifier"; in ADFS it is the Federation Service Identifier.
- **IdP SSO URL:** The SAML 2.0 Single Sign-On endpoint the Agent redirects to when a user clicks "Sign in with SSO".
- **IdP Certificate (PEM):** The IdP's X.509 signing certificate in PEM format. Used to verify the digital signature on SAML assertions. Paste the full certificate including -----BEGIN CERTIFICATE----- and -----END CERTIFICATE----- headers.
- **IdP Metadata URL:** Optional. The URL to the IdP's federation metadata XML document. Click "Fetch" to auto-populate the IdP Entity ID, SSO URL, SLO URL, and certificate from the metadata.

Service Provider (SP) Settings

- **SP Entity ID:** The identifier this Agent presents to the IdP. Must exactly match the "Identifier (Entity ID)" configured in the IdP's SAML application. Default: neuropacs-agent.

- **SP Base URL:** The public HTTPS URL users use to access this Agent. Used to construct the ACS URL and SP metadata URL.
- **NameID Format:** The format the Agent requests for the user identifier in the SAML assertion. "Email Address" is recommended for most deployments. Must match the NameID format configured in the IdP.
- **Require signed assertions:** When on, the Agent rejects SAML assertions that are not digitally signed by the IdP. Recommended to leave enabled for security.

Role-Based Access Control (RBAC)

- **Admin Group:** IdP group name or Object ID whose members receive admin rights (e.g. 'NeuropacsAdmins' or an Azure AD group ID)
- **User Group (optional):** IdP group name or Object ID for standard users. If left blank, all authenticated SAML users who are not in the Admin Group receive standard access.

Deidentification and Privacy Settings:

The Neuropacs Agent applies HIPAA Safe Harbor–compliant de-identification procedures, removing protected health information (PHI) elements prior to transmission. Only non-identifiable data required for analysis is sent to the Neuropacs Cloud. The Agent also sends a periodic heartbeat message to the Neuropacs Cloud for monitoring of registered Agents. The content of these heartbeat messages can be configured to include only the desired information.
DEIDENTIFICATION SETTINGS ONLY AVAILABLE FOR CLOUD PROCESSING ENABLED SYSTEMS.

- **Date Handling Strategy:** Defines how date information is managed during de-identification. The Agent can either shift dates to preserve relative timelines while removing identifiable calendar references, or remove dates entirely for maximum privacy protection.
    - **Recommended value: Shift dates**
- **Shift Date Range:** Specifies the range of days used when shifting dates during de-identification. Dates are randomly adjusted within this range to preserve relative timelines while preventing re-identification.
- **Mask Zip Codes:** Controls whether ZIP codes are masked or generalized to protect patient identity.
    - **Recommended value: ON**
- **Send Accession Number to Neuropacs™ Cloud:** Controls whether the DICOM Accession Number is included in uploads to the Neuropacs Cloud. Useful for tracking, but may introduce identifying information.

- **Include System Metrics (CPU, Memory, Disk):** When enabled, system resource usage metrics are included in heartbeats for monitoring and troubleshooting.
- **Include Order Statistics:** When enabled, study/order counts (processed, failed, in-progress) are included in heartbeats.
- **Include Health Status:** When enabled, application and database health status are included in heartbeats. [RECOMMENDED]

Results Settings:

Manages visibility of completed studies in the Results page

- **Show External Orders in Results:** Displays completed orders from the neuropacs™ Cloud that were not initiated by this Agent but are associated with the same organization as the service user.

Upload Settings:

Controls how the Neuropacs Agent manages the transfer of imaging data to the neuropacs™ Cloud, including retry behavior and upload reliability. **Maximum upload size:** 5GB.

- **Maximum Upload Retries:** Specifies how many times the Agent will retry an image upload if it fails.

Quality Control Settings:

Controls what quality control checks are enforced by the Agent for incoming imaging. Disabling quality control checks can allow incomplete or non-compliant studies to continue processing. Use only when you intentionally accept this risk. Bypassed checks will be marked as bypassed in the QC report and treated as passing.

- **DICOM QC Checks:** Toggle QC checks for DICOM-based datasets.
- **NIfTI QC Checks:** Toggle QC checks for NIfTI-based datasets.

Group Settings:

The Neuropacs Agent supports grouping of studies into cohorts or project-defined collections, even when the studies originate from different subjects. These settings define how the Agent organizes, processes, and manages operations for studies that are associated with the same group.

- **Auto-Assign DICOM Uploads to Group:** All newly received DICOM studies will be automatically assigned to a specified group upon ingestion.
- **Automatic Report Sending:** Controls how group reports are configured to automatically send to a destination.
    - **Auto-Send Report When Group Completes:** When enabled, reports are automatically generated and sent once all orders within a group

have reached a terminal state. Additional studies may be added to the group after completion, in which case a new report will be generated that includes the newly added studies.
    - **Send Group Reports on a Scheduled Interval:** Select how frequently group reports should be sent automatically, even when no new study completions have occurred. Multiple days may be selected for recurring weekly delivery, or you may modify the cron expression directly to define a custom scheduling pattern.
- **Group Report Format:** Define the format of group reports, including the report type and filename to be generated and sent.
    - **Report Type:** Choose an available report type
        - **Prediction Indexes:** A CSV file containing the aggregated prediction indices generated by each selected product for every study within the group.
    - **Filename Template:** Define a filename template for group reports. The format supports the following placeholders: `{group_name}`, `{report_type}`, and `{date}`. Placeholders will be auto-populated prior to submission.
- **Group Report Delivery:** Specify the destination to which group reports will be delivered. **TIP:** Test your selected delivery method using the **"Test Delivery"** button.
    - **Delivery Method:** Supports email, REST API/webhook and None (manual download only).

Notification Settings:

Defines how the Neuropacs Agent delivers event-based notifications to the configured destination.

- **Delivery Protocol:** Controls the protocol used to route reports to external systems and the destination parameters. Supports email and REST API/webhook. **TIP:** Test your selected delivery method using the **"Test Notification"** button. The following events are available:
    - **New Dataset Received**
    - **New Dataset Received**
    - **Quality Control Passed**
    - **Quality Control Failed**
    - **Upload Failed**
    - **Upload Succeeded**
    - **Report Delivered Successfully (Report Available)**
    - **Report Delivery Failed**
    - **Processing Failed (Failure Report Available)**
    - **Group Processing Completed**

- **Group Report Sent Successfully**
- **Group Report Send Failed**
- **Local Processing Completed**
- **Local Processing Failed**

Result Delivery Settings:

Defines how the Neuropacs Agent delivers and routes individual analysis result reports to external HIS systems. Up to 5 destinations can be configured. Deliveries are attempted 3 times before failing. Reports remain available for manual delivery or download.

- **Notification Protocol:** Controls the protocol used to route notifications and the destination parameters. **TIP:** Test your selected delivery protocol using the **"Test Delivery"** button.
  - **DICOM Encapsulated PDF:** Embeds the report within a DICOM object and transmits it using standard DICOM messaging.
  - **DICOM Structured Report (SR):** Embeds the report content within a DICOM SR object and transmits it using standard DICOM messaging.
  - **REST API:** Reports are sent base-64 encoded as part of an HTTP POST or PUT request to a configured REST endpoint in the chosen format. Basic and token auth are supported. The request body is configurable using templated strings.
  - **Email:** Reports are sent as an email attachment to a set of email addresses (comma-separated). The email subject and body are configurable using templated strings.
  - **None:** Reports are not transmitted to any external systems and made available for manual download.

Failure Report Delivery Settings:

Defines how the Neuropacs Agent delivers and routes individual failure reports. Deliveries are attempted 3 times before failing. Reports remain available for manual delivery or download.

- **Notification Protocol:** Controls the protocol used to route notifications and the destination parameters. . **TIP:** Test your selected delivery protocol using the **"Test Delivery"** button.
  - **REST API:** Failure reports are sent base-64 encoded as an HTTP POST or PUT request to a configured REST endpoint in the chosen format. Basic and token auth are supported. The request body is configurable using templated strings.
  - **Email:** Failure reports are sent as an email attachment to a set of email addresses (comma-separated). The email subject and body are configurable using templated strings.

    - **None:** Failure Reports are not transmitted to any external systems and made available for manual download.

Scheduler Settings:

Controls when automated management tasks are executed by the Neuropacs Agent, including scheduled processing, cleanup, and maintenance activities. The frequency of these task executions determines how often the Agent communicates externally with the Neuropacs Cloud.

- **Available options:**
    - Every minute
    - Every 5 minutes
    - Hourly
    - Daily at midnight
    - Daily at 3:00 AM
    - Weekly on Sunday at 3:00 AM

Local Processing Settings:

ONLY AVAILABLE FOR LOCAL PROCESSING ENABLED SYSTEMS. Specifies how the Neuropacs Agent manages local preprocessing workloads, including when preprocessing is performed and how computational resources are utilized.

- **Enforce CPU and Memory Limits:** When enabled, local worker containers run with the configured CPU core and memory limits below. Disable to let containers run without explicit CPU/memory caps.
- **Container Configuration:** Determines how containers are configured for local preprocessing.
    - **Maximum Concurrent Containers:** Defines the maximum number of preprocessing containers that may run simultaneously.
    - **CPU Cores per Container:** Specifies the number of CPU cores allocated to each preprocessing container.
    - **Memory Limit per Container:** Specifies the amount of memory (in GB) allocated to each preprocessing container.

Data Retention Policy:

Defines how the Neuropacs Agent manages data retention policies, including the duration under which studies, logs, and related artifacts are stored or removed. All associated PHI stored for a study is purged when the associated order is deleted.

- **Order Retention Period:** Specifies the number of days that order-related information is retained.
- **Log File Retention Period:** Specifies the number of days that log data is preserved.

Configuring a PACS to Send Studies to the Neuropacs Agent

To allow your PACS to route studies to the Neuropacs Agent, you must create a new **DICOM destination** in the PACS that points to the Agent's **Storage SCP** service. This destination tells the PACS where to send imaging studies over DICOM.

1. **Gather Neuropacs Agent DICOM Settings**

Before configuring the PACS, open the **Settings page or virtual machine network configuration** and record the following:

```
AE Title:        NEUROPACS_SCP
DICOM Host:      <agent_ip_address>
DICOM Port:      <agent_port>
```

Example:

```
AE Title:        NEUROPACS_SCP
DICOM Host:      10.10.20.50
DICOM Port:      115
```

These values identify the Neuropacs Agent on your network. In a virtual machine deployment, the Agent's reachable IP address and port must be determined from the VM's network configuration. If your environment requires use of a non-standard DICOM port, configure the appropriate network translation or port-forwarding rules so inbound DICOM traffic is correctly routed to port `104` inside the VM, where the Dockerized application is listening.

If TLS will be used, also confirm:

- TLS is enabled on the Neuropacs Agent
- The server certificate that the PACS should trust
- Any required client certificate settings, if mutual TLS is enabled

Only **TLS 1.2+** is supported. When DICOM TLS is enabled, sending PACS systems must be configured to connect using TLS. Non-TLS connections will be rejected.

2. **Create a New DICOM Destination in the PACS**

Log in to your PACS administration interface and create a new outbound DICOM destination. When prompted, enter the Neuropacs Agent values exactly as configured in the Agent.

3. **OPTIONAL - Configure TLS**

If your PACS and environment require encrypted DICOM communication, configure TLS for the new destination before testing connectivity.

In the destination or DICOM security settings for the new node:

- Enable TLS or Secure DICOM
- Select the appropriate certificate trust configuration
- Import the Neuropacs Agent certificate or the issuing CA certificate if required
- Configure a client certificate if the environment uses mutual TLS
- Ensure the PACS is pointed to the correct secure DICOM port

TLS configuration is not standardized across PACS user interfaces. Some PACS platforms only require enabling TLS and trusting the server certificate, while others require full certificate import, cipher configuration, or mutual certificate validation.

If TLS is enabled on one side but not the other, the connection test will fail.

4. **Save the configuration.**

After entering the connection details, save the new DICOM destination in the PACS. Once saved, the PACS should recognize the Neuropacs Agent as a valid DICOM routing target.

5. **Test the Connection**

Most PACS systems provide a DICOM Echo / C-ECHO test.
Run:

- Ping or Verify
- Test Connection
- C-ECHO

If TLS is enabled, this test validates both:

- network connectivity to the Agent
- successful TLS negotiation between the PACS and the Agent

Navigating the UI

Navigation Menu

This menu allows the user to:

- Navigate between pages in the interface
- Toggle dark mode
- Initiate a manual upload
- Refresh the page or enable automatic refresh

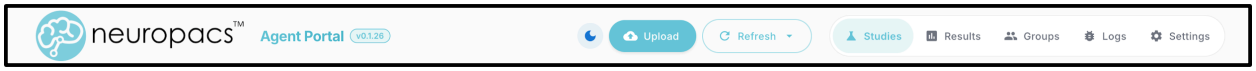

Figure 3: Navigation menu

Viewing Studies

Access the **Studies** page to view all imaging studies ingested by the Neuropacs Agent. Studies appear automatically once they are received and validated. Newly received studies will initially display **"No"** under the **Stable** column until processing completes and the study is marked as stable.

The **Actions** column provides the following options:

1. **Check Status** – Displays the current processing status (enabled after processing begins)
2. **View Quality Control Results** – Opens the QC details for the study
3. **Preview Report** – Preview PDF reports for completed/failed studies
4. **Cancel Order** – Cancels the study's processing request
5. **View Study Information** – Shows metadata and study details
6. **Assign to Group** – Assign study to a group

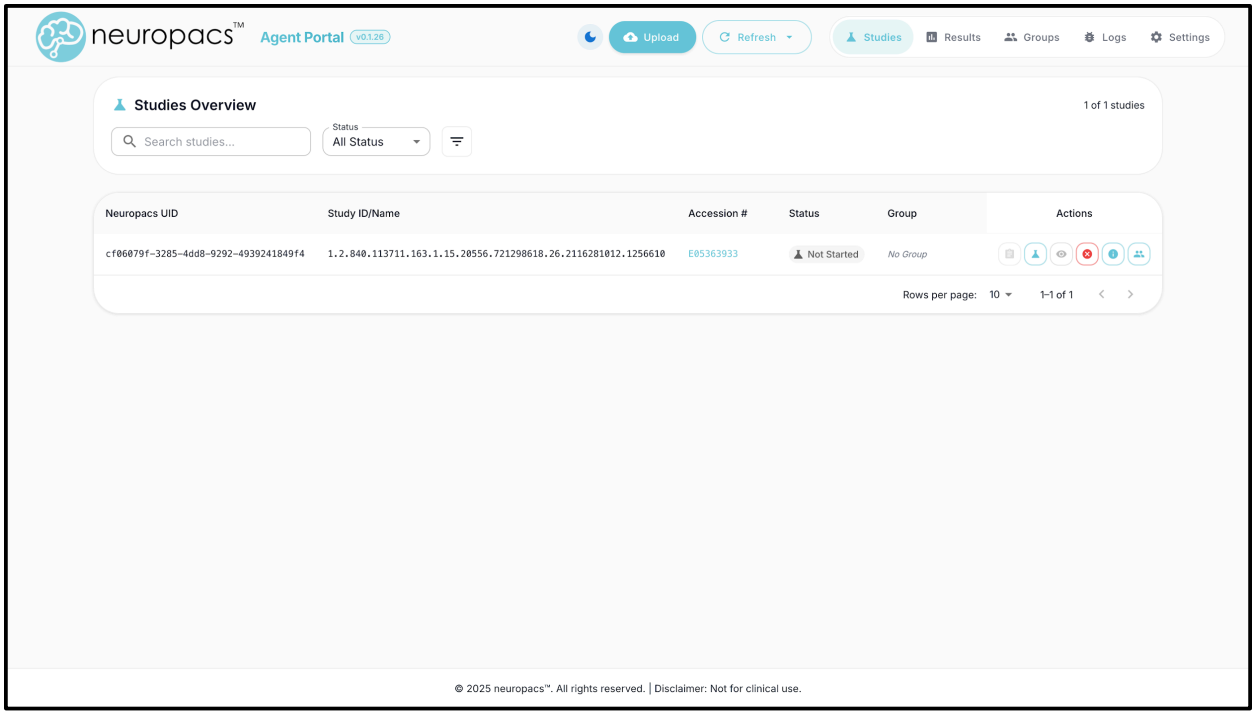

Figure 4: Unstable study on Studies page

View Additional Information

To view additional information for a study/order, click the **Clipboard** button in its corresponding row.

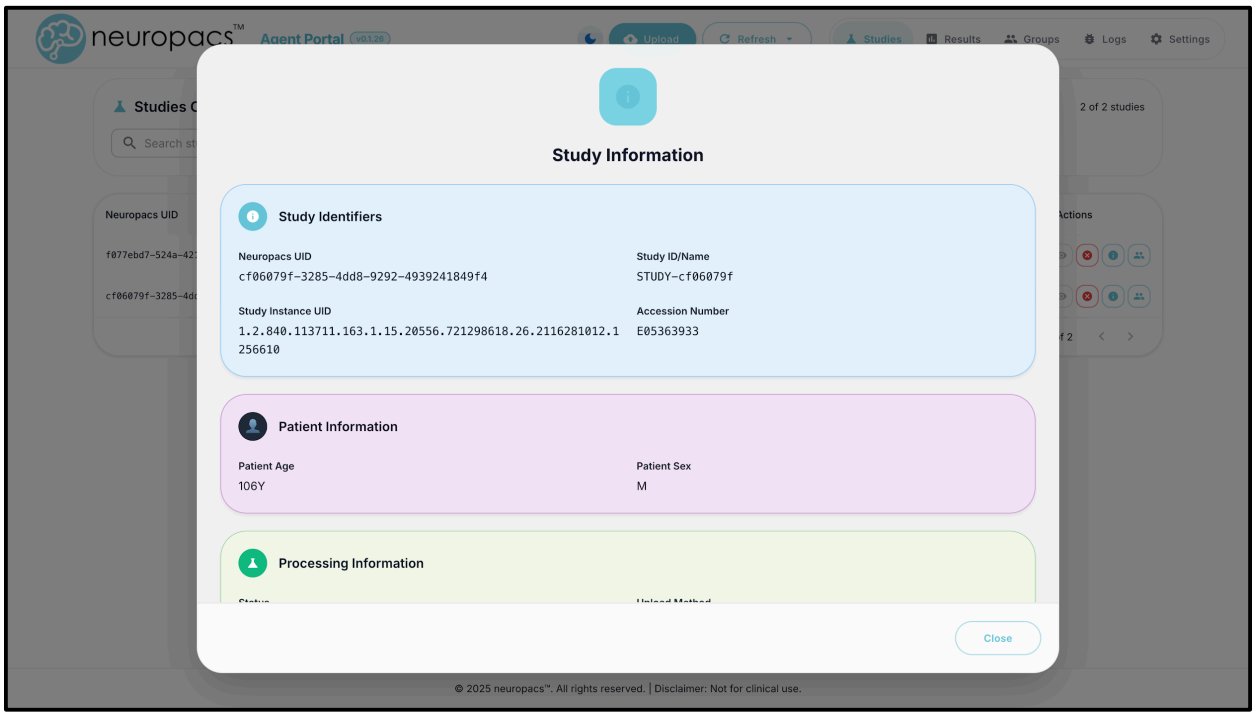

Figure 5: Study information popup

Quality Control Reports

After a study received through the DICOM SCP interface becomes stable, it is automatically evaluated by the Neuropacs quality control system to ensure that only appropriate datasets proceed to processing. To view the quality control results for a study, click the **Eye** button in its corresponding row.

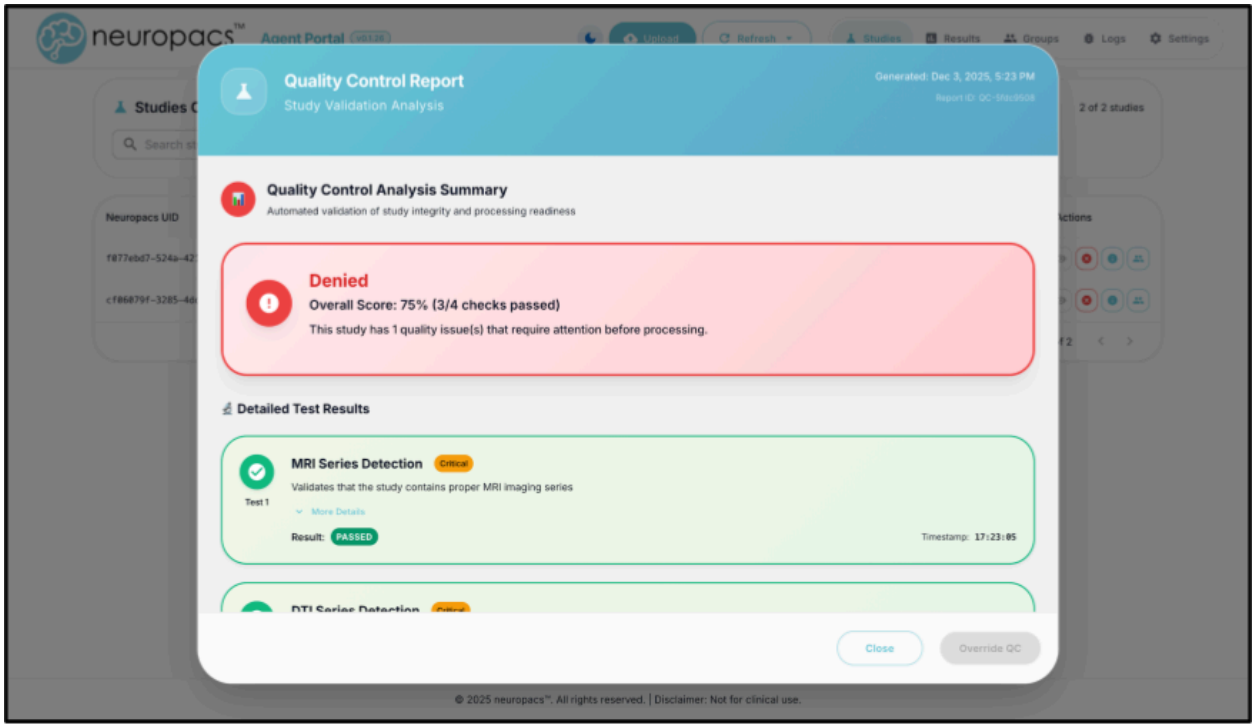

Figure 6: Quality control report 1/2

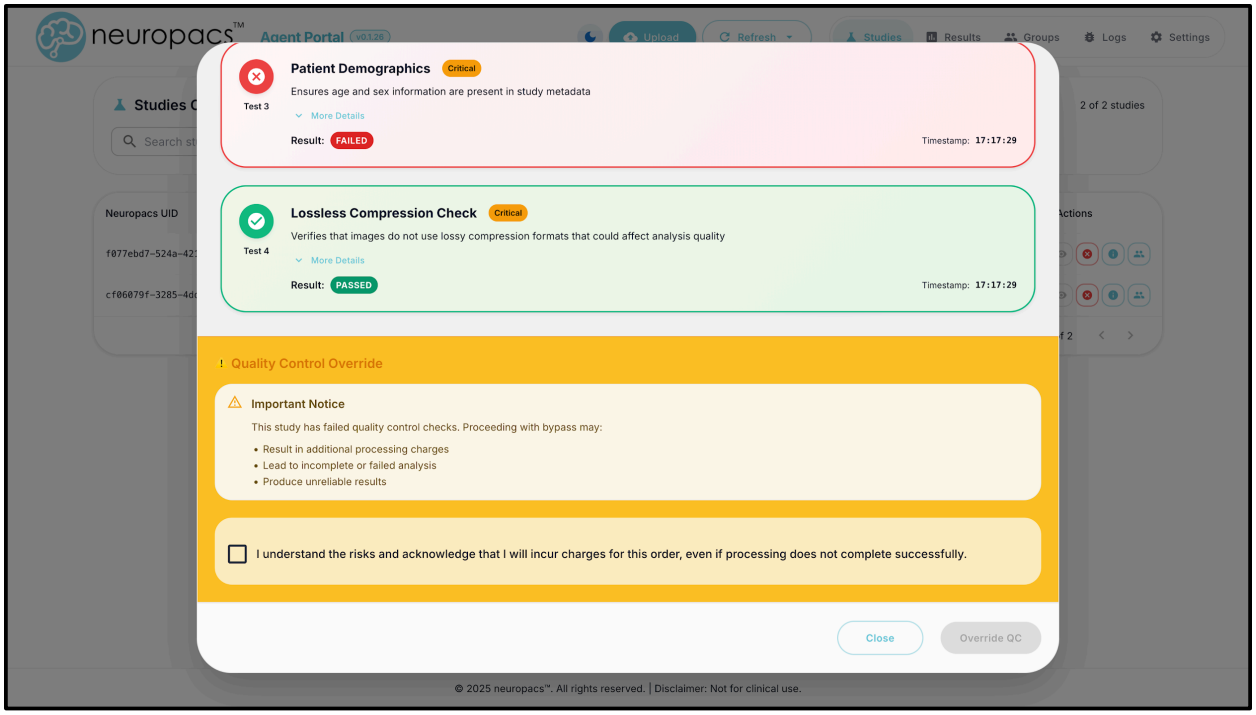

Figure 7: Quality control report 2/2

A failed quality control check may be bypassed when necessary; however, this should be done with caution.
Warning: Bypassing QC can result in additional processing charges, incomplete or failed analyses, or unreliable results.

Group Assignment

To manually assign a study to a group, select the **Group** button in the corresponding row. After assignment, a group tag will appear in that row under the *Group* column.

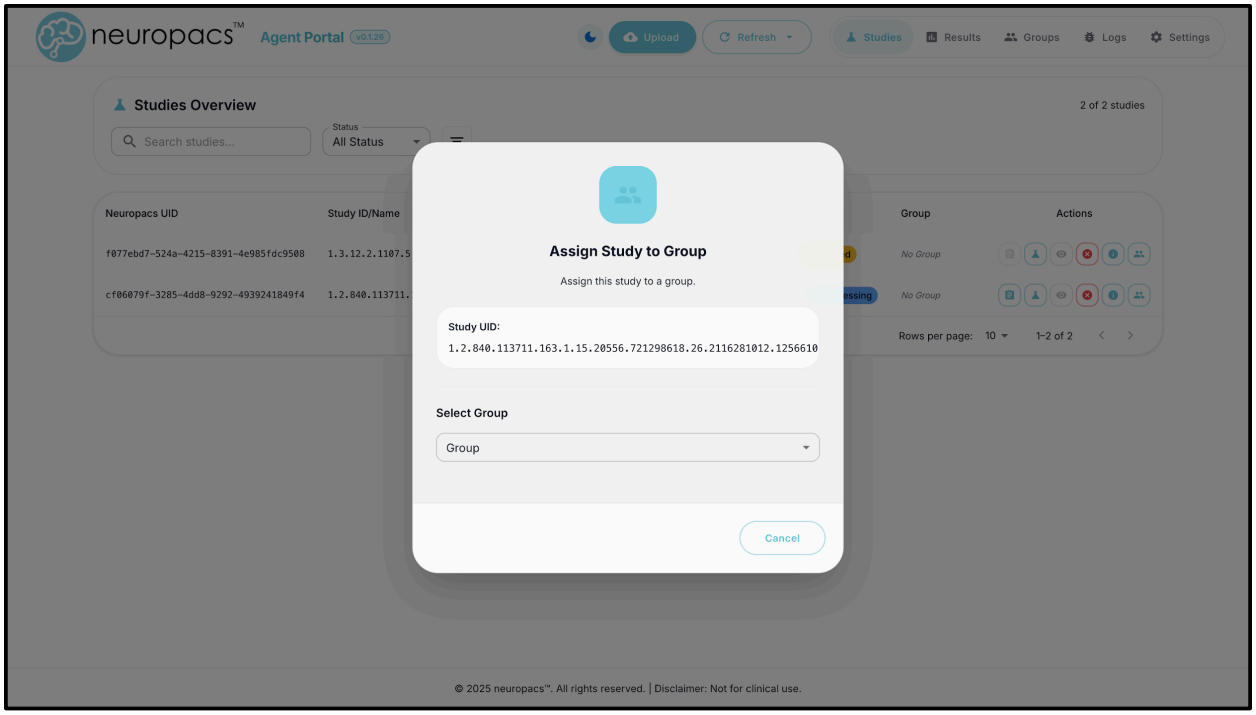

Figure 8: Group assignment dialog

Manual Uploads

To manually upload an order, click the **Upload** button in the navigation bar. Then follow the steps below to submit a new study:

1. **Study Name (Optional):** Provides a human-readable identifier for the study and is combined with a Neuropacs-generated UUID to form a unique reference. The name must be 5–50 characters and may include letters, numbers, spaces, slashes (/), colons (:), and hyphens (-).
2. **Accession Number (Optional):** Enter an accession number if available. This field supports downstream tracking and reconciliation with clinical systems.
3. **Patient Age (Optional):** Enter the patient's biological age associated with the imaging study.
4. **Patient Sex (Optional):** Enter the patient's biological sex associated with the imaging study.
5. **Assign to group (Optional):** Assign the study to a group. If selected, the product selection will be autopopulated with the products of that group.
6. **Select Product(s):** Select one or more products to be run on the uploaded study.
7. **Select Upload Type:** Choose the type of study data being uploaded. Available options include:

- **Folder:** A folder containing DICOM files with optional JSON sidecar, or NIfTI/BVEC/BVAL files with required JSON sidecar.
- **ZIP Archive:** A single ZIP archive containing DICOM files with optional JSON sidecars, or NIfTI/BVEC/BVAL files with required JSON sidecar

8. **Select Upload Button:** Start the upload process. A progress bar will appear that will show the progress of the upload. Do not navigate away or close the window during an upload as this may corrupt the upload process.

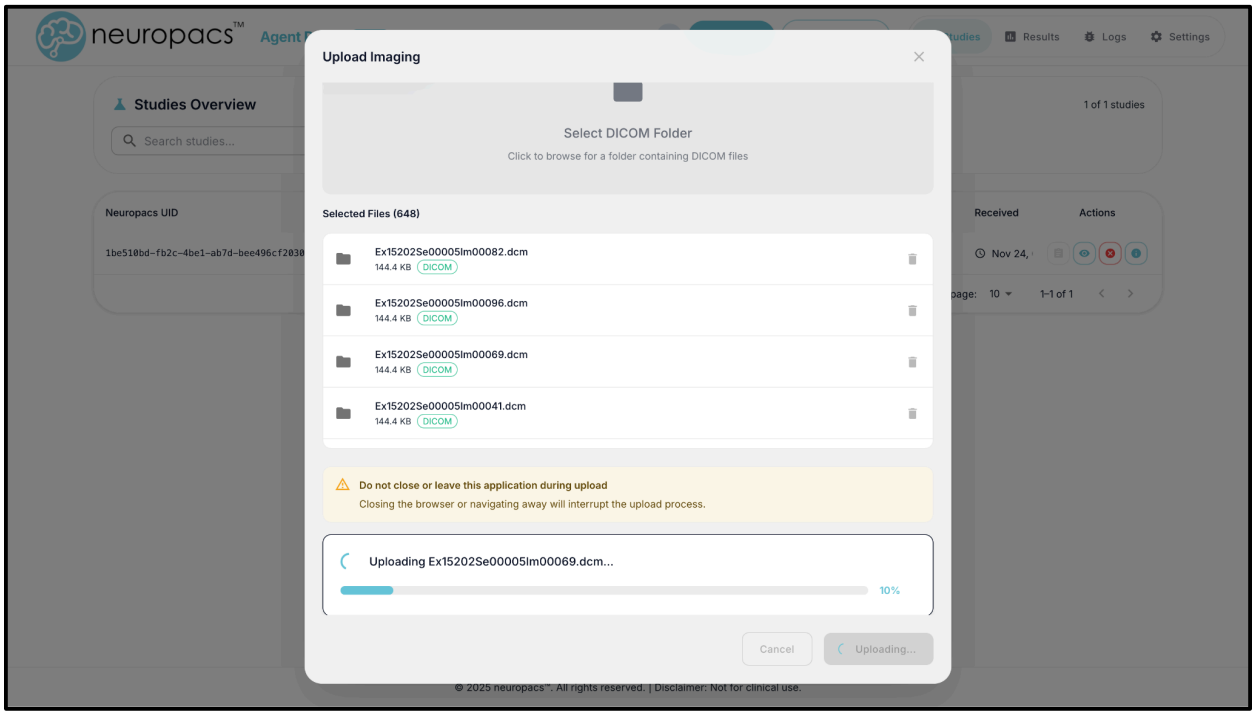

Figure 9: Manual upload indicator

Note: Patient biological age and sex must be provided through one of the following:

- this upload form,
- DICOM metadata, or
- a JSON sidecar when uploading via ZIP archive

Note: The Neuropacs Agent applies the following order of precedence when determining metadata values (highest to lowest):

1. Metadata entered in the upload form
2. JSON sidecar metadata
3. DICOM metadata (when available)

Note: For more information regarding upload specifications, visit our [Documentation Website](#).

Viewing Results

Access all available reports through the **Results** page. This page allows you to view reports directly in the browser, download them in multiple formats (JSON, PNG, PDF, TXT, XML), and manually re-send reports if delivery fails or a retransmission is required.

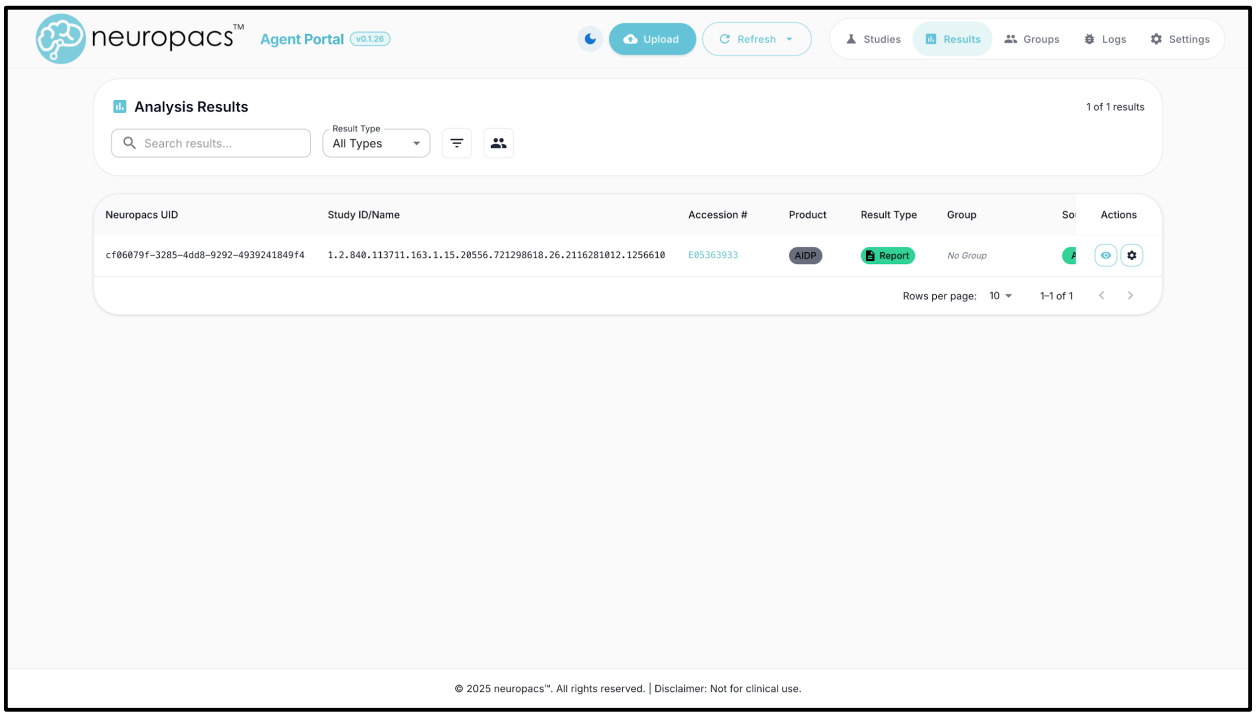

Figure 10: Integrated UI Results page

Manage results by selecting the **Gear** icon in the **Actions** tab.

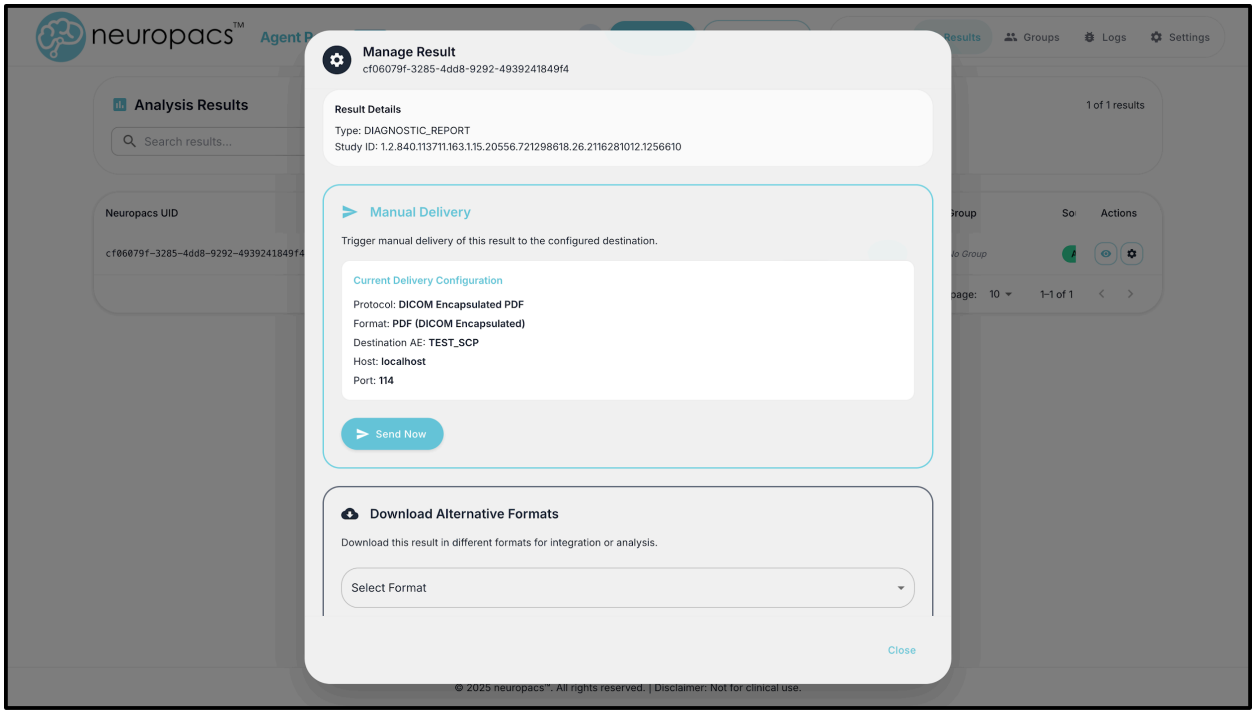

Figure 11: Manual result management popup

Manage results by selecting the **Eye** icon in the **Actions** tab.

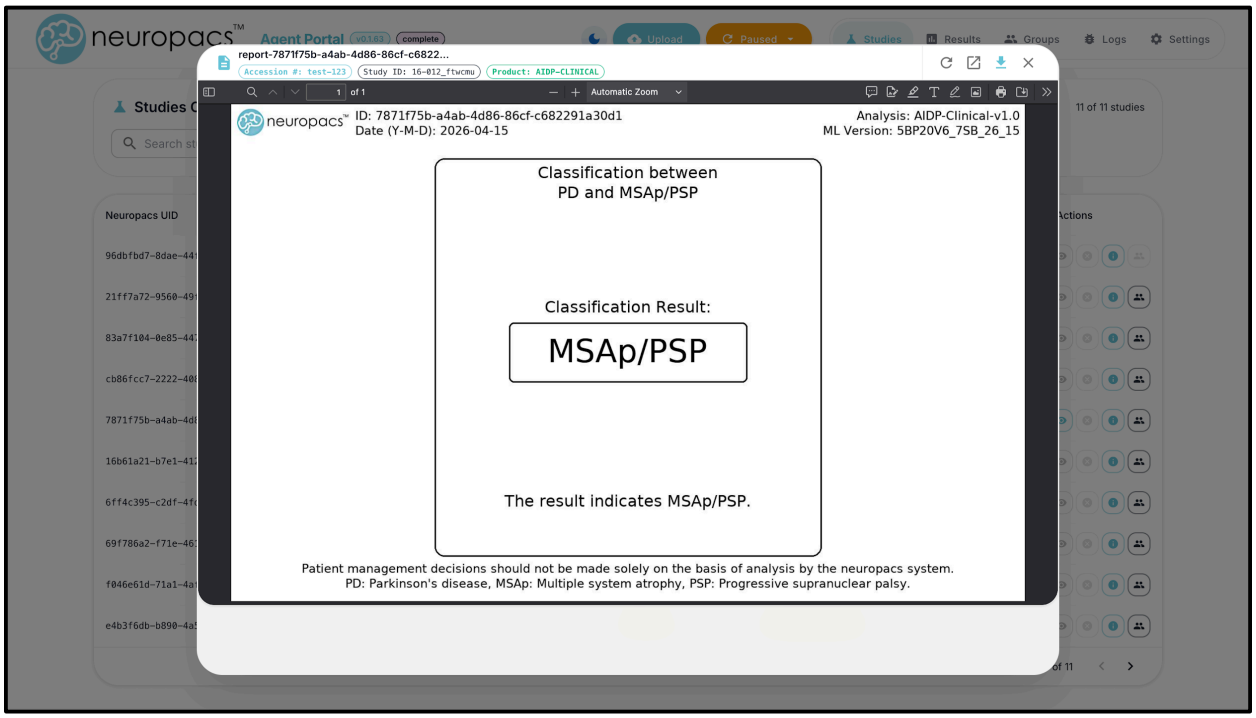

Figure 12: Integrated PDF report viewer

Group Management

Access the **Groups** page to create and manage groups and download group reports generated by the Neuropacs Agent. Configurations related to group reports can be managed in the **Settings** page.

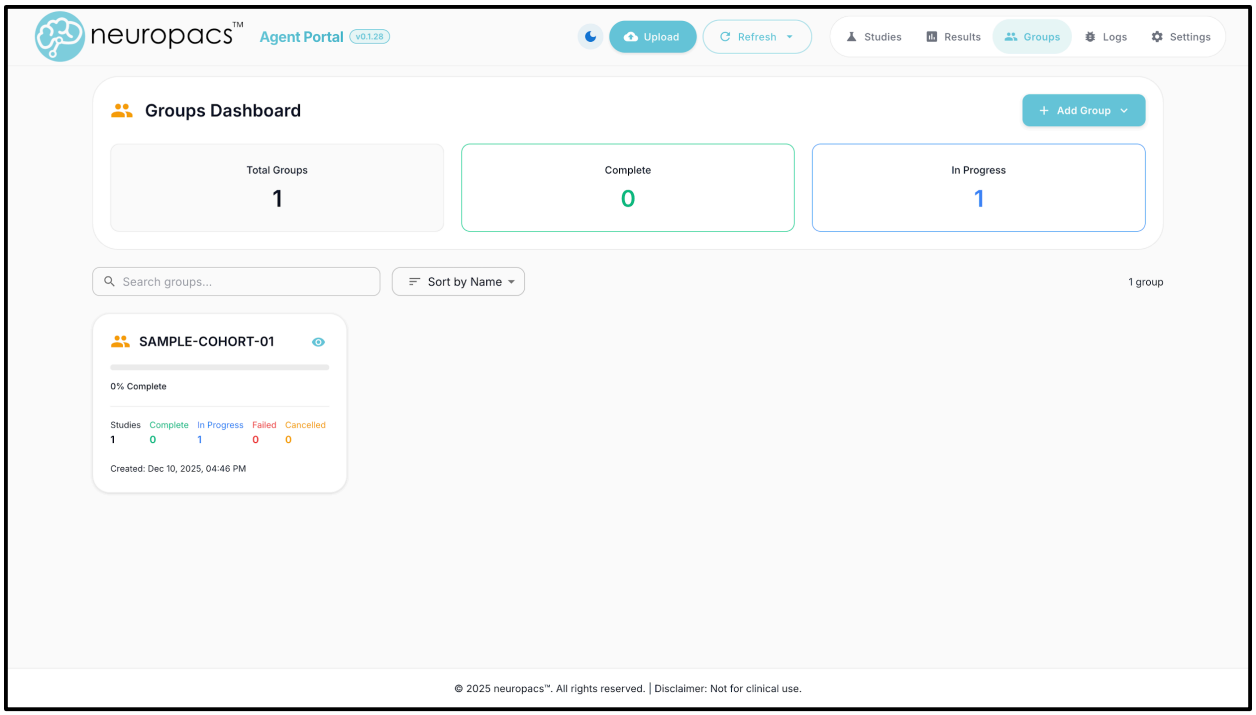

Figure 13: Groups page

Select the **Eye** icon for a group to view details and manually manage reports. Select **Send Report** to manually send the report to the configured destination. Select **Download Report** to instantly download the CSV report to disk.

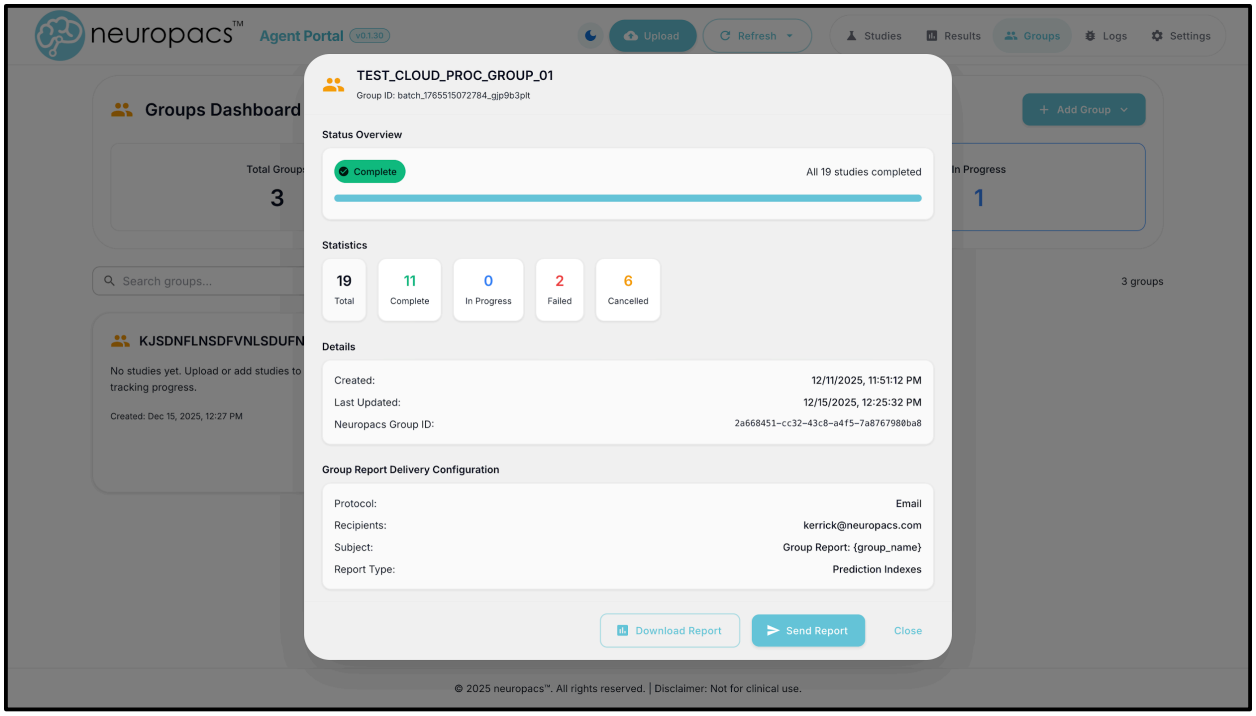

Figure 14: Group details

Create groups by selecting the **“Create New Group”** button. **Note:** All studies associated with a group MUST use the same products.

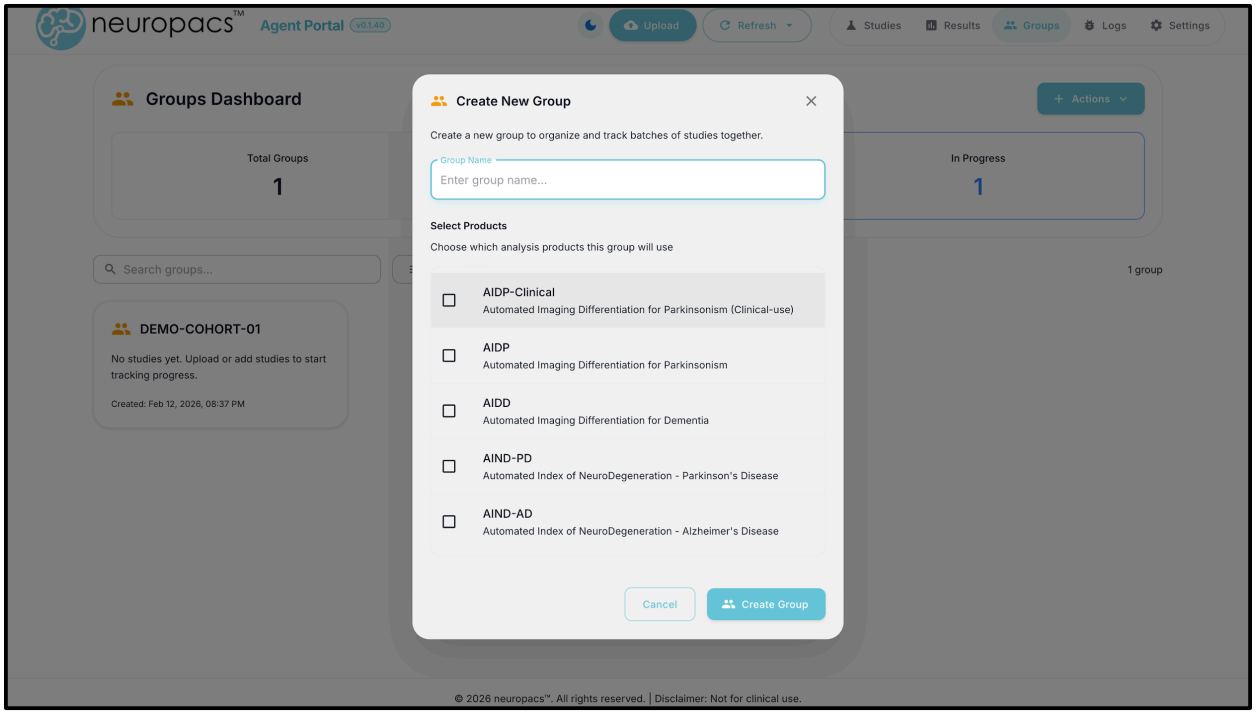

Figure 15: Group creation form

Uploads a group of studies and auto-assign to a group by selecting the **“Upload New Group”** button. **Note:** A group of studies must be uploaded as a single folder containing a single ZIP archive for each study. Each ZIP must include either a DICOM study folder with an optional JSON sidecar, or a NIfTI (.nii/.nii.gz) with matching .bvec, .bval, and JSON sidecar. Select **“Download JSON sidecar template”** to download a template file.

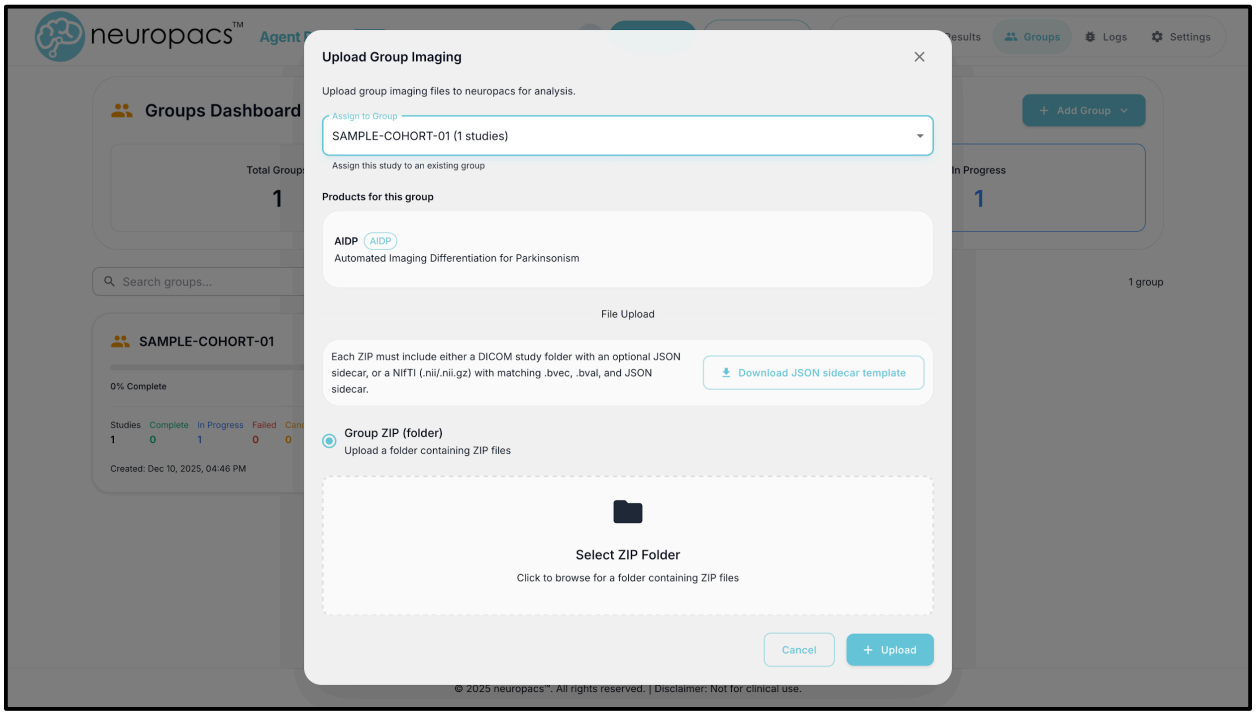

Figure 16: Group upload form

Log Viewer

Access the **Logs** page to view, filter, search, stream, and download logs generated by the Neuropacs Agent. Selecting **File, CSV, or JSON** options will download the current log file in the corresponding format. Selecting the **ZIP** download option will download all available log files. Select **Stream logs** to enable live streaming of new logs.

Neuropacs Cloud logs can be provided upon request.

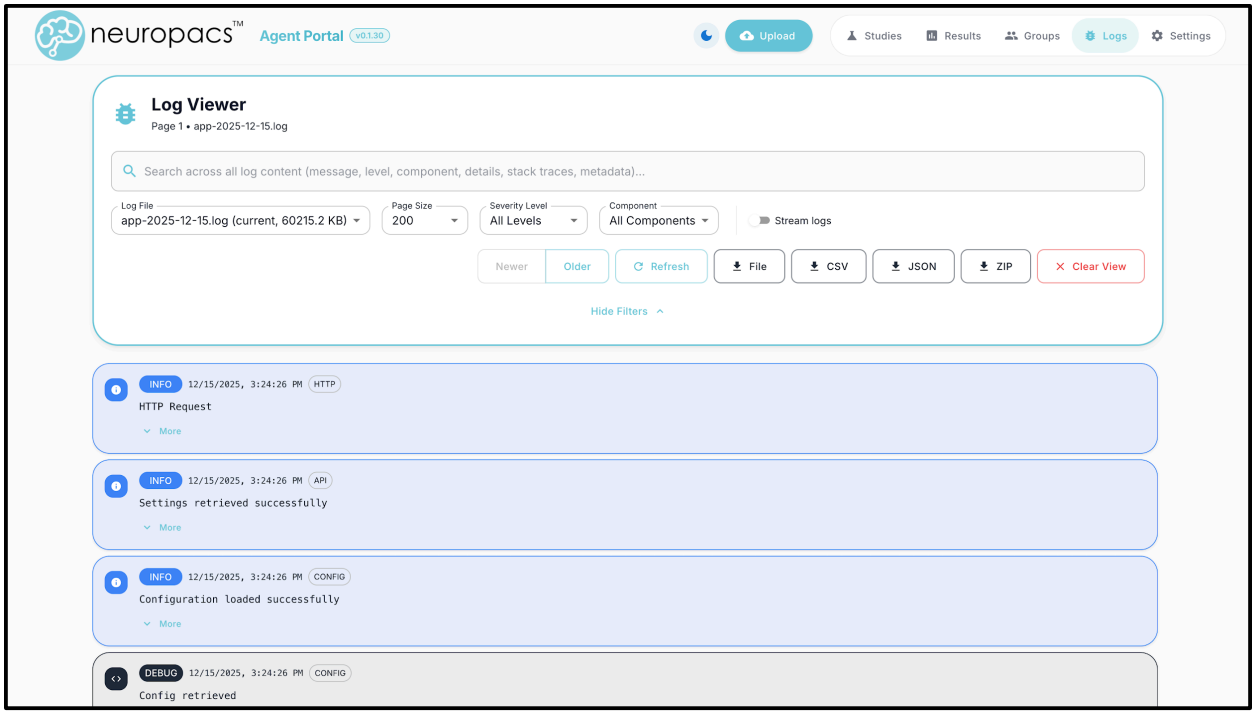

Figure 17: Logs page

Settings

Access the **Settings** page to view/modify Neuropacs Agent configurations.

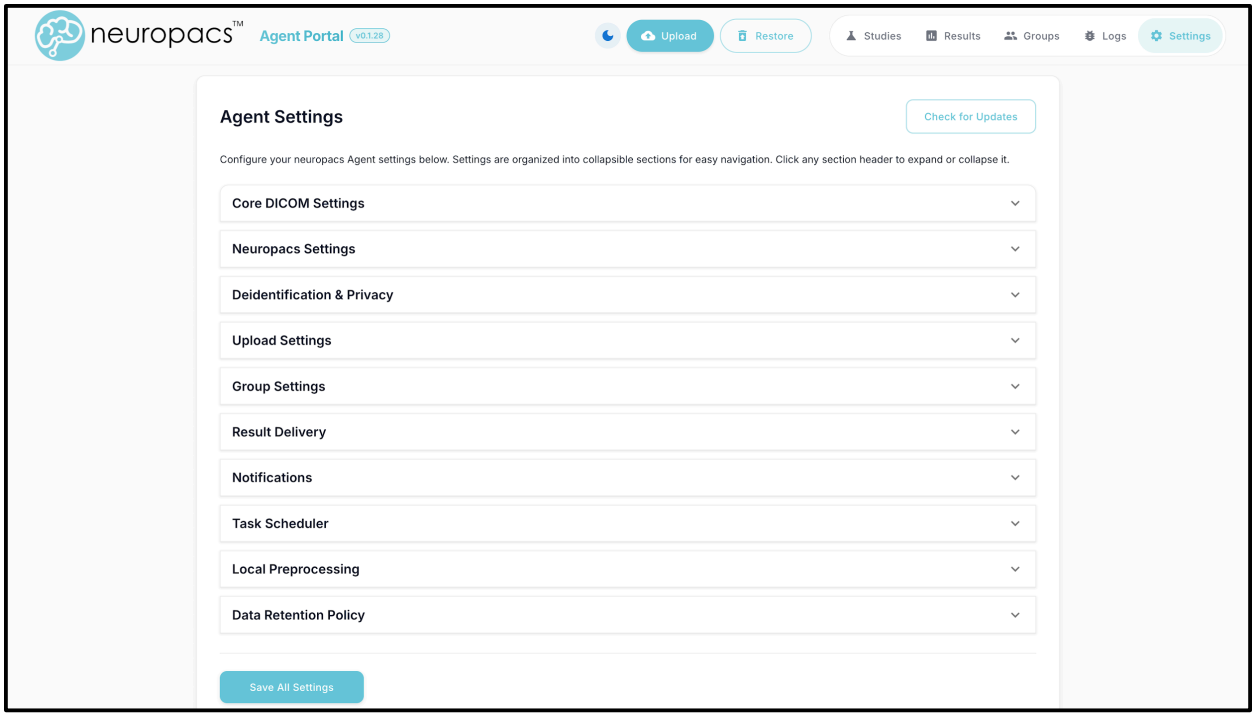

Figure 18: Settings page

Updates

Updates can be performed directly in the **Settings** page. To check if your installation is out-of-date, select **“Check for Updates”**. If your system application version does not match the latest release, an option will be available to update your system. This process takes ~5 minutes and retains all application data.

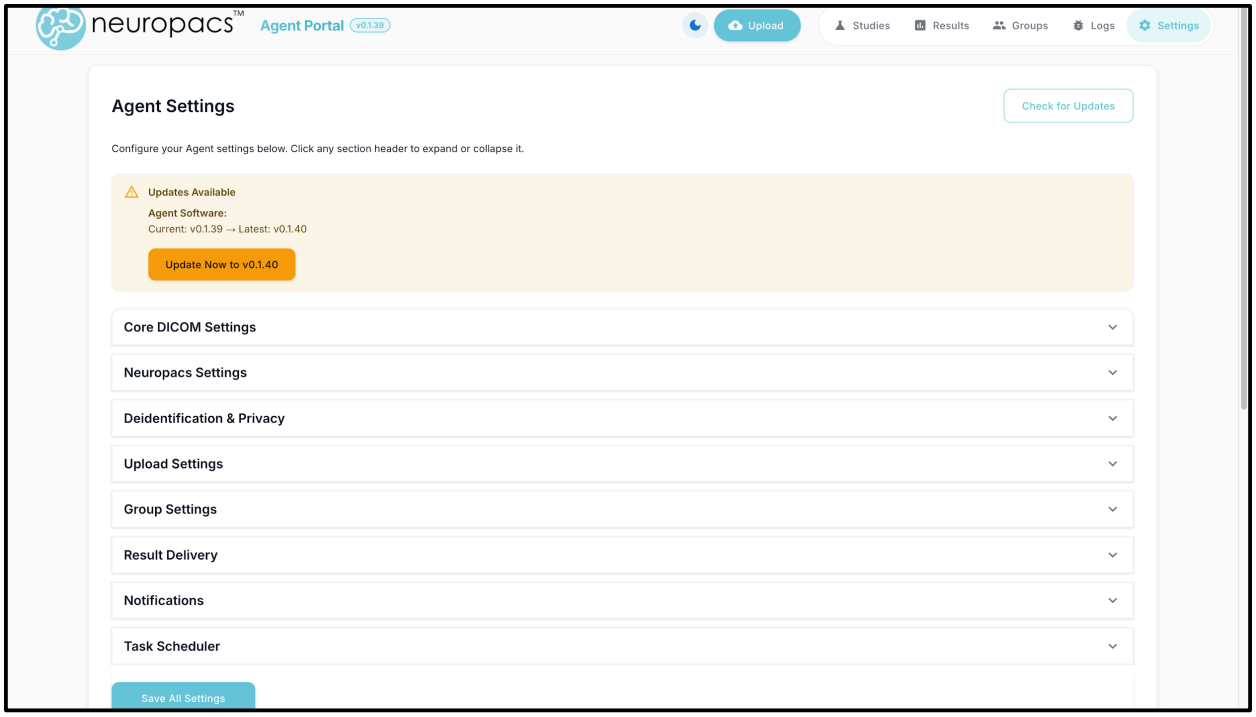

Figure 19: Update dialog

API Documentation

The Neuropacs Agent includes built-in interactive API documentation powered by Swagger UI. It provides a complete reference for all REST endpoints exposed by the agent, organized by functional area: health checks, study management, order lifecycle, result retrieval, batch operations, delivery, settings, updates, logs, and more.

Access: Navigate to /api-docs on your running agent (e.g. https://localhost:3001/api-docs).

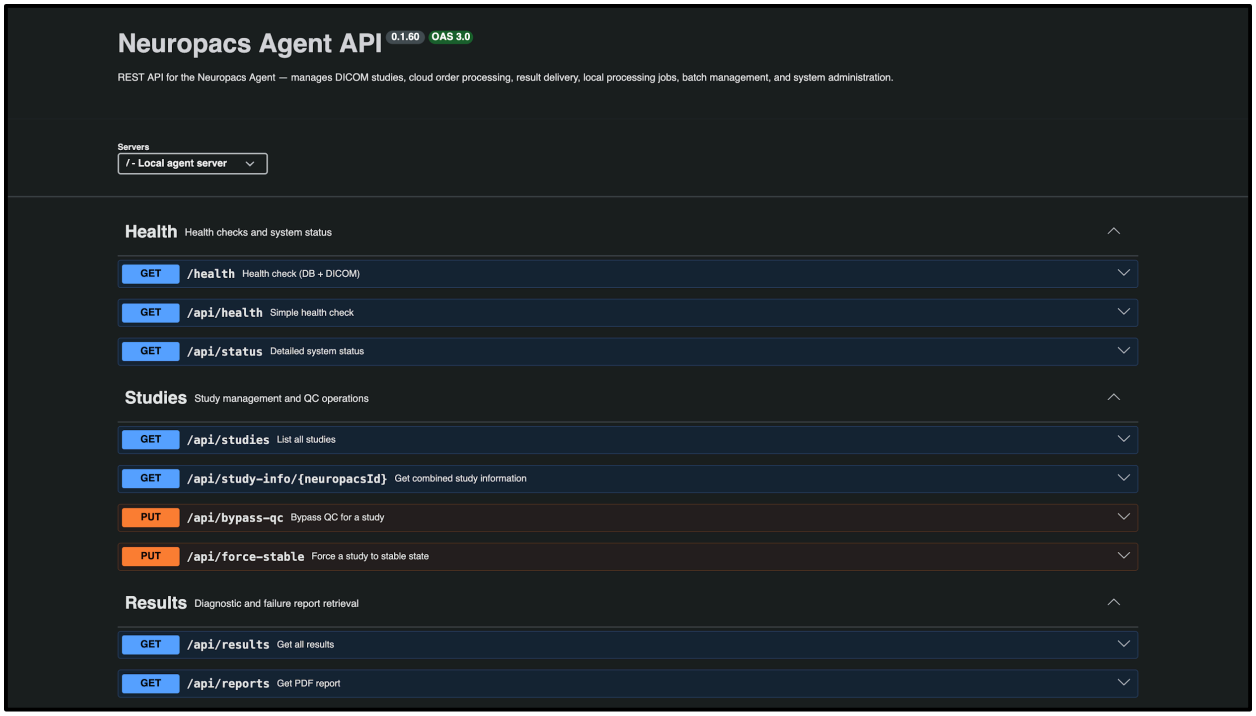

Figure 20: API Swagger page

Authentication + Role-Based Access Controls

The Neuropacs Agent supports two authentication methods for accessing the web interface: local service password and SAML 2.0 Single Sign-On.

Local password authentication uses a bcrypt-hashed service credential with HMAC-SHA256 session tokens stored in HttpOnly cookies. SAML SSO enables SP-initiated login against an external Identity Provider (e.g., Microsoft Entra ID, ADFS), with assertions validated using the configured IdP signing certificate and sessions maintained via signed tokens that are automatically invalidated on SAML configuration changes.

The authentication mode is configurable as local-only, SAML-only, or both. Role-Based Access Control enforces two roles: admin and user. Users who authenticate via the service password always receive the admin role.

SAML-authenticated users are assigned a role based on IdP group claims matched against the Admin Group and User Group values configured in the Agent settings; users whose groups match the Admin Group receive admin privileges, while all others default to the user role. The user role permits read access to studies, results, batches, and logs, as well as file uploads and order creation, but restricts write operations such as modifying settings, triggering updates, uploading TLS certificates, bypassing QC, and manually delivering results.

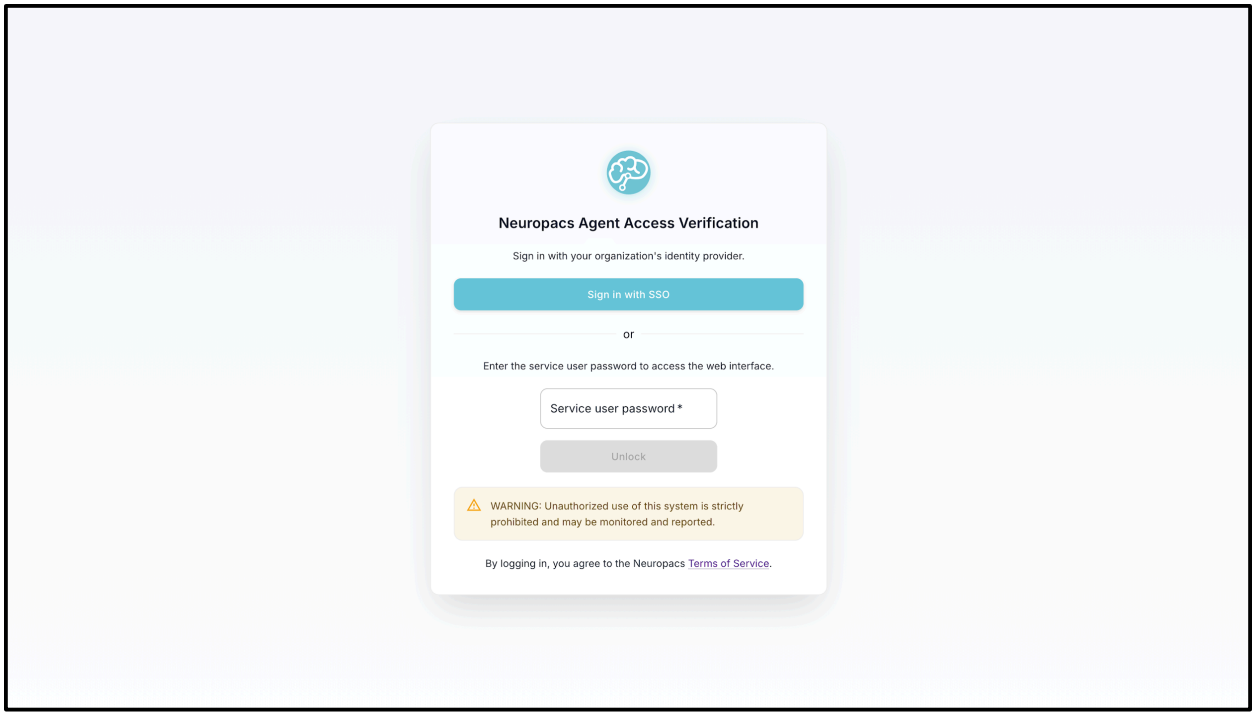

Figure 21: Authentication prompt

Authenticated sessions, maintained through a browser cookie, are automatically terminated after 8 hours of inactivity.

If the service user password has been updated after session expiration, users must enter the **new password** at the authentication prompt. Upon successful authentication, the application will synchronize the updated credentials and grant access.

On-Premise Quality Control (QC)

The Neuropacs Agent performs on-premise quality control to validate the integrity and suitability of submitted imaging data prior to processing. Quality control checks vary based on the imaging data type and are designed to ensure compliance with Neuropacs analysis and clinical requirements.

DICOM-Based Studies

- **MRI Series Validation** — Confirms the study contains appropriate MRI imaging series.
- **DTI/DWI Detection** — Identifies the presence of Diffusion Tensor Imaging (DTI) or Diffusion Weighted Imaging (DWI) sequences.
- **Patient Demographics Validation** — Ensures required patient metadata, including age and sex, is present.
- **Instructions for Use (IFU) Compliance** — Verifies that study and series descriptions conform to Neuropacs Instructions for Use (IFU) requirements.
- **Lossless Compression Verification** — Confirms images are not encoded using lossy compression formats that may degrade analysis quality.

NIfTI-Based Studies

- **NIfTI File Detection** — Validates the presence of NIfTI-format imaging files.
- **DTI Required File Validation** — Ensures all required DTI files are present (.nii/.nii.gz, .bvec, .bval, and .json).
- **Metadata Sidecar Validation** — Confirms the JSON sidecar contains required patient demographic information.

Important: Bypassing QC checks may:

- Result in additional processing charges
- Lead to incomplete or failed analyses
- Produce unreliable or non-diagnostic results

Support

For additional information regarding the Neuropacs Agent or this user guide, please contact our support team at **support@neuropacs.com**.