

# Plunger Analysis System (PAS) API: Script-Based Access Developer Guide

---

## Contents

<b>Contents</b> .....	<b>1</b>
<b>List of figures</b> .....	<b>1</b>
<b>Additional information</b> .....	<b>2</b>
<b>Disclaimer</b> .....	<b>2</b>
<b>Safety</b> .....	<b>2</b>
<b>1 Overview</b> .....	<b>2</b>
<b>2 PowerShell script-based access</b> .....	<b>3</b>
2.1 General setup .....	3
2.2 Fault Detection access.....	4
2.3 Optimization access .....	5

---

## List of figures

Figure 1-1: Setup for authentication by PAS .....	3
Figure 2-1: Script package contents and folder structure .....	3
Figure 2-2: Sample script for access to PAS Fault Detection.....	4
Figure 2-3: Sample script for access to PAS Optimization.....	5

## Additional information

Additional free publications are available for download at [www.abb.com/upstream](http://www.abb.com/upstream).



[Plunger Analysis Software homepage](#)

**Table 0-1: Related documentation**

Documents	Document number
Plunger Analysis System (PAS) User Guide	<a href="#">2107720</a>
Plunger Analysis System (PAS) API: Sample Code for Access and File Conversion	<a href="#">2107722</a>

## Disclaimer

This document describes software scripts that serve as an example of scripting logic to access the PAS API for fault detection or optimization. The examples are provided as an aid for customers designing their own custom PAS client implementations, but not as a product that is supported by ABB. ABB is not liable for the incorrect use of any part of the scripts shown in this document. Customers are encouraged to review this document and examine the code before copying any or all lines of the code. ABB expects that customers carefully examine their requirements and create the solutions appropriate to their own environment. If any part of the ABB code is used or integrated into a custom solution, it is expected that customers conduct their own testing process and verify that the results meet their requirements.

## Safety

The sample scripts shown in this document use input files with data created by ABB devices. These files are used for calls to the Plunger Analysis System API. Analysis results based on the input data can detect faults or provide values to optimize the operation of a plunger system. Any custom or third-party function or script that fetches the analysis results must be carefully tested. Modifications and optimization of a plunger system based on analysis results may affect real-time operation or production. Be sure to test applications and analysis results before changing actual plunger system configuration to prevent service disruption or safety hazards.

## 1 Overview

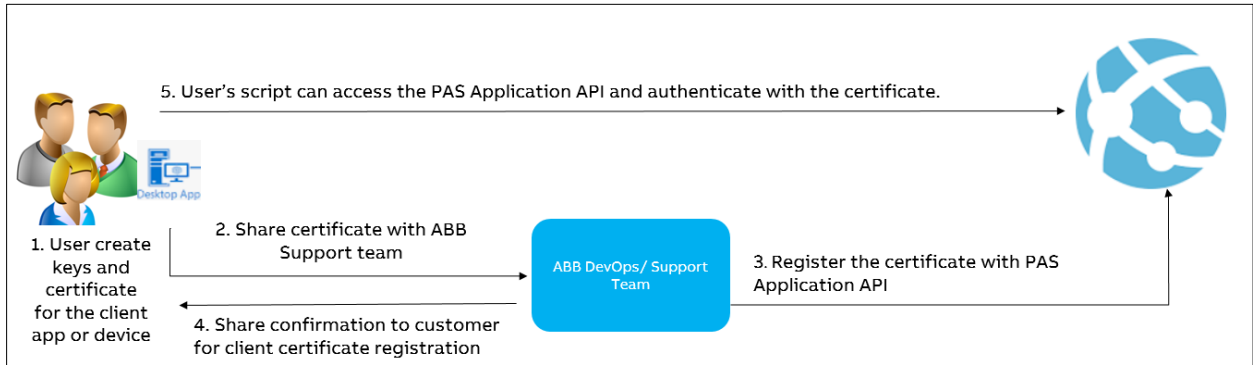
This document describes examples of scripts designed to access the Plunger Analysis System (PAS) REST-based Application Programming Interface (API). Scripts access to the API automates the request for fault detection or optimization analyses.

To access the API, the client scripts must authenticate with a CA signed certificate. [Figure 1-1](#) shows the required process to ensure client applications or scripts can access PAS. The customer must create or obtain the appropriate (CA signed) certificate for use by the scripts or client applications (1). The customer must provide the public certificates to ABB (2). ABB registers the certificate on the PAS API (3). ABB notifies the customer when registration is complete and effective (4). Once registration is complete, the user can have the certificate ready to incorporate into their client apps or scripts (5).



**IMPORTANT NOTE:** ABB does not provide certificates. Ask your IT administrator to obtain a CA signed certificate. Certificate management is the sole responsibility of the customer. Certificates must be updated prior to expiration.

**Figure 1-1: Setup for authentication by PAS**



PAS APIs support:

- Fault Detection: <https://<domainname>/plungerlift/api/FaultDectcion/GetReport>
- Optimization: <https://<domainname>/plungerlift/api/Optimization/GetReport>

This section provides two sample scripts and the file structure that the customer must create to run the sample scripts successfully. The scripts expect to find the required files and certificates in the file structure as described in section [2.1 General setup](#). Create the required folders and copy required files before running the scripts.



**IMPORTANT NOTE:** To describe examples of scripts, this document uses Windows® PowerShell scripting language. The scripts do not implement definitions for all use cases. It is up to the customer's developers to use the script language of their choice, run environment and systems, and the logic that meets their specific needs.

## 2.1 General setup

[Figure 2-1](#) shows the required file structure to run the sample ABB scripts shown in this document. Before running the script:

1. Create the script root directory named: PSScriptRoot. Scripts and additional subfolders will be under this directory.
2. In the PSScriptRoot folder:
  - a. Copy scripts. It is assumed you have created the scripts following the examples provided.
  - b. Create a folder named: Input. This folder should contain the input file to be analyzed. For example:
    - Input file for Fault Detection: FaultDetection.json
    - Input file for Optimization: Optimization.json
  - c. Create a folder named: Output. When scripts complete execution, they save their output files or results in this folder.
  - d. Create a folder named: Certificate. Place the valid CA certificate in this folder (if using the exact examples in this document, the certificate name should match the name in the script: TestCertificate.cer).

**Figure 2-1: Script package contents and folder structure**

Certificate	9/1/2021 9:19 PM	File folder	
Input	9/1/2021 9:15 PM	File folder	
Output	9/1/2021 9:21 PM	File folder	
PASAPI_Call_FaultDetetcion_Read_Certificate.ps1	9/1/2021 9:15 PM	Windows PowerS...	2 KB
PASAPI_Call_Optimization_Read_Certificate.ps1	9/1/2021 9:15 PM	Windows PowerS...	2 KB

## 2.2 Fault Detection access

[Figure 2-3](#) shows the PowerShell script: PASAPI\_Call\_FaultDetection\_Read\_certificate.ps1. This is the script sample to access PAS Fault Detection using API.

**Figure 2-2: Sample script for access to PAS Fault Detection**

```
PowerShell

[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12

Try
{
    $InputFilePath = $PSScriptRoot + '\Input\FaultDetection.json'
    $OutputFilePath = $PSScriptRoot + '\Output\Output_FaultDetection.json'
    $CertificatePath = $PSScriptRoot + '\Certificate\ TestCertificate.cer'
    $Body = (Get-Content $InputFilePath)
    $uri = "https://<domainName>/plungerlift/api/FaultDetection/GetReport"
    $certificate = New-Object
System.Security.Cryptography.X509Certificates.X509Certificate2 ($CertificatePath
,$null,[System.Security.Cryptography.X509Certificates.X509KeyStorageFlags]::Mac
hineKeySet)

    Write-Output '***** Fault Detection API call Processing*****'
    $response = Invoke-RestMethod -Uri $uri -method "POST" -body $Body -
Certificate $certificate -ContentType "application/json"
    $response | ConvertTo-Json | Out-File $OutputFilePath
    Write-Output '* Fault Detection API call Processed Successfully..Please
check output folder*****'
}
Catch
{
    $_.Exception.ToString()
    $error[0] | Format-List -Force
}
}
```

## 2.3 Optimization access

[Figure 2-3](#) shows the PowerShell script: PASAPI\_Call\_Optimization\_Read\_certificate.ps1. This is the script sample to access PAS Optimization using API.

**Figure 2-3: Sample script for access to PAS Optimization**

```
PowerShell

[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12

Try
{
    $InputFilePath = $PSScriptRoot + '\Input\Optimization.json'
    $OutputFilePath = $PSScriptRoot + '\Output\Output_Optimization.json'
    $CertificatePath = $PSScriptRoot + '\Certificate\PublicKeyCertificateTest.cer'
    $Body = (Get-Content $InputFilePath)
    $uri =
"https://paswebapppipe.azurewebsites.net/plungerlift/api/Optimization/GetReport"
    $certificate = New-Object
System.Security.Cryptography.X509Certificates.X509Certificate2($CertificatePath,$nul
l,[System.Security.Cryptography.X509Certificates.X509KeyStorageFlags]::MachineKeySet
)
    Write-Output '**** Optimization API call Processing*****'
    $response = Invoke-RestMethod -Uri $uri -method "POST" -body $Body -Certificate
$certificate -ContentType "application/json"
    $response | ConvertTo-Json | Out-File $OutputFilePath
    Write-Output '***** Optimization API call Processed Successfully..Please check
output folder*****'
}
Catch
{
    $_.Exception.ToString()
    $error[0] | Format-List -Force
}
```

---

**ABB Inc.**

Measurement & Analytics

Quotes: [US-IAMA.inquiry@us.abb.com](mailto:US-IAMA.inquiry@us.abb.com)

Orders: [US-IAMA.order@us.abb.com](mailto:US-IAMA.order@us.abb.com)

Training: [US-IAMA.training@us.abb.com](mailto:US-IAMA.training@us.abb.com)

Support: [upstream.support@us.abb.com](mailto:upstream.support@us.abb.com)

+1 800 442 3097 (opt. 2)

Additional free publications are available for download at:

[www.abb.com/upstream](http://www.abb.com/upstream)

**Main Office - Bartlesville**

7051 Industrial Blvd  
Bartlesville, OK 74006  
Ph: +1 918 338 4888

**Kansas Office - Liberal**

2705 Centennial Blvd  
Liberal, KS 67901  
Ph: +1 620 626 4350

**Texas Office - Houston**

3700 W. Sam Houston  
Parkway S., Suite 600  
Houston, TX 77042  
Ph: +1 713 587 8000

**Texas Office – Odessa**

8007 East Business 20  
Odessa, TX 79765  
Ph: +1 432 272 1173

**Texas Office – Pleasanton**

150 Eagle Ford Road  
Pleasanton, TX 78064  
Ph: +1 830 569 8062



We reserve the right to make technical changes or modify the contents of this document without prior notice. With regard to purchase orders, the agreed particulars shall prevail. ABB does not accept any responsibility whatsoever for potential errors or possible lack of information in this document.

We reserve all rights in this document and in the subject matter and illustrations contained therein. Any reproduction, disclosure to third parties or utilization of its contents - in whole or in parts - is forbidden without prior written consent of ABB.

Windows® is a registered trademark of Microsoft.

2107724MNAA

Copyright© 2021 ABB all rights reserved