ESMIG U2A

Qualified Configurations

V1.3

Author	4CB
Version	1.3.2
Date	01/12/2021
Status	Final
Classification	Unclassified
Accessible	
Classified until	

History of releases

RELEASE	DATE	ISSUES	STATUS[1]
1.0	01/03/2021	First version. Applicable for TIPS	Draft
1.0.1	06/04/2021	Second version, clarifications on support for terminal servers	Draft
1.1	05/05/2021	Third version. Extension to CLM and RTGS GUIs	Final
1.2	26/07/2021	Added terminal server support for Ascertia client. Extension to ECMS. Ascertia client URLs changed. Minor clarifications on U2A configurations. Added section in the annex concerning the GSD multi-user solution	Final
1.3	20/08/2021	Minor integrations to GSD multi-user solution installations Notes and typos amended	Final
1.3.1	15/10/2021	Added notes about HSM based certificate usage. Added clarifications in the "GSD multi-user solution installation" Annex.	
1.3.2	01/12/2021	GSD MU installation procedure revised according to 6.9 client delivery	

[1] Status value : Draft, Open, Final, Dismiss

Table of contents

1 INTRODUCTION

1.1 Purpose and Objectives

This document describes the general configuration that ESMIG users shall be complaint with in order to access TIPS, ECMS, RTGS, CLM and CRDM GUI via the ESMIG web portal. A specific section is devoted to describe the technical framework needed to fully implement the non-repudiation of origin functionality (NRO). This solution will be implemented in TIPS via the Change Request TIPS-0034-SYS, when the applet technology will be decommissioned in favour of a browser's java plugin independent solution. In RTGS and CLM GUIs the same solution will be implemented according to the official plan.

1.1.1 Background remarks

The aim of the ESMIG qualified configurations is to provide ESMIG users with a specific configuration that is proved to be fully working.

As already mentioned, the NRO solution, based on the Ascertia Go>Sign Desktop application, will be the unique U2A NRO solution to be adopted for TARGET services, therefore only one version of the Go>Sign Desktop client will be used and distributed across the different services.

Important also to hightlight that Go>Sign Desktop client applications are already in use in TARGET2 for Internet Access and Contingency Network and 4CBs will guarantee that no different versions are needed by the relevant services using the client, before the go-live of CSLD project.

1.1.2 Qualified configurations

As already mentioned, the 4CB has qualified a specific subset of the NSPs compatibility matrix. These configurations have been extensively tested and support on them is guaranteed.

NSP	SWIFT	SIA-COLT
OS	Windows 10	
Browser	Google Chrome 88.0+, Firefox 78.0+	
Go>Sign Desktop	> 6.0.0.14	

These cryptographic key stores, used to access the signing keys, are supported:

- PKCS#11 for hardware-based tokens

- HSM based certificates (as per NSP specifications)

The 4CB will ask customers running a software version lower than that qualified to upgrade to a qualified version in order to proceed with problem investigation.

The 4CB will investigate issues experienced by customers while running a software version higher than that qualified: If the root cause is linked to the specific software version, then the 4CB will attempt to find a workaround (which may involve customers downgrading their software to a qualified version). The 4CB will evaluate whether a fix for the issue can be included in a future relevant TARGET Service GUI release.

Customers using totally or partially different system components or versions than those mentioned are then responsible to verify the full compatibility with the relevant TARGET Service GUI in the test environments and the system. The 4CB will in any case, provide support to the maximum extent possible for checking / testing alternative configurations in order to support troubleshooting process.

The local customer system set-up, i.e. adaption of the local firewall and security policies in order to enable the client installation, HTTPS transfer communication, access to the certificates on the USB tokens from the client machines (either physical or remote workstations) is under the sole responsibility of the end users (that may also need to involve their internal IT Dept. as well as external providers, in case of product specific issues).

1.2 Technical requirements and recommendations

1.2.1 SIgnle user download URLs

The client is available for download on the ESMIG portal (after log-in) at the following URLs:

EAC PORTAL

https://esmig-eac-portal.u2a.sianet.sia.eu/gosign/download/64bit_client

https://esmig-eac-portal.emip.swiftnet.sipn.swift.com/gosign/download/64bit_client

UTEST PORTAL

https://esmig-cert-portal.u2a.sianet.sia.eu/gosign/download/64bit_client

https://esmig-cert-portal.emip.swiftnet.sipn.swift.com/gosign/download/64bit_client

PROD PORTAL

https://esmig-portal.u2a.sianet.sia.eu/gosign/download/64bit_client

https://esmig-portal.emip.swiftnet.sipn.swift.com/gosign/download/64bit_client

(64bit version recommended ; 32bit version still available in case of need).

The full installation guide provided by Ascertia is distributed separately and it can be used as reference for specific needs (e.g. automated installations).

Downloading and installing the Go>Sign Desktop client is a mandatory step to sign U2A requests. Installation requires administrative privileges; local IT support must be involved to make sure the installation correctly ends.

Please make sure the correct version Go>Sign desktop is installed. To check this please right click on the go sign icon and choose "about". After that the following window appears:

Detailed installation steps and troubleshooting tips for Multi User environment are reported in the Annex.

1.2.2 Go>Sign Desktop Client Requirements

The client invocation on user side will be triggered by the web application (via Javascript) with the first attempt to sign an instruction (and each time the user needs to sign one) and is transparent to users.

ADSS Go>Sign Desktop relies on TLS communication only with the web application (port 8782). This communication is secured using a TLS server certificate having hostname:

client.go-sign-desktop.com.

Therefore, the local client machine must be able to resolve this FQDN (Fully Qualified Domain Name complete domain name for a specific computer, or host, on the internet) to itself.

In order to achieve this, the standard procedure foresees that the Go>Sign Desktop Installer automatically adds the entry :

127.0.0.1 client.go-sign-desktop.com

in the Operating System host file to register the client.go-sign-desktop.com as a local domain (Windows OS: C:\Windows\System32\Drivers\etc\hosts).

This will add the FQDN client.go-sign-desktop.com to resolve to IP address 127.0.0.1.

The default value client.go-sign-desktop.com must not be changed.

The TLS server certificate will be self-signed and different for each workstation where the client will be installed. Once loaded into Windows OS, it is expected to be found in the WINDOWS-ROOT CA keyring (i.e. and not in the personal certificate keyring).

The end users have to ensure that the security settings of their institutions, i.e. firewalls, allow for installation of the applet/desktop client as well as for code signing certificate revocation check, if not generally disabled. By default, User Account Control Settings (UAC) are enabled in windows. ADSS Go>Sign Desktop needs user permissions to make changes on the installing device. Windows always prompt a dialog to get the user permissions if user granted the permissions then ADSS Go>Sign Desktop would be installed on the device.

1.2.3 Other technical requirements

Here following a list of items to be checked before starting the test sessions or in case of exceptions that may block the testing. Internal IT support may be needed to perform these checks because security restrictions may be in place preventing the end users to complete them autonomously.

- As a general remark, please make sure that the configurations listed in the relevant NSPs documentation are applied (as a not exhaustive example, the mandatory changes on the pac file). For further details please refer to the "SWIFT's Solution for ESMIG U2A Setup Guide Step-by-Step" document and the " SIAnet.XS Connectivity Services for ESMIG U2A User Guide"
- In case of local network exceptions in the browser (i.e. **TUNNEL CONNECTION FAILED, NAME NOT RESOLVED**) during first interaction with new Ascertia infrastructure: add DSS host certificates in browsers keyring (e.g Chrome and Firefox). Host names following for information:

SIA TST	esmig-tst-dss.u2a.sianet.sia.eu
SIA CRT	esmig-cert-dss.u2a.sianet.sia.eu
SIA PRD	esmig-dss.u2a.sianet.sia.eu
SWIFT TST	esmig-tst-dss.emip.swiftnet.sipn.swift.com
SWIFT CRT	esmig-cert-dss.emip.swiftnet.sipn.swift.com
SWIFT PRD	esmig-dss.emip.swiftnet.sipn.swift.com

The same above URL may need to be added to the browsers trusted sites.

- In case of Cross-Origin Resource Sharing (CORS) issue during first interaction with new Ascertia infrastructure, please temporarily disable CORS checks both in Chrome and/or FF

a. FF --> https://addons.mozilla.org/es/firefox/addon/access-control-allow-origin/ + Toggle ON

b. Chrome --> "C:\Program Files (x86)\Google\Chrome\Application\chrome.exe" --disable-web-security --user data-dir="C:\.....\Chrome" (for single user environment)

c. Chrome --> "C:\Program Files (x86)\Google\Chrome\Application\chrome.exe" --disable-web-security (for multi user environment)

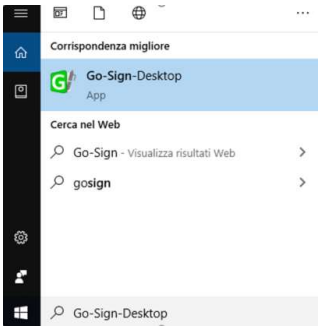
- Check windows host file for the definition 127.0.0.1 client.go-sign-desktop.com

- Check installation / install "client.go-sign-desktop.com" certificate in Chrome and Firefox explicitly (or check first if it is in all browsers keyring after GSD client installation). Without this exception error code 404 may be displayed. Also ensure that Firefox is allowed to check / read certificates from Windows keystore.

It is finally suggested to ensure that one token at time is connected to a workstation during signing operation.

1.3 Running the Application Go-Sign-Desktop

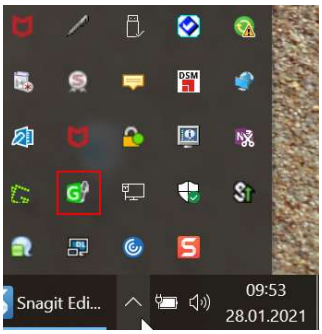
Once the application is installed, it is usually configured to run automatically when a Windows session is started. However, due to specific security settings, this might not happen, resulting signature attempts to end with a similar error "Go>Sign desktop not running/installed". In this case, it is necessary to run it manually before initiating a browsing session in ICM. It is possible to lookup for the Go>Sign via the Windows Search bar:



If the icon is not found, this can be probably linked to problems occurred during the installation. The local IT support needs to be involved in this case.

1.3.1 Verifying Go>Sign application running

Ensure that the Go>Sign icon is featured in the system tray.



In addition, it is requested to verify that Go>Sign is running properly, by accessing the URL

https://client.go-sign-desktop.com:8782/gosign-desktop

The screenshot below is the expected result with Mozilla Firefox:

1.4 Troubleshooting information - Logging information

ADSS Go>Sign Desktop application has two log levels. First informational, which is for normal use, and second, debug, which should only be used when investigating performance issues, functionality problems, etc. For Windows OS and go>sign client configuration, users can view ADSS Go>Sign Desktop application logs at:

C:\Users\[User_Name]\AppData\Roaming\Ascertia\Go-Sign-Desktop\logs\

and should send the send the "GoSignDesktopLog.txt" when opening the incident to 4CB Service Desk.

1.4.1 Changing logging level

By default, ADSS Go>Sign Desktop logging level is set to INFO. To enable detailed debug logging, follow these instructions:

1. Go to ADSS Go>Sign Desktop installation path → C:\Program Files\Ascertia\Go-Sign-Desktop\app\conf\
2. Edit the gosign_desktop.properties file using a suitable text editor.
3. Change the value of the property GOSIGN_DESKTOP_LOG_LEVEL from INFO to DEBUG and save the file.
4. Stop ADSS Go>Sign Desktop application → right click ADSS Go>Sign Desktop application icon and select the option Quit.
5. Start ADSS Go>Sign Desktop application → Start Menu

2.1 GoSign Desktop (GSD) Client – Terminal server Installation Guide

Installation steps are reported in the following paragraph; they may subject to further changes / improvements in order to simplify the overall process. Installation consists in a code upgrade of the standard GSD single user client plus some customizations into the terminal server environment.

2.1.1 Remove previous installation of GSD MU client (6.6.0.14 + MU code add-on)

Follow these preliminary steps in order to remove previously installed GSD MU client:

- Stop GSD service
- Uninstall GSD application from Control Panel
- Remove the service ("sc delete "Go Sign Service")
- Remove nssm tool previously installed
- Delete c:\progran Files\Ascertia folder
- Reboot server

2.1.2 Download GSD multi user client (ESMIG and T2S customers)

- ESMIG customers can download the client from the following URLs (after log-in to ESMIG portal):

EAC PORTAL

https://esmig-eac-portal.u2a.sianet.sia.eu/gosign/download/64bit_client_MU

https://esmig-eac-portal.emip.swiftnet.sipn.swift.com/gosign/download/64bit_client_MU

UTEST/CERT PORTAL

https://esmig-cert-portal.u2a.sianet.sia.eu/gosign/download/64bit_client_MU

https://esmig-cert-portal.emip.swiftnet.sipn.swift.com/gosign/download/64bit_client_MU

- T2S customers can download the client from the following URLs (after log-in to ESMIG portal):

EAC STAGE

https://t2s-eac-gui.sia-colt.target-ssp.eu/ICMWeb/Ascertia/ADSS-Go-Sign-Desktop-Win64-SU.msi

https://t2s-eac-gui.ssp.swiftnet.sipn.swift.com/ICMWeb/Ascertia/ADSS-Go-Sign-Desktop-Win64-SU.msi

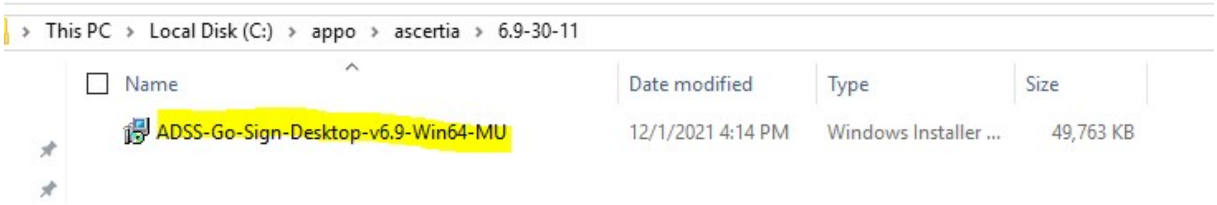
UTEST STAGE

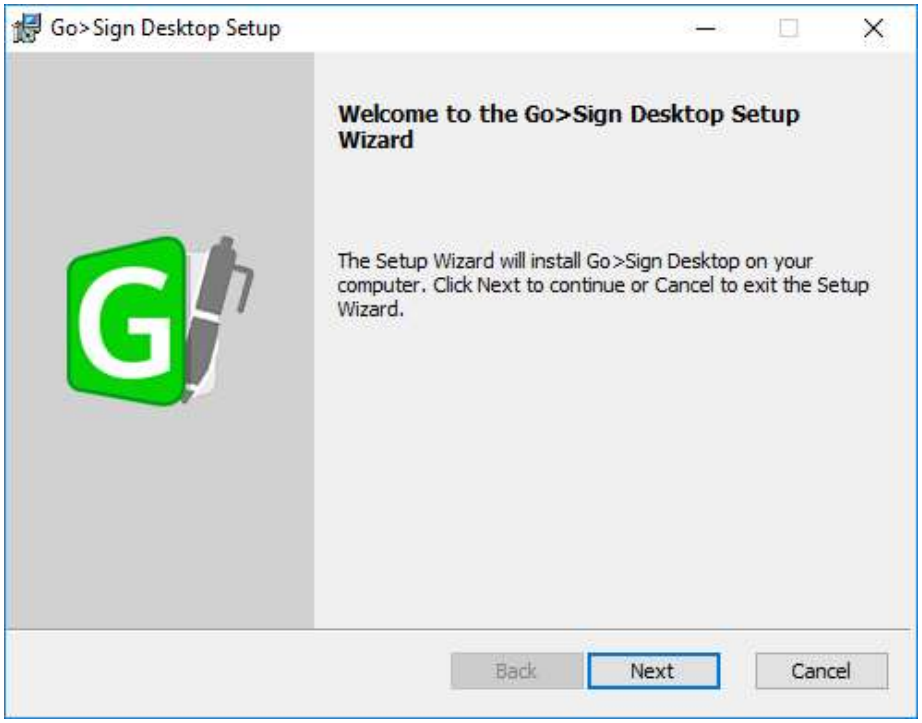
https://t2s-utest-gui.sia-colt.target-ssp.eu/ICMWeb/Ascertia/ADSS-Go-Sign-Desktop-Win64-SU.msi

https://t2s-utest-gui.ssp.swiftnet.sipn.swift.com/ICMWeb/Ascertia/ADSS-Go-Sign-Desktop-Win64-SU.msi

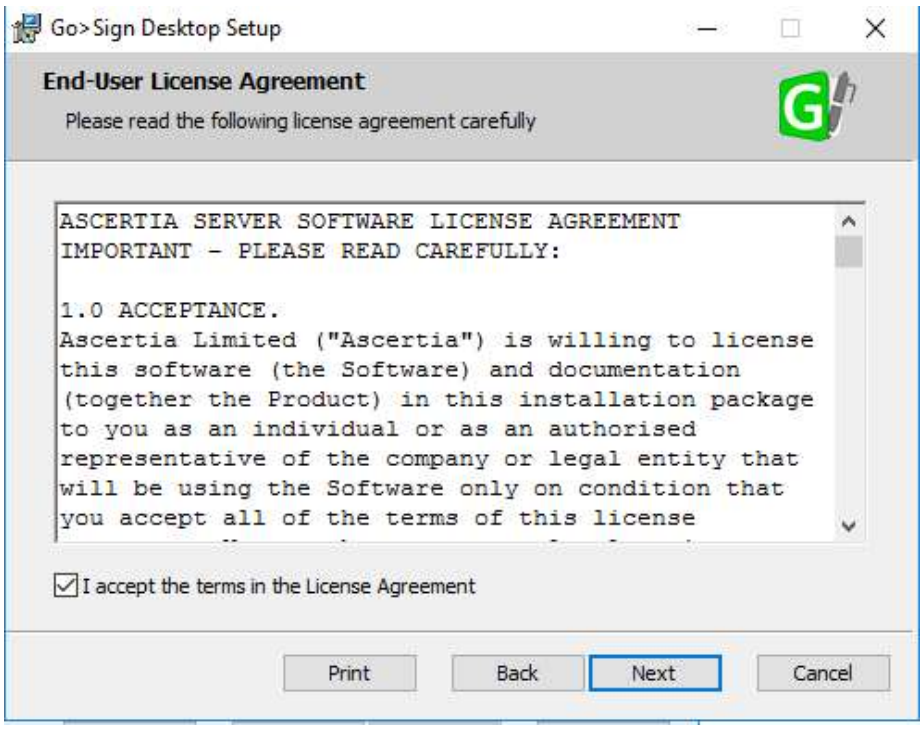
2.1.3 Setup GSD multi user client

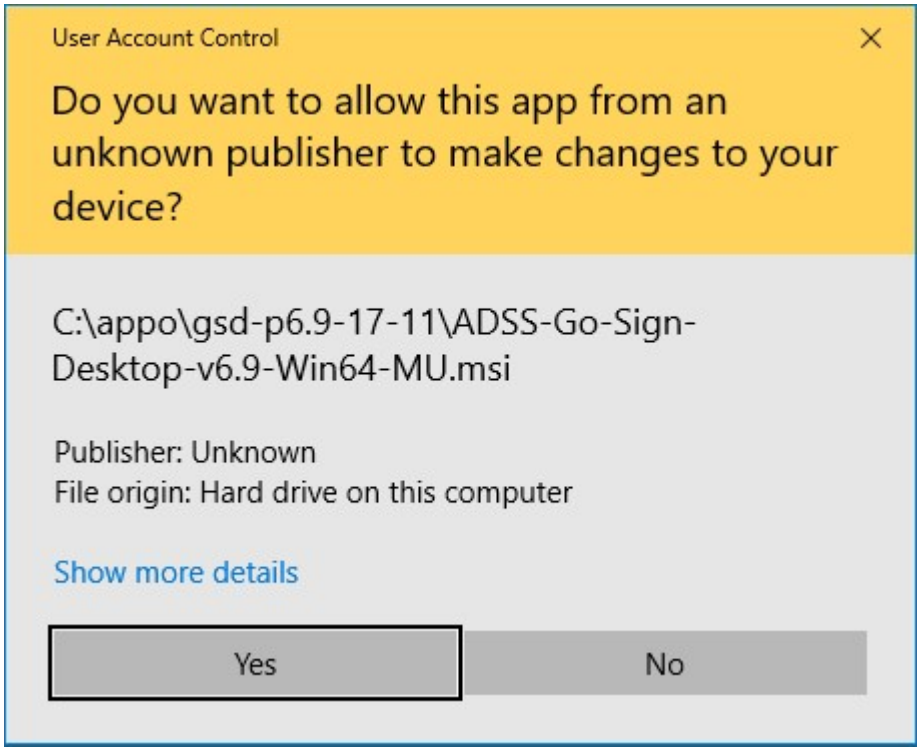
- Open Command prompt as Administrator
- Execute command: chgusr /install
- then run ADSS-Go-Sign-Desktop-v.6.9-Win64-MU.msi installation package



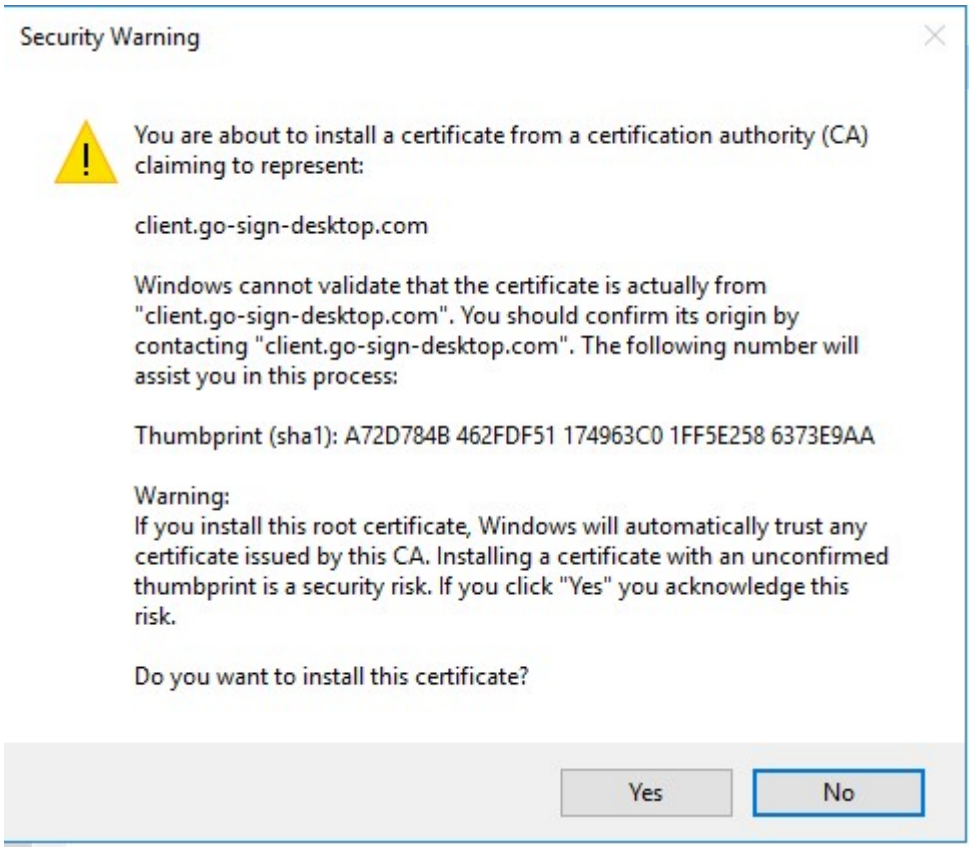


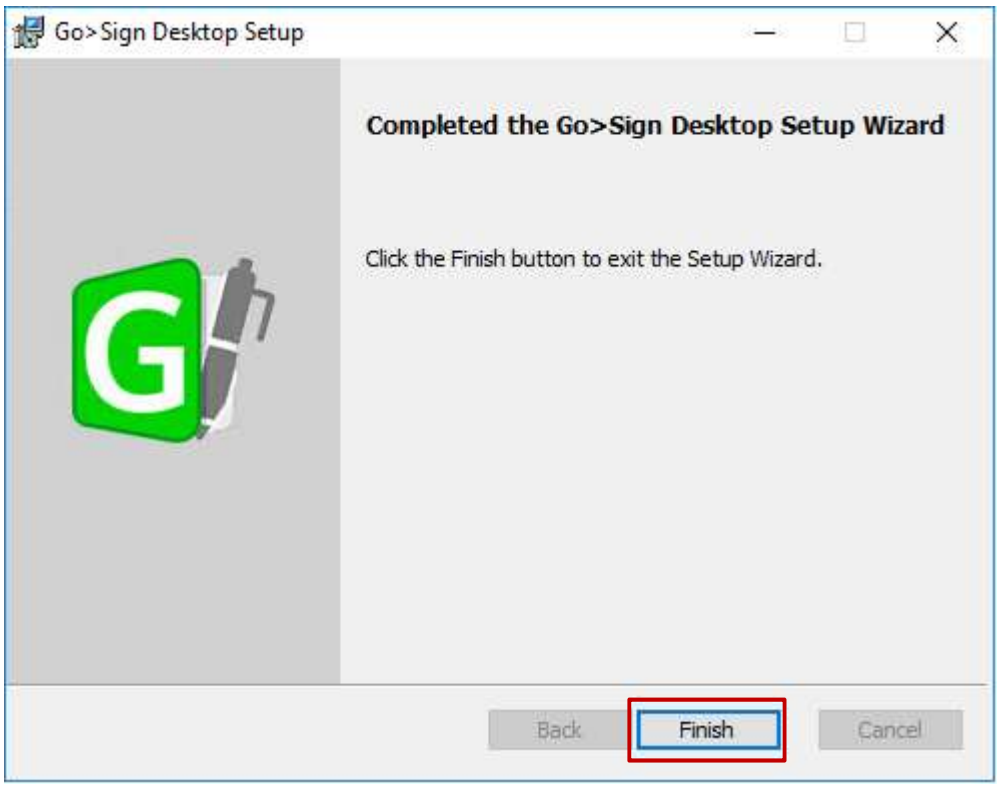
Click Next and accept End User License Agreement


Go>Sign Desktop Setup
End-User License Agreement
Please read the following license agreement carefully
ASCERTIA SERVER SOFTWARE LICENSE AGREEMENT
IMPORTANT - PLEASE READ CAREFULLY:
1.0 ACCEPTANCE.
Ascertia Limited ("Ascertia") is willing to license this software (the Software) and documentation (together the Product) in this installation package to you as an individual or as an authorised representative of the company or legal entity that will be using the Software only on condition that you accept all of the terms of this license
I accept the terms in the License Agreement
Print
Back
Next
Cancel


User Account Control
Do you want to allow this app from an unknown publisher to make changes to your device?
C:\appo\gsd-p6.9-17-11\ADSS-Go-Sign-Desktop-v6.9-Win64-MU.msi
Publisher: Unknown
File origin: Hard drive on this computer
Show more details
Yes
No

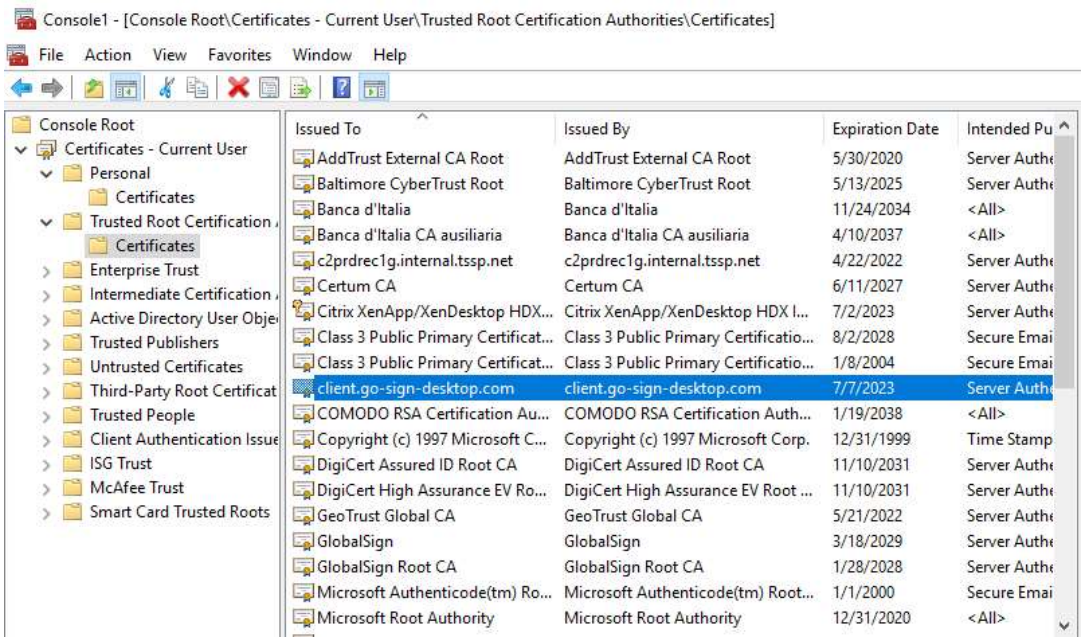

User Account Control
Do you want to allow this app to make changes to your device?
Windows Command Processor
Verified publisher: Microsoft Windows
Show more details
Yes
No


Security Warning
You are about to install a certificate from a certification authority (CA) claiming to represent:
client.go-sign-desktop.com
Windows cannot validate that the certificate is actually from "client.go-sign-desktop.com". You should confirm its origin by contacting "client.go-sign-desktop.com". The following number will assist you in this process:
Thumbprint (sha1): A72D784B 462FDF51 174963C0 1FF5E258 6373E9AA
Warning:
If you install this root certificate, Windows will automatically trust any certificate issued by this CA. Installing a certificate with an unconfirmed thumbprint is a security risk. If you click "Yes" you acknowledge this risk.
Do you want to install this certificate?
Yes
No

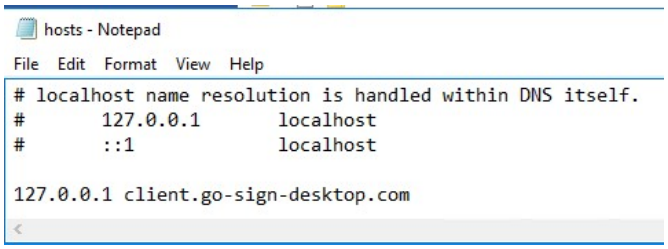


2.1.4 Post installation checks

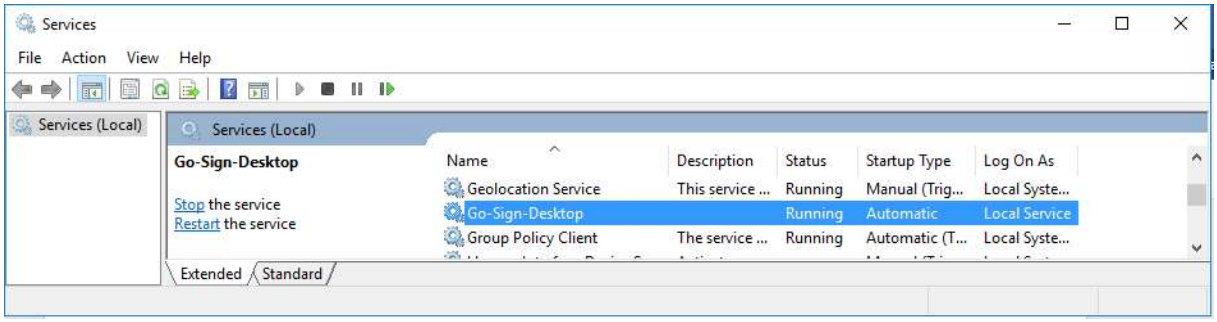
Check that certificate client.go-sign-desktop.com is imported in the (Administrator) User Certificate store by running *certmgr.msc* tool:

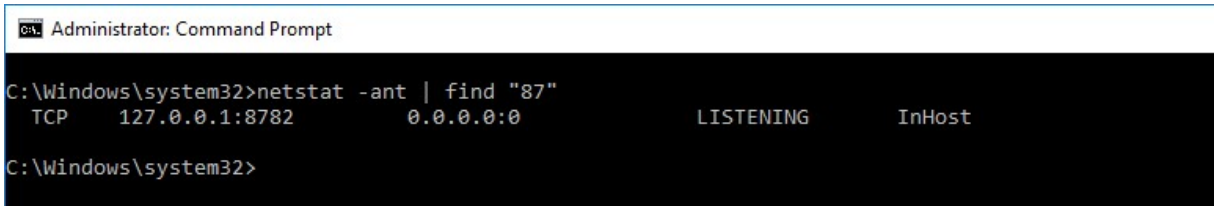


Check for the following definition in the host file:



Check service running and listening on 8782 (with Local Service account)





2.1.5 Publish applications GSD.exe and Chrome in Citrix Farm

This task is not required anymore as the following changes are applied to local machine registry:

[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\Ascertia]

"URL Protocol"="GSD"

[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\Ascertia\shell]

[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\Ascertia\shell\open]

[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\Ascertia\shell\open\command]

@="C:\\Program Files\\Ascertia\\Go-Sign-Desktop\\GSD.exe %1"

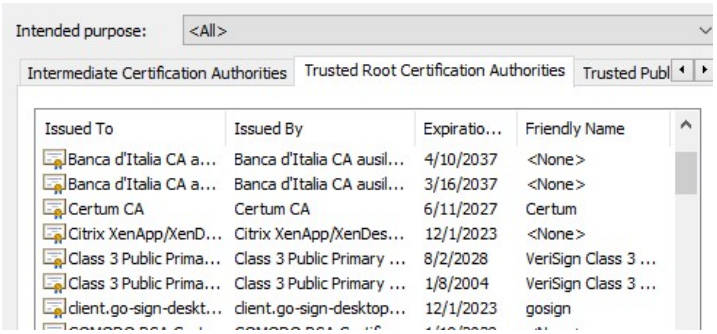
The above will allow browsers to correctly trigger the start of GSD user/child instances, when user will be prompted to do so during an NRO task.

Above settings allow to not perform any change to user registry settings and are valid both for persistent and non-persistent profiles terminal server environments.

2.2 GSD CLIENT TS installation– user actions

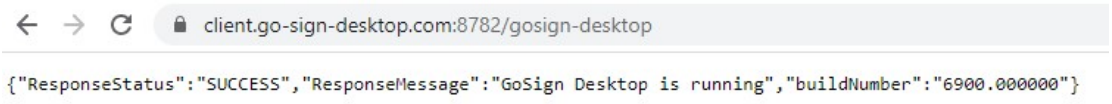
2.2.1 Import client.gosign certificate into WINDOWS-ROOT user keystore

This step is not required anymore as the gosign certificate is imported at computer level during the installation and is visible by the user in the browser keystore:



2.2.2 Check GSD running and start NRO task

Once all the above steps will be completed and GSD service running with Local Service user, business user could start an NRO task. By using the standard test URL, the expected result is:

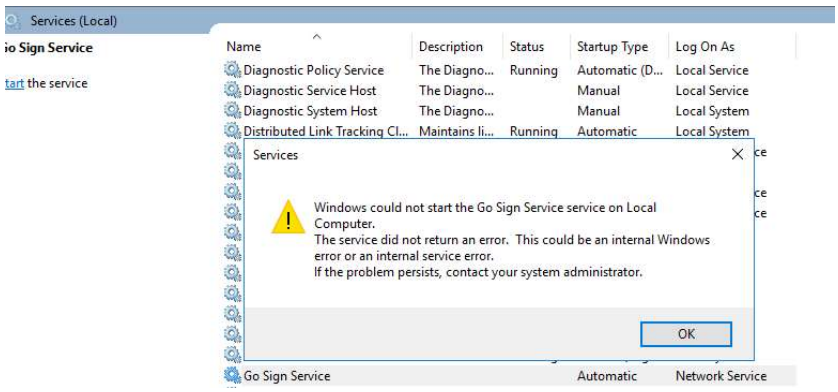


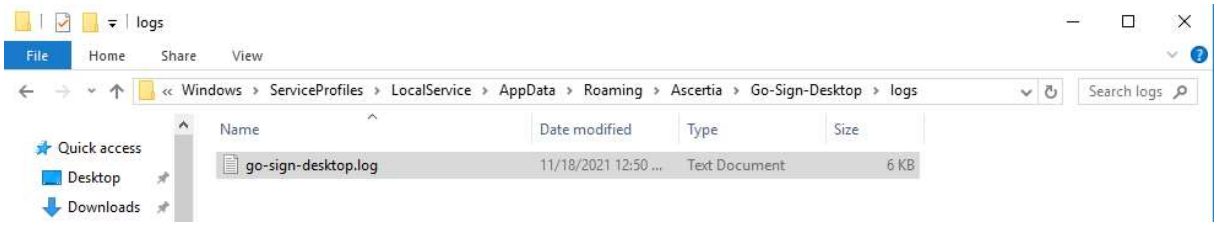
In order to properly perform NRO task, users are expected to allow execution of GSD user instance (before actual signature) in order to allow start of a child GSD application that will then communicate with the GSD service/parent instance. GSD child instances will listen on greater ports than the GSD service/parent one (8782) and will start dedicated Go-Sign-Desktop.exe application for each different web origin.

2.3 Issues

2.3.1 Service do not start

In case of issues with service start, please check and share Ascertia service logs from the following path C:\Windows\ServiceProfiles\LocalService\AppData\Roaming\Ascertia\Go-Sign-Desktop\logs





As said, when service correctly started, IT Administrator should see process listening on port 8782. The service / parent instance has to be started before any child/user instance.

2.3.2 Failed to load keystore issue during NRO task

In case user is displayed ”failed to load keystore” issue during NRO task, please double check and ensure that gosoign_app.properties file is present in Ascertia user folder and that user is allowed to properly write in this file (i.e. no GPO or User Account Control restricing access to it).

NRO will not work properly in case GSD application launched by the user during NRO task can not create and/or update the gosign_app.properties file.

2.3.3 Useful information for troubleshooting

In case of exceptions, users are requested to collect and share:

- browser console log and screenshots
- GSD service/parent log file
- GSD user/child instance log file

(Terminal server IT administrator support may be required).

2.3.4 Change logging level

By default, ADSS Go>Sign Desktop logging level is set to INFO. To enable detailed debug logging, follow these instructions:

1. Go to ADSS Go>Sign Desktop installation path C:\Program Files\Ascertia\Go-Sign-Desktop\app\conf\

2. Edit the gosign_desktop.properties file using a suitable text editor.

3. Change the value of the property GOSIGN_DESKTOP_LOG_LEVEL from INFO to DEBUG and save the file.

4. Stop and start ADSS Go>Sign Desktop service and perform again the NRO task

2.3.5 Local network issues related to new DSS URL

In case of local network exceptions in the browser (i.e. TUNNEL CONNECTION FAILED, NAME NOT RESOLVED) during first interaction with new Ascertia infrastructure: add DSS host certificates in browsers keyring (e.g Chrome and Firefox). Host names following:

SIA TST	esmig-tst-dss.u2a.sianet.sia.eu
SIA CRT	esmig-cert-dss.u2a.sianet.sia.eu
SIA PRD	esmig-dss.u2a.sianet.sia.eu
SWIFT TST	esmig-tst-dss.emip.swiftnet.sipn.swift.com
SWIFT CRT	esmig-cert-dss.emip.swiftnet.sipn.swift.com
SWIFT PRD	esmig-dss.emip.swiftnet.sipn.swift.com