Inextor IT setup guide . . . 5
Inextor IT setup guide: Installation process flow . . . 6
Installation Process Flow: Vault new installation . . . 8
Installation Process Flow: Vault upgrade . . . 9
Installation Process Flow: Vault disaster recovery . . . 10
Installation Process Flow: Gate Installation (Factory System) . . . 12
Installation Process Flow: Gate general/regulation configurations . . . 13
Installation Process Flow: Liz installation (Line System) . . . 14
Inextor IT Setup Guide: General Installation Guide . . . 16
General installation guide: Network engineers . . . 18
Vault network: Required firewall settings . . . 19
Gate network: Required firewall settings . . . 20
Gate network: Network bandwidth . . . 21
Gate network: Examples traffic Gate - Vault . . . 22
Liz network: Required firewall settings . . . 24
Vault network: REST API: GetSystemHealth . . . 25
Vault network: Mexico VPN deployment . . . 26
General installation guide: Database administrators . . . 29
Database administrators: SQL server quick summary . . . 30
SQL Server Installation . . . 31
SQL Server Summary: SQL Server Backup . . . 33
SQL Server Summary: Restore . . . 39
Database Administors: Vault Database Administration . . . 42
Database administrators: Contingency plans . . . 50
Database Contigency Plans: Always On . . . 51
Database Contigency Plans: Database mirroring . . . 53
Database administrators: Vault proxy database setup . . . 54
Database administrators: Advanced maintenance and optimization . . . 58
Database administrators: Database . . . 60
General installation guide: Integration developers . . . 66
Integration developers: Inextor Gate . . . 67
Gate Integration Developers: Operation Drivers . . . 68
Gate Integration Developers: Data sources and flows . . . 70
Gate Integration Developers: Product integration for the BasicHttpWcf operation driver . . . 71
How to configure an operation driver for https . . . 74
Dynamic Attributes Editor . . . 75
Creating or Editing Reports . . . 76
Decompose codes with Ait Service (Gate) . . . 78
Product Catalog Integration Views . . . 79
Integration Developers: Web Services List . . . 84
Vault web services . . . 87
Vault Web Service: AIT Web Service, V3 (Vault) . . . 88
Vault Web Service: VaultWS Web Service (Vault) . . . 89
Vault Web Service: EPCIS Query Web Service (Vault) . . . 92
Vault Web Service: TrackingInterfacesService Web Service (Vault) . . . 93

Vault Web Service: WCFCheckerService Web Service (Vault) . . . 94
Vault Web Service: PortalQueryService Web Service (Vault) . . . 95
Vault Web Service: AIT Web Service, V1 (Vault) . . . 96
Web services list: External code provider operation driver . . . 97
Web Service: Gate: ExternalCodeProviderWsProxy . . . 98
Web services list: Basic Http operation driver . . . 99
Web Service: Gate: ManufacturedProductInterface Web Service (Gate) . . . 100
Web Service: Gate: ProductionInterface Web Service . . . 101
Web Service: Gate: SKUInterface Web Service . . . 102
Web Service: Gate: GateProduction Web Service . . . 103
Web services list: Interfaces driver operation driver . . . 104
Operation Driver Interfaces: InterfacesGateServiceWsProxy . . . 105
Web services list: PMI MES operation driver . . . 106
PMI MES Operation Driver Web Service: PmiMesServiceWsProxy . . . 107
External Web Services . . . 108
Web Services List: Print On Demand (Webservices, Liz) . . . 109
Web Services List: External Service Operation Driver . . . 110
External Service Operation: Driver ExternalStockManagerProxy . . . 112
External Service Operation: ExternalProductionOrderManager . . . 113
External Service Operation: ExternalProductManager . . . 114
External Service Operation: ExternalLizManagerProxy . . . 115
External Service Operation: GateMasterData . . . 116
Integration Developers: Vault . . . 117
Vault Integration Development: Supported Browsers . . . 118
Vault Integration Development: Service HealthCheck . . . 119
Vault Integration Development: Decompose code with InterfacesService . . . 121
Vault Integration Development: Installing the QTSP Signing Proxy . . . 124
Vault Integration Development: Installing the QTSP Signing Proxy - Inexto managed . . . 131
Vault Integration Development: Decompose codes with Ait Service (Vault) . . . 138
General installation guide: Windows administrators . . . 144
Windows Administrators: Vault Proxy . . . 145
Vault Proxy: Prerequisites . . . 146
Vault Proxy: Web Application . . . 148
Vault Proxy: IIS Configuration . . . 149
Vault Proxy: Inextor Integration . . . 150
Windows Administrators: Certificate Management . . . 153
Certificate Tool . . . 154
Certificate Import for Id Issuers . . . 160
Legacy . . . 161
Certificate Manager . . . 162
Certificate Install (Windows Certification Store) . . . 172
Windows Administrators: Special Configurations for Windows . . . 173
Microsoft Failover Clustering . . . 174
Configuring the Schneider Style . . . 177
Enabling/Disabling Tobacco features . . . 178

How to Automate Resources (resx) Localization with the Translation Tool . . . 180
Windows Administrators: Master Data Services . . . 181
MDS Importer . . . 182
MDS Extractor . . . 185
SQL Server Master Data Services in Azure . . . 188
Windows Administrators: Gate . . . 199
Gate Installation . . . 200
Configuration Files . . . 209
Gate Administration . . . 210
Microsoft Failover Clustering Gate . . . 211
Configure a port on the Gate to be used by an Operation Driver . . . 212
How to Register a Server Certificate for WCF Operation Drivers . . . 213
Installing Report Files on the Gate Server . . . 214
User Authentication Gate . . . 215
Gate Service . . . 216
Windows Administrators: Vault . . . 222
Windows Service . . . 223
Vault Web applications installation . . . 227
Enable SSL . . . 233
Enable SSL on Multiple Servers . . . 237
Verify Installation . . . 238
Authentication Methods . . . 240
Configuration Files (Vault Configuration) . . . 241
Vault Service . . . 243
Vault Website . . . 250
Web Services . . . 251
Configure Vault to use NetNamedPipeBinding (Configuration) . . . 252
Configuring Web Clients with Limiting Access . . . 254
Exposing Vault Web Services . . . 256
Migration (Configuration) . . . 262
Security (Configuration) . . . 263
Patch for RRP . . . 265
Configuration Protection Tool . . . 266
Use Entra ID / Azure AD to authenticate users . . . 268
Create an Application Registration on Azure . . . 272
Vault Administration UI . . . 276
IIS Configuration - Setup . . . 279
Windows Administrators: Liz . . . 289
LIZ Installation and Setup . . . 290
Configuration Files (Liz Configuration) . . . 294
Liz Service - Configuration . . . 295
Liz Operation - Configuration . . . 298
Unattended and Mass Liz Deployment . . . 299
User Authentication on Liz Server . . . 303
General installation guide: IT infrastructure . . . 304

IT Infrastructure: On Premise vs. Cloud . . . 305
IT Infrastructure: Hardware and Software Requirements . . . 306
Hardware and Software Requirements: Vault Requirements . . . 307
Hardware and Software Requirements: Gate Requirements . . . 308
Hardware and Software Requirements: Liz Requirements . . . 310
IT Infrastructure: Vault for IT . . . 311
IT Infrastructure: Prerequisites for Implementation of the Vault . . . 312
General installation guide: Procedures . . . 315
General Installation Procedures: Technical Overview . . . 316
Manufacturers Checklist . . . 318
General Installation Procedures: Upgrade Rules . . . 323
General Installation Procedures: Vault Procedures . . . 324
Vault Procedures: Introduction . . . 325
Vault Procedures: Migration Checker v4 to Vault v10 . . . 326
Vault Procedures: Contingency Procedure . . . 332
Vault Procedures: Azure Managed Identities . . . 335
Vault Procedures: Configuration . . . 338
Vault Procedures: Disaster Recovery . . . 339
Vault Procedures: Setup Vault . . . 343
General Installation Procedures: Gate Procedures . . . 348
Gate Installation Procedures: Introduction . . . 349
Gate Installation Procedures: Configuration . . . 350
Gate Installation Procedures: Disaster Recovery Gate . . . 351
Gate Installation Procedures: Contingency Procedures . . . 354
Gate Installation Procedures: Prerequisites . . . 356
Gate Setup: Installation Sequence . . . 358
General installation procedures: Liz procedures . . . 360
Liz Procedures: Introduction . . . 361
Liz Procedures: Configuration . . . 362
Liz Procedures: Disaster Recovery Liz . . . 363
Liz Procedures: Contingency Procedures Liz . . . 367
Liz Procedures: Prerequisites for Implementation of the LIZ . . . 368
General installation procedures: Vault proxy procedures . . . 369
New Installation Steps for Vault Proxy Setup . . . 370
General Installation Procedures: Migrating from Codentify to Inextor . . . 371
Migration (COG v4 to Liz v10) . . . 374
Codentify Inextor Migration: Migration (MCGv4 to Gate v10) . . . 378

Inextor IT setup guide

- Inextor IT setup guide: Installation process flow
- Inextor IT Setup Guide: General Installation Guide

Inextor IT setup guide: Installation process flow

Document Information Table	
Document Type	Index
Classification	Company Confidential
Document ID	INEX-INDX-00004
Version Number	v1.0
Owner	@Michael Houlding Lead Technical Writer - Engineering

Introduction

Document INEX-INDX-00004: Inextor Installation Process Flow Index, provides a central point of reference for the required setup Use Case scenarios necessary to be performed during site implementation of Inextor a product of the Inexto SA Suite range of serialization, authentication and track and trace solutions.

Purpose

The purpose of INEX-INDX-00004 is to capture the requirements necessary to successfully set up Inextor correctly to support client business needs based on specific Use Case scenarios.

Scope

The scope of this document is limited to the Use Case Scenarios relating to the setup of Inextor product for clients of Inexto SA.

Target audience

The intended target audience of this document to complete the Inextor installation and configuration is as follows:

- Hands-on IS operators and administrators
- Developers and Testers
- Solution Architects and Integration Experts

This document is not intended for:

- Business Managers
- Sales and Marketing
- Quality Management and Certificate Issuers

Use case installation index table

Use Case & Document tile	Document ID / link

Inextor Installation Process Flow Diagram - **Vault** New Installation	INEX-DIA-00001
Inextor Installation Process Flow Diagram - **Vault** Upgrade	INEX-DIA-00002
Inextor Installation Process - **Vault** Disaster Recovery	INEX-DIA-00003
Inextor Installation Process - Configuration (**Vault** configuration)	Page link
Inextor Installation Process - Migration Checker v4 to **Vault** v10	Page link
Inextor Installation Process Flow - **Gate** Installation (Factory System)	INEX-DIA-00004
Inextor Installation Process Flow Diagram - **Gate** General/Regulation configurations	INEX-DIA-00005
Inextor Installation Process - **Gate** Disaster Recovery	Page link
Inextor Installation Process Diagram - **Liz** Installation (Line System)	INEX-DIA-00006
Inextor Installation Process Flow Diagram - **Liz** Disaster Recovery	Page link
Inextor Installation Process Flow Diagram - Configuration Files (**Liz** Configuration)	Page link
Inextor Installation Process - Migration (COG v4 to **Liz** v10)	Page link

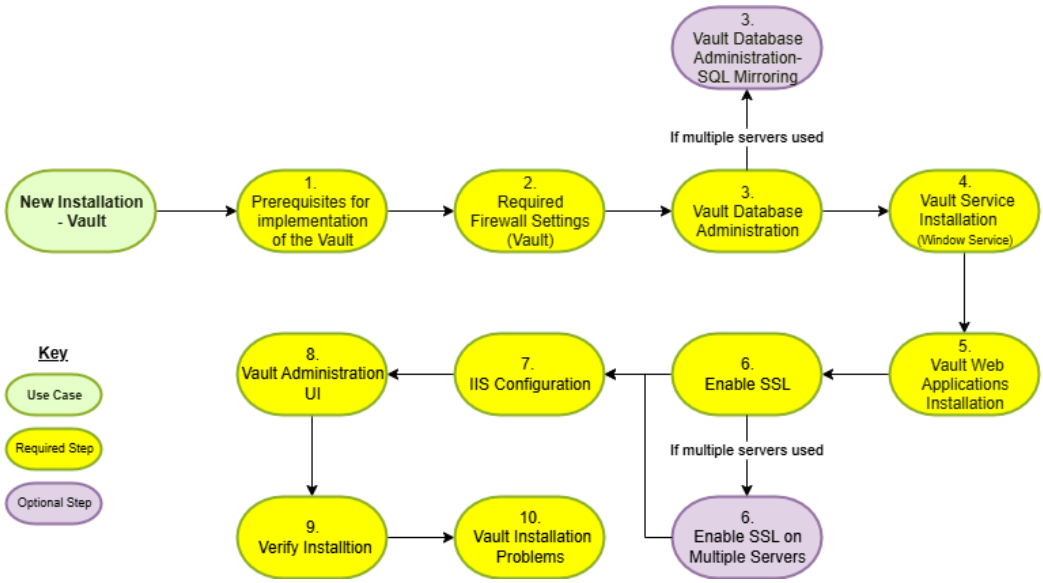
Reference documents

Document ID	Document Title	Revision
SOP 8.6	Technical Documentation Writing	1.0
INEX-TMP-00018	Installation Process Flow Index Template	1.0

****END OF DOCUMENT****

Installation Process Flow: Vault new installation

Document Information Table	
Document Type	IT Setup
Classification	Company Confidential
Document ID	INEX-DIA-00001
Version Number	v1.0
Owner	Michael Houlding Lead Technical Writer - Engineering



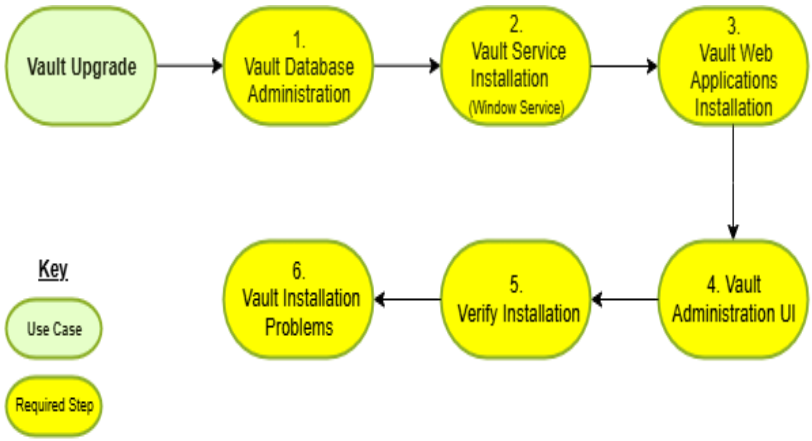
Reference documents

Document ID	Document Title	Revision
SOP 8.6	Technical Documentation Writing	v1.0
INEX-TMP-00019	Diagram Template	v1.0

****END OF DOCUMENT****

Installation Process Flow: Vault upgrade

Document Information Table	
Document Type	IT Setup
Classification	Company Confidential
Document ID	INEX-DIA-00002
Version Number	v1.0
Owner	Michael Houlding Lead Technical Writer - Engineering



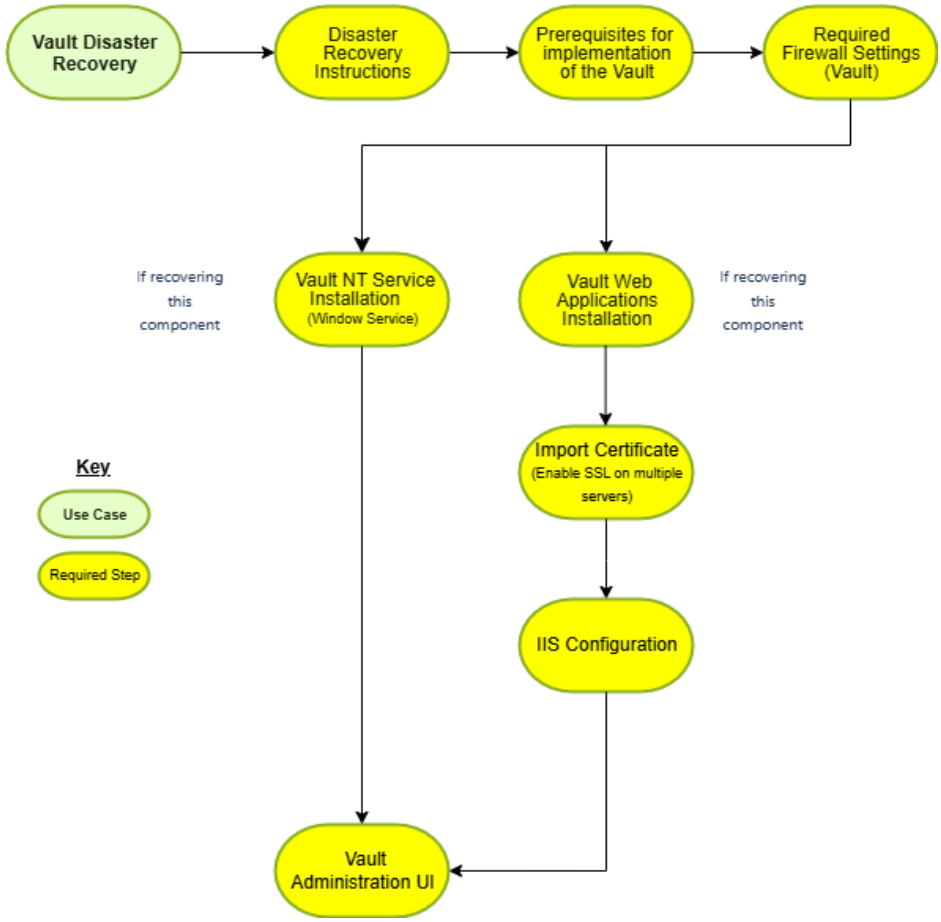
Reference documents

Document ID	Document Title	Revision
SOP 8.6	Technical Documentation Writing	v1.0
INEX-TMP-00019	Diagram Template	v1.0

****END OF DOCUMENT****

Installation Process Flow: Vault disaster recovery

Document Information Table	
Document Type	IT Setup
Classification	Company Confidential
Document ID	INEX-DIA-00003
Version Number	v1.0
Owner	Michael Houlding Lead Technical Writer - Engineering



Reference documents

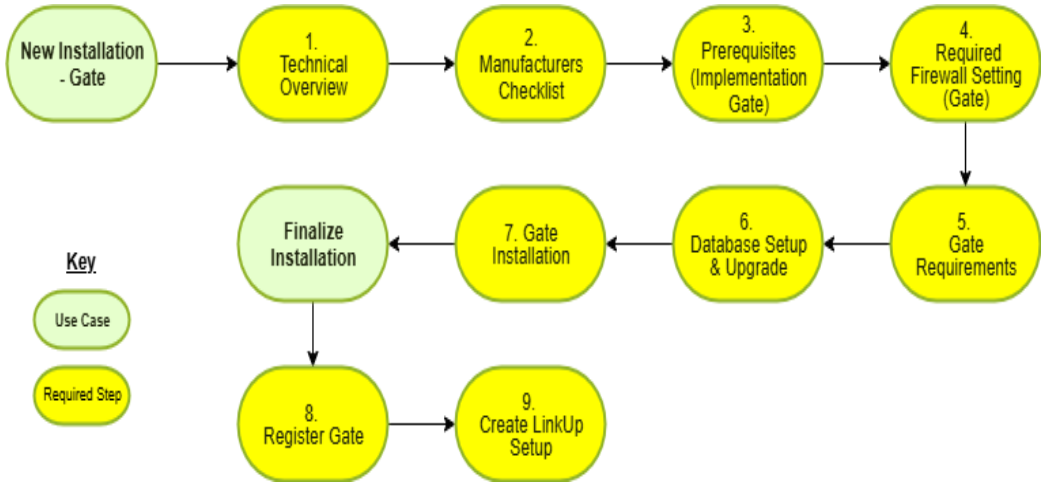
Document ID	Document Title	Revision
SOP 8.6	Technical Documentation Writing	v1.0

INEX-TMP-00019	Diagram Template	v1.0

****END OF DOCUMENT****

Installation Process Flow: Gate Installation (Factory System)

Document Information Table	
Document Type	IT Setup
Classification	Company Confidential
Document ID	INEX-DIA-00004
Version Number	v1.0
Owner	Michael Houlding Lead Technical Writer - Engineering



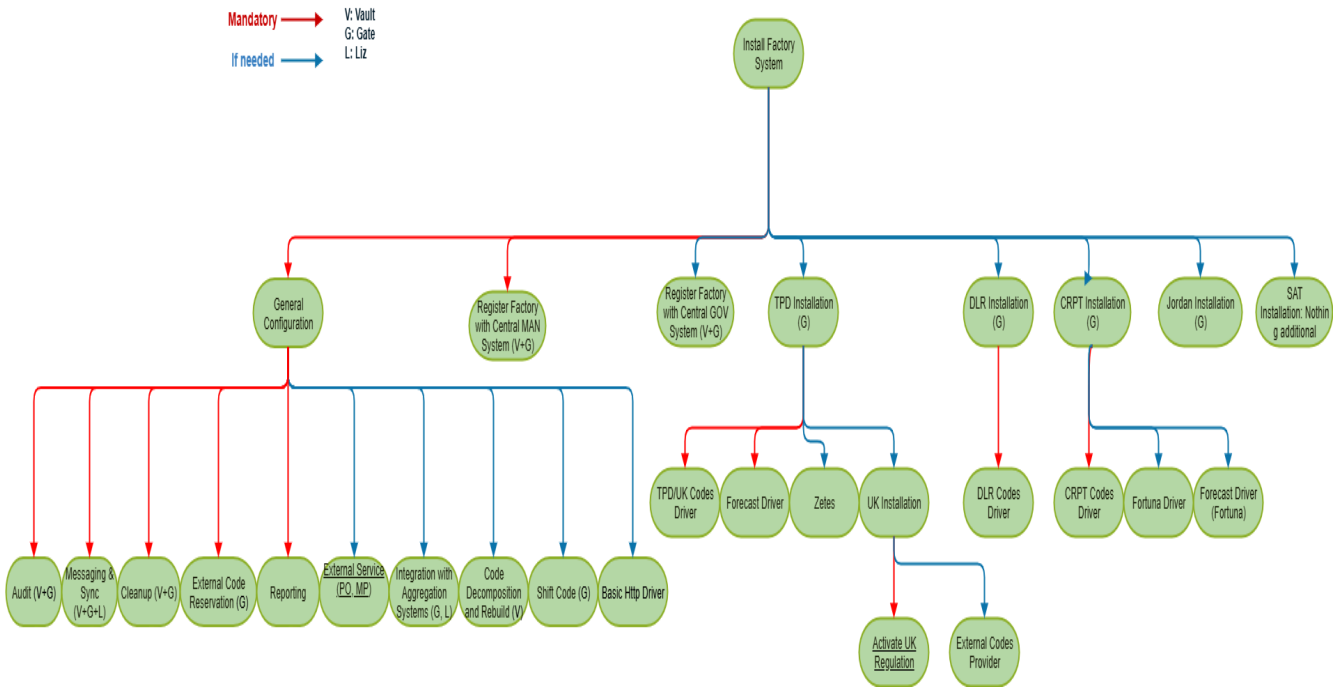
Reference documents

Document ID	Document Title	Revision
SOP 8.6	Technical Documentation Writing	v1.0
INEX-TMP-000x	Diagram Template	v1.0

****END OF DOCUMENT****

Installation Process Flow: Gate general/regulation configurations

Document Information Table	
Document Type	IT Setup
Classification	Company Confidential
Document ID	INEX-DIA-00005
Version Number	v1.0
Owner	Nathalie Chiva Lead Technical Writer - Engineering



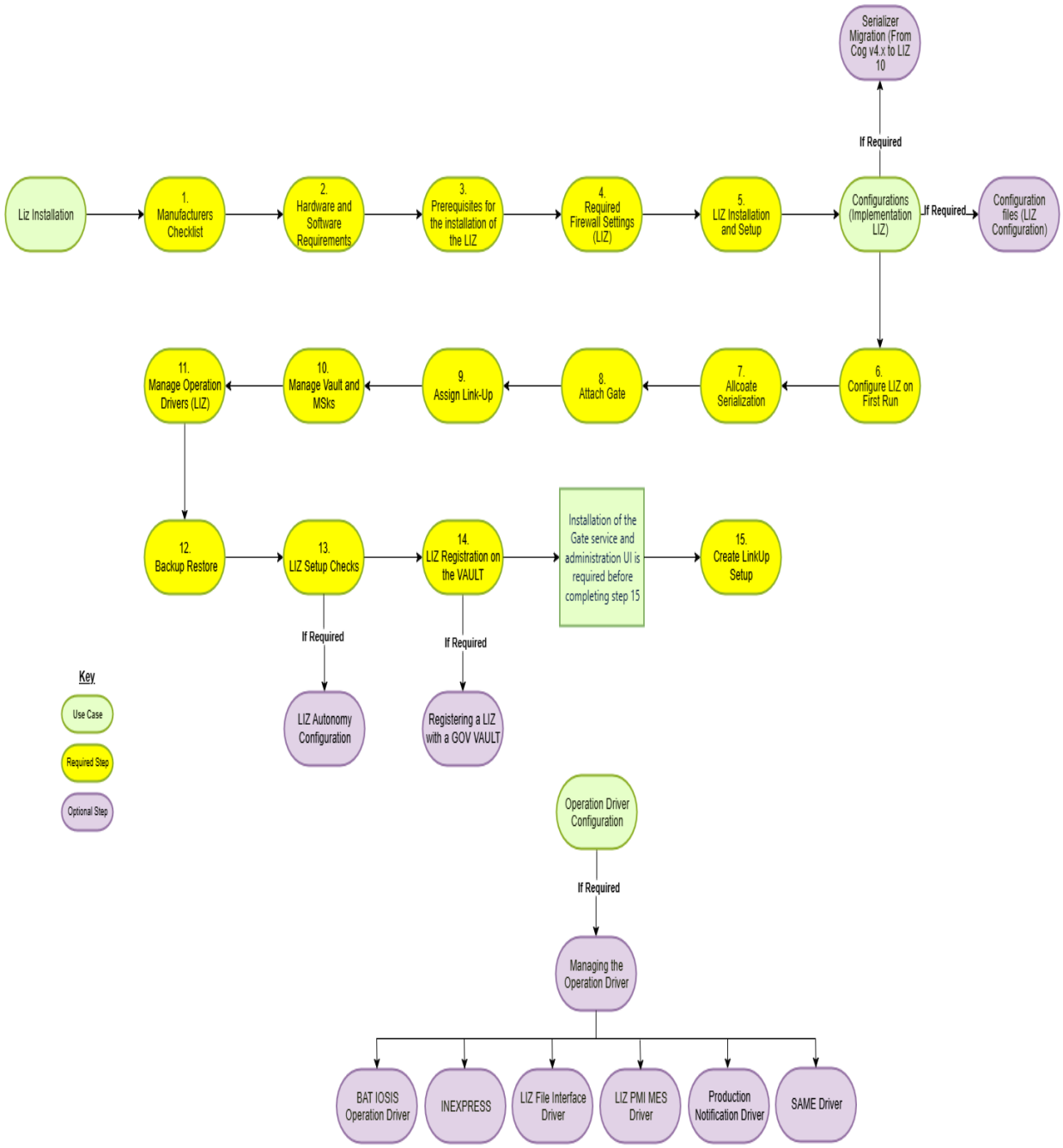
Reference documents

Document ID	Document Title	Revision
SOP 8.6	Technical Documentation Writing	v1.0
INEX-TMP-00019	Diagram Template	v1.0

****END OF DOCUMENT****

Installation Process Flow: Liz installation (Line System)

Document Information Table	
Document Type	IT Setup
Classification	Company Confidential
Document ID	INEX-DIA-00006
Version Number	v1.0
Owner	Michael Houlding Lead Technical Writer - Engineering



Reference documents

Document ID	Document Title	Revision
SOP 8.6	Technical Documentation Writing	v1.0
INEX-TMP-00019	Diagram Template	v1.0

****END OF DOCUMENT****

Inextor IT Setup Guide: General Installation Guide

Document Information Table	
Document Type	IT Setup Guide
Classification	Company Confidential
Document ID	INEX-STUP-00007
Version Number	v1.0
Owner	@Michael Houlding Lead Technical Writer - Engineering

Introduction

Document INEX-STUP-00007: Inextor General Installation Guide, outlines the required steps necessary to be performed during the planning and installation of Inextor a product of the Inexto SA Suite range of serialization and track and trace solutions.

Purpose

The purpose of INEX-STUP-00007 is to capture the required planning and installation requirements necessary to successfully set up Inextor correctly to support client business needs.

Scope

The scope of this document is limited to the functional use of the Inextor product by clients of Inexto SA.

Target Audience

The intended target audience of this document to complete the Inextor installation and configuration is as follows:

- Hands-on IS operators and administrators
- Developers and Testers
- Solution Architects and Integration Experts

This document is not intended for:

- Business Managers
- Sales and Marketing
- Quality Management and Certificate Issuers

Document content

⚠ For Advanced Content Navigator for Confluence to display, you need to allow the app to access Atlassian products on your behalf.

Allow access

Reference documents

Document ID	Document Title	Revision
SOP 8.6	Technical Documentation Writing	1.0
INEX-TMP-00004	IT Setup Guide Template	1.0

Revision history

Version Number	Section	Description of change	Justification of change
v1.0	All	Original Document	New Document

General installation guide: Network engineers

- Vault network: Required firewall settings
- Gate network: Required firewall settings
- Gate network: Network bandwidth
- Gate network: Examples traffic Gate - Vault
- Liz network: Required firewall settings
- Vault network: REST API: GetSystemHealth
- Vault network: Mexico VPN deployment

Vault network: Required firewall settings

Please allow the following ports as exceptions in the firewall for incoming traffic:

Server	Name	Port	Type	Note
Presentation Servers	Internet or Intranet	443	TCP	
Business Servers	Vault Service	8086	TCP	1,3
Data Servers	SQLServer	1433	TCP	
Data Servers	Mirroring	5022	TCP	2

Note 1: If Business and Presentation Servers components are deployed on the same server, then the Vault Service port does not have to be opened. **Note 2:** Only required if using SQL Server mirroring with or without witness **Note 3:** The port numbers could be any valid port number. The one described in the table are the default ones and can be changed. In order to change the Vault Service port, you have to change the port number on the business server (see Vault Service Listen Port) and on the presentation servers (see Vault Web site - Web.config" and Web Services - Web.config").

Gate network: Required firewall settings

On server platforms, the Windows firewall is typically disabled. However, if there is a hardware firewall between the Gate and Liz VLANs, the same rules should be applied. We recommend creating a dedicated Group Policy Object (GPO) that includes the following firewall exceptions:

Exception Name	Network Port	Type	Scope	Description
Gate Liz Standard	8088	TCP	*	Input (Gate Service)
Gate Standard	8089	TCP	*	Input (Gate UI)
Gate Liz Non-Standard	8188	TCP	*	Input (Gate Service)
Gate Non-Standard	8189	TCP	*	Input (Gate UI)
Gate Liz Alternative	9088	TCP	*	Input (Gate Service)
Gate Alternative	9089	TCP	*	Input (Gate UI)

There are three sets of TCP ports available. You may already be utilizing one set of TCP ports for a previous version of Inextor (i.e., Standard). Consequently, the Non-Standard and Alternative sets are TCP ports recognized by the DCTA Technology Center for upgrading INEXTOR.

Gate network: Network bandwidth

The GateService application exchanges data with the follow systems:

- Gate SQL Server database
- Liz(s)
- Vault(s)

> **Warning about the Gate SQL Server location**
>
> it is critical that the Gate SQL Server is located in close proximity to the GateService Application.
>
> Indeed, the GateService is chatty with his database and a high latency would severely impair the performance of the Gate but also all the Lizs connected to the Gate.

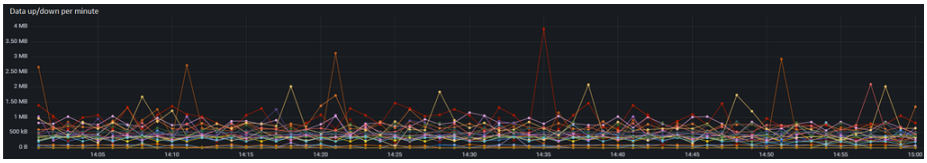
Data usage between Gate and Vault

The data consumption is primarily influenced by the configuration of the periodic jobs sending real-time status data: if data is updated more frequently, background bandwidth consumption will increase. For example, the "Liz Health Report" job is configured to run every 10 minutes out of the box.

The downloading of external codes and production data reporting are of course proportional to the production volume of the factory. But these factors are somehow less important than the previous point.

Minimal bandwidth up & down : 100 kBytes / s

Recommended bandwidth up & down : 1 MBytes / s

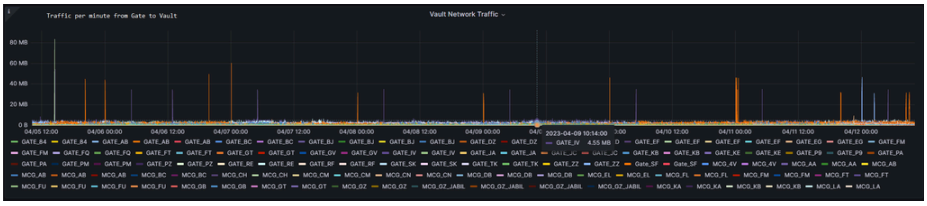


Gate network: Examples traffic Gate - Vault

	Regular	RRP
Krakow, Poland	PC	PA
MTB, Italy		ZZ, IC
Romania	RB	RRP_RB
Greece		GT
+ 5 other locations with only a single or a few link-ups.		

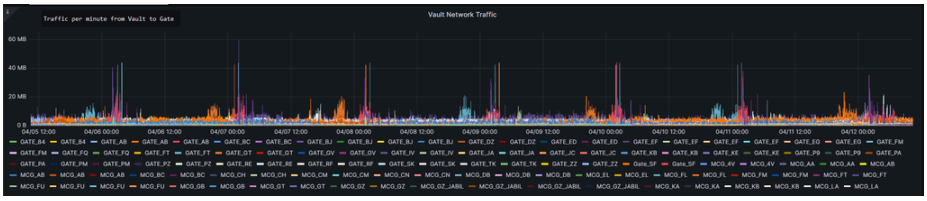
Upstream: Gate to Vault

5MB/min = 80 kbytes / s



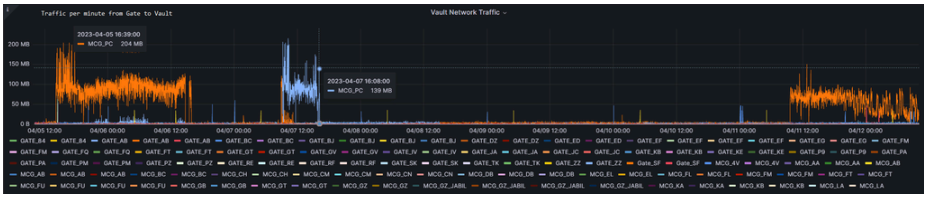
Downstream: Vault to Gate

10MB/min = 160 kbytes / s



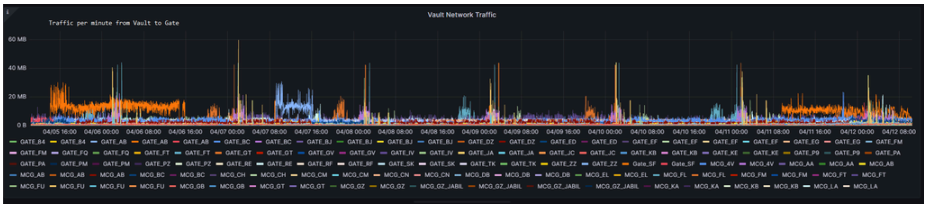
Upstream: Gate to Vault (outlier)

100MB/min = 1,6 MB / s



Downstream: Vault to Gate (outlier)

15MB/min = 250 kbytes / s


Vault Network Traffic
Traffic per minute from Vault to Gate
60 MB
40 MB
20 MB
0 B
04/05 16:00
04/06 00:00
04/06 08:00
04/06 16:00
04/07 00:00
04/07 08:00
04/07 16:00
04/08 00:00
04/08 08:00
04/08 16:00
04/09 00:00
04/09 08:00
04/09 16:00
04/10 00:00
04/10 08:00
04/10 16:00
04/11 00:00
04/11 08:00
04/11 16:00
04/12 00:00
04/12 08:00

Liz network: Required firewall settings

Inexto SA recommends establishing a dedicated GPO (Group Policy Object) that includes the following firewall exceptions:

Exception Name	Network Port	Type	Scope	Description
Liz MES Standard	8001	TCP	*	Input (PmiMes Op. Driver)
Liz MES Alternative	8002	TCP	*	Input (PmiMes Op. Driver)
Liz Gate Standard	8087	TCP	*	Input (Liz UI, Liz Service)
Liz Gate Non-Standard	8187	TCP	*	Input (Liz UI, Liz Service)
Liz Gate Alternative	9087	TCP	*	Input (Liz UI, Liz Service)

There are three types of TCP ports available. You may already be utilizing one set of TCP ports for a previous version of Inextor (i.e., Standard). Consequently, the Non-Standard and Alternative TCP ports are recognized by the DCTA Technology Center for upgrading Inextor.

MES stands for Manufacturing Execution System, a crucial component in modern manufacturing environments. It serves as a bridge between enterprise resource planning (ERP) systems and the shop floor, enabling real-time monitoring and control of production processes. By providing detailed insights into manufacturing operations, MES helps organizations improve efficiency, reduce costs, and enhance product quality.

Key functionalities of MES include tracking production progress, managing inventory, scheduling tasks, and ensuring compliance with industry standards. By leveraging MES, manufacturers can make informed decisions, respond swiftly to changes in demand, and optimize their overall production performance.

Vault network: REST API: GetSystemHealth

Introduction

This service allows users to verify the current status of the Vault and its Database, ensuring they have up-to-date information on both, which is crucial for effective data management and security.

Specific packages used

The service utilizes the *Inexto.Common.Healthcheck* package.

Logic

In order:

1. *GetSystemHealth()* gets the IVaultWS
2. it calls *GetHealthCheck()* (from *VaultService*) to verify the status of the Database
3. GetHealthCheck() makes a query (*GetBatchesByLizIdAndStartDate* with basic parameters, it could have been any other simple query)
   a. if the query succeeds then the database is working and a new *ComponentHealthInformation* with *Status.Operational* is returned
   b. if the query was not successful an exception is captured and the method returns a new *ComponentHealthInformation* with *Status.MajorOutage*

<to be continued>

Vault network: Mexico VPN deployment

Introduction

If the Vault needs to connect with the Mexican authorities (SAT), a VPN connection must be established between the Azure network and Alestra, the mandated provider in Mexico. This deployment is a collaborative effort between Inexto and Alestra.

Azure VPN creation

To set up the VPN components in Azure, please refer to the following document: VPN Connection between PMI Vault Servers - Alestra - SAT Authorities. It is essential to coordinate with Alestra for specific technical details, including the shared key and IP ranges.

Connection testing

Once Inexto and Alestra have set up the VPN, a testing phase will commence. It may require several iterations before the VPN connection is successfully established.

Azure metrics

On the Azure portal, you can monitor various metrics that offer valuable insights into the status of your connection. These metrics help you determine whether the connection is functioning properly.

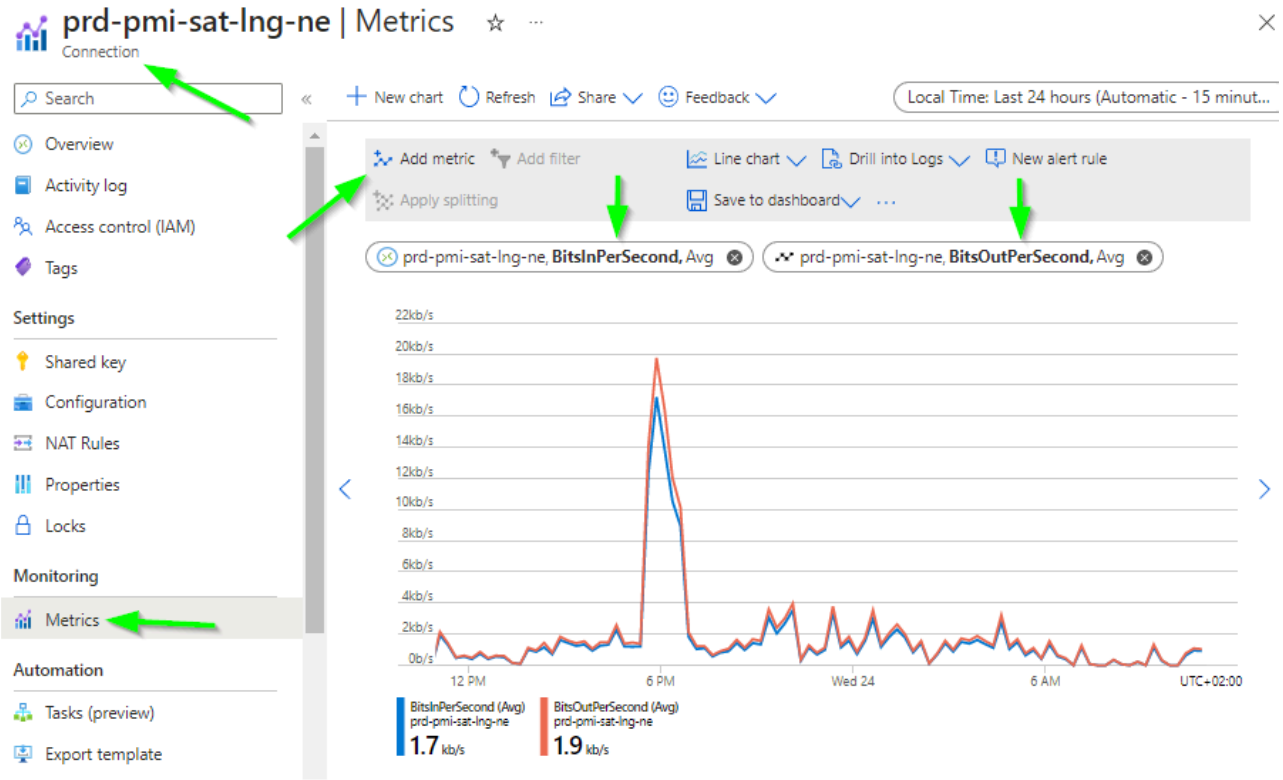


Figure 1: VPN Connection component. On the Metrics page, incoming and outgoing bits per second can be monitored.

In the upcoming section, when you send requests from Azure to Alestra, you will observe **outbound** data. Conversely, when Alestra sends requests to our Azure virtual machine, you will see **inbound** data.

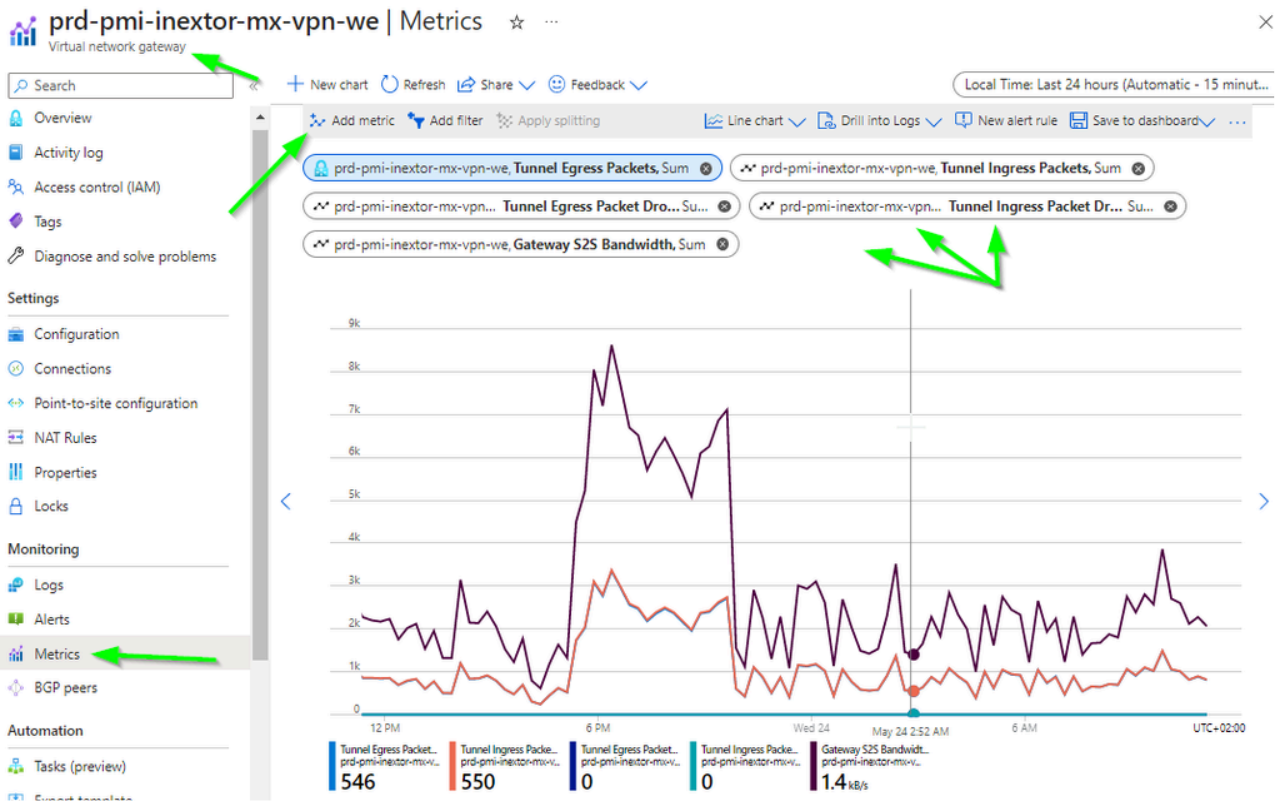


Figure 2: On the Virtual Network Gateway component, other interesting metrics can be monitored. Some interesting metrics are the egress (outbound) and ingress (inbound) packets, the dropped packets, the bandwidth for site-to-site (S2S).

Ping

To test the VPN connection, begin by attempting to ping machines across the networks.

Azure → Alestra

Connect to the Vault's virtual machine using remote desktop. Once connected, launch PowerShell. Utilize the Test-NetConnection command to perform a ping test. Even if the ping fails, it will still provide valuable information regarding the specific step that encountered an issue. Additionally, monitor the metrics to gather further insights.

```
Test-NetConnection 189.206.117.123 -port 443 -InformationLevel "Detailed"
```

Code 1: The port is the default HTTPS one. Use the IP address that the Vault is supposed to use. It is a public IP exposed by Alestra. Alestra will redirect the call to the SAT server.

Another useful testing tool is tracert.

```
tracert -h 15 189.206.117.123
```

Code 2: This command will show you the routers on the route taken by the request. The option -h can be used to limit the number of hops.

Alestra → Azure

Request Alestra to ping the Vault's machine. Keep an eye on the metrics to check for any inbound packets.

Ent-to-end test

Once you have confirmed that the connection is open in both directions, you can proceed with an end-to-end test. If you do not have a QA environment available, you will need to utilize the PRD. In this situation, it is crucial to ensure that there are no open orders or production reports.

1. Configure "the Vault to connect to SAT.
2. Create a code order request for a small quantity. Use a random payment reference.
3. The request should return an error, because we didn't use a real payment. The error should come from SAT and explain why the request was refused.
4. If you get a connection error, this means something is not yet properly configured.

General installation guide: Database administrators

- Database administrators: SQL server quick summary
- Database Administors: Vault Database Administration
- Database administrators: Contingency plans
- Database administrators: Vault proxy database setup
- Database administrators: Advanced maintenance and optimization
- Database administrators: Database

Database administrators: SQL server quick summary

- SQL Server Installation
- SQL Server Summary: SQL Server Backup
- SQL Server Summary: Restore

SQL Server Installation

General

For installing SQL Server you should use the Installation guide provided from Microsoft. This section only serves as an aid for installing SQL Server.

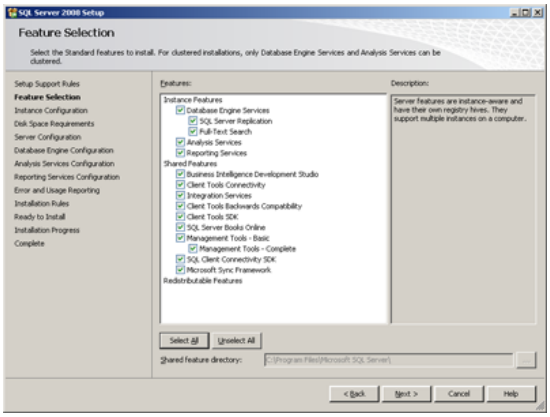
Prerequisites

If some software, required by SQL Server, is missing, it has to be installed before installing the SQL Server. Most such components are provided on the installation disk for SQL server and the verification of missing components is done during the installation process.
Refer to INEXTOR Requirements to check the list of compatible SQL versions.

SQL Server installation

This section describes the important configuration parameters, but does not go through the entire installation process. The screenshots are taken from SQL server 2008. For instructions about more recent versions of SQL server, please refer to Microsoft documentation.

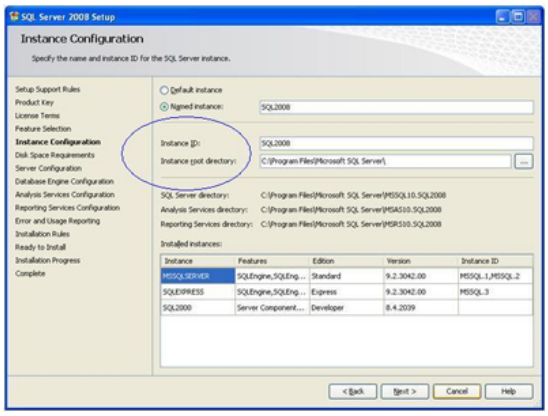
Feature selection



Not all features are necessary for Inextor. The following features can be unchecked:

- Analysis Services
- Reporting Services
- Business Intelligence development Studio
- Client Tools Connectivity
- Integration Services
- Client Tools Backwards Connectivity
- Client Tools SDK
- SQL Server Books Online
- Management Tools – **These are however needed if it is necessary to administer the SQL server locally**
- SQL Client Connectivity SDK
- Microsoft Sync Framework

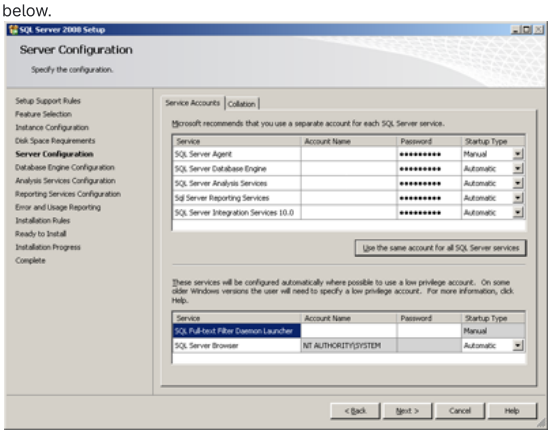
Instance naming



Default instance:
If the SQL server being installed is the first instance of an SQL server on the computer the default instance can be used. Otherwise a new instance has to be created by selecting Named instance and specifying an instance name e.g. MSSQL2019. If the default instance is not used the name instance has to be a part of the connection string in the config file for the Vault and the Gate, see documents Setup the Vault and Setup the Gate.
Instance ID:
By default, the instance name is used as the Instance ID. This is used to identify installation directories and registry keys for your instance of SQL Server. This is the case for default instances and named instances. For a default instance, the instance name and instance ID would be MSSQLSERVER. To use a non-default instance ID, specify it in the Instance ID field.

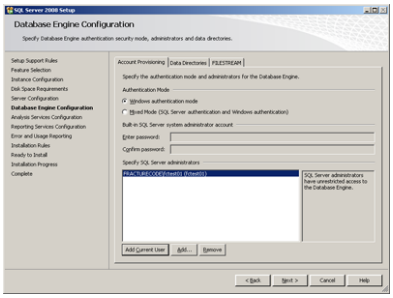
Service account

This part requires information on which account should run the services. Depending on the chosen features this dialog might display other services than the ones listed below.



If the SQL server is going to run under a windows account it is preferable to specify it during installation since it will provide the necessary rights to the account so that the service has minimum access to resources required by SQL server. If the account(s) is/are to be changed after installation, the account(s) specified will have to be a member of the necessary Windows groups on the server. Refer to SQL server documentation for details. If desired, the service account can be specified to run under NT AUTHORITY\SYSTEM which will give the service the rights of the local system account. However if the service requires network resources it will need to be a Windows account. Network resources could be required for backup to network drives, mirroring (if not using certificates for authentication) and other network-related tasks.

Authentication mode



This dialog specifies the authentication mode which can be either Windows or Mixed. Select Mixed Mode if you also want to be able to log on with an SQL Server account such as "sa".
The other important part is to decide which account is to be the Administrator of the SQL server.

SQL Server Summary: SQL Server Backup

General

There are several ways of generating backups. This section describes the following methods:

- Scripts and Task Manager
- SQL Server Agent
- Log Shipping

Scripts and Task Scheduler

If the principal and mirror databases run on SQL Server Express Servers, then the backup cannot be handled by SQL Server Agent, since there is no SQL Server Agent on SQL Express. The backup jobs can then be handled by a scheduled task – and here the Task scheduler in Windows is described. Optionally the SQL files could be stored procedures in the SQL server.

First create the folder C:\Backup

Full Database Backup

Create a .bat file and a .sql file and save them on the SQL Server (e.g. C:\Scripts\backup.bat and C:\Scripts\backup.sql)

The backup.bat file should contain the following line:

```
sqlcmd -S(local)\<INSTANCENAME> -E -ibackup.sql(remove \<INSTANCENAME> if no instance name)
```

The backup.sql file should contain the following:

```
DECLARE @name VARCHAR(50)       -- database name
DECLARE @path VARCHAR(256)      -- path for backup files
DECLARE @fileName VARCHAR(256)  -- filename for backup

SET @path = 'C:\Backup\ '
SET @name = '<database_name>'
BEGIN
    SET @fileName = @path + @name +'.bak'
    BACKUP DATABASE @name TO DISK = @fileName
END
```

If the backup file name also should contain the day number, the backup.sql file should contain the following:

```
DECLARE @name VARCHAR(50)       -- database name
DECLARE @path VARCHAR(256)      -- path for backup files
DECLARE @fileName VARCHAR(256)  -- filename for backup
DECLARE @fileDate VARCHAR(20)   -- used for file name

SELECT @fileDate = CONVERT(CHAR(2), GETDATE(), 4)
SET @path = 'C:\Backup\ '
SET @name = '<database_name>'

BEGIN
    SET @fileName = @path + @name +'_' + @fileDate + '.bak'
    BACKUP DATABASE @name TO DISK = @fileName
END
```

Make sure to copy these backup files to another location also.

Make sure that the Task Scheduler Service is running and that it starts automatically.

Create a new Scheduled Task, using the Task Scheduler on Windows. The new task launches the backup.bat file and is run daily.

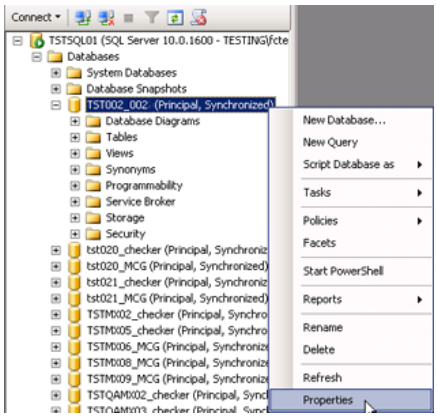
Transaction Log Backup (separate files)

If the Recovery Model for the database is “full”, then it is necessary to create Transaction Log Backups also. This must be done for the Vault and Gate databases.

To check the Recovery Model do the following:

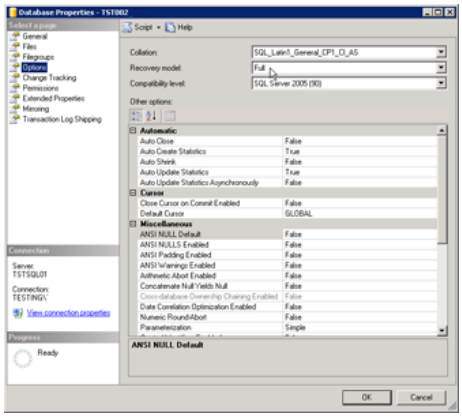
Start “SQL Server Management Studio.

Explore “Databases”, right-click on the database and select “Properties”



Database properties

Click on “Options” in the menu to the left. The Recovery Model can be found in row two.



Database properties Options

Add the following line to the backup.bat file:

```
del C:\Backup\<database_name>_*_log.*
```

Create a .bat file and a .sql file and save them on the SQL Server (e.g. C:\Scripts\backuplog.bat and C:\Scripts\backuplog.sql)

The backuplog.bat file should contain the following line:

```
sqlcmd -S(local)\<INSTANCENAME> -E -ibackuplog.sql #(remove \<INSTANCENAME> if no instance name)
```

The backuplog.sql file should contain the following:

```
DECLARE @name VARCHAR(50)       -- database name
DECLARE @path VARCHAR(256)      -- path for backup files
DECLARE @fileName VARCHAR(256)  -- filename for backup
DECLARE @fileDate VARCHAR(20)   -- used for file name

SELECT @fileDate =CONVERT(CHAR(2), GETDATE(), 8)
SET @path ='C:\Backup\ '
SET @name ='<database_name>'
BEGIN
    SET @fileName = @path + @name + @fileDate + '_log.bak'
    BACKUP LOG @name TO DISK= @fileName
END
```

Here the name of the backup file made at 2 PM for database “mydatabase” will be “mydatabase14_log.bak”.

Create a Scheduled Task for the backuplog.bat file also. It should run once every hour.

Make sure to copy these backup files to another location also.

Transaction Log Backup (same file as the database)

The transaction logs can be added to the full database backup file if wanted. This does not overwrite the backup file but only append the logs to the backup file.

If the Recovery Model for the database is “full”, then it is necessary to create Transaction Log Backups also. This must be done for the Vault and Gate databases .

To check the Recovery Model do as in the previous section

Create a .bat file and a .sql file and save them on the SQL Server (e.g. C:\Scripts\backuplog.bat and C:\Scripts\backuplog.sql)

The backuplog.bat file should contain the following line:

```
sqlcmd -S(local)\<INSTANCENAME> -E -ibackuplog.sql #(remove \<INSTANCENAME> if no instance name)
```

The backuplog.sql file should contain the following:

```
DECLARE @name VARCHAR(50)       -- database name
DECLARE @path VARCHAR(256)      -- path for backup files
DECLARE @fileName VARCHAR(256)  -- filename for backup
SET @path = 'C:\Backup\ '
SET @name = '<database_name>'

BEGIN
    SET @fileName = @path + @name +'.bak'
    BACKUP LOG @name TO DISK= @fileName
END
```

Create a Scheduled Task for the backuplog.bat file also. It should run once every hour.

Make sure to copy the backup file to another location also.

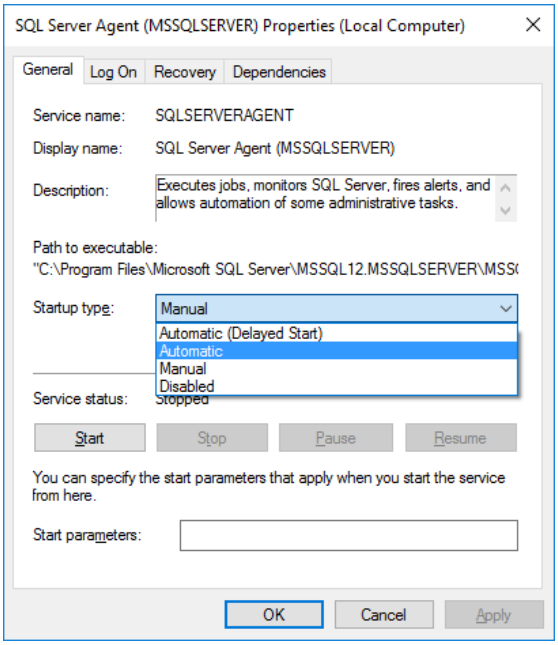
SQL Server Agent

The following will serve as an example. There are several ways to setup a backup strategy.

In the “Steps” part below there is a part of the “Command” written in bold italic that could be used when in a mirroring configuration. It should not be used when not using mirroring.

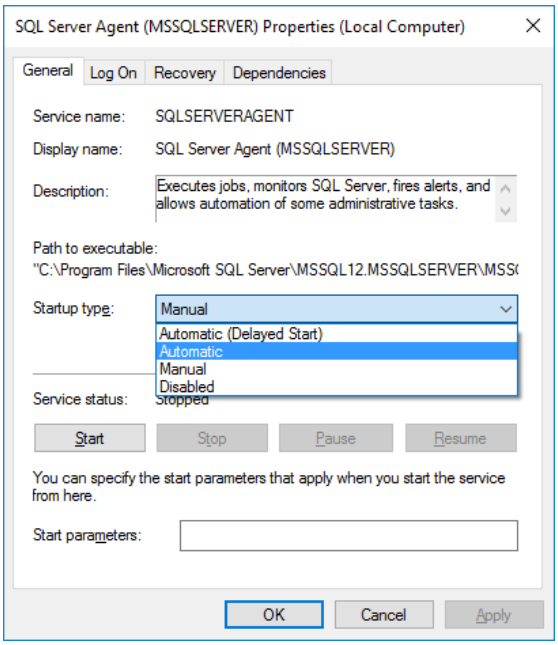
Start the SQL Server Agent Service and configure it to start automatically.

- From the “control panel”, start “administrative tools” and “services”. Double click on the SQL service.
- Set “Startup type” to Automatic.



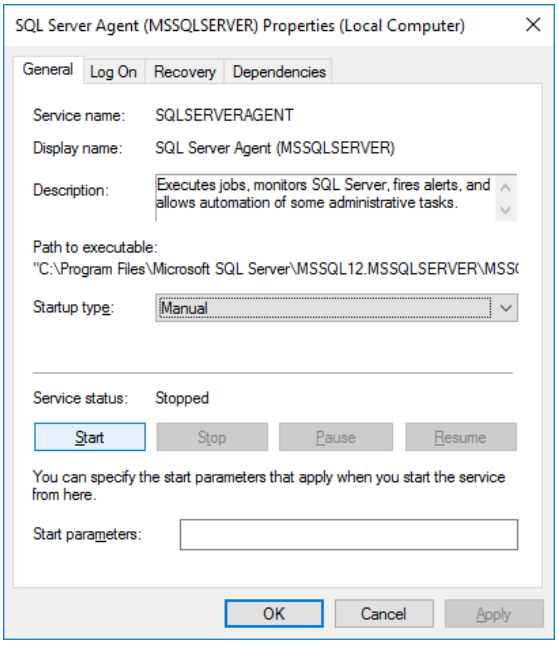
Startup type Automatic

- Click “Apply”.



Apply Properties

- Click “Start” and “OK”.



Start and Exit Properties

If the backup is to be performed on another server, the SQL Server Agent Service will be required to run under an account that has access to the network where the other Server is located. In this case the SQL server Agent service account must be changed from Local System account to a user account. This account can be the same as for SQL server.

Open SQL Server Management Studio.

- Explore Server Objects

Right-click on Backup Devices and select New Backup Device

- Enter a Device name (e.g. <db_device_name>)
- Enter the path to the Database Backup file in File
- Click ok

Explore the SQL Server Agent.

Create Database backup:

- Right-click on Jobs and select New Job
  General
  - Enter a name for the job (e.g. Database Backup <DATABASE_ NAME>)

- Select the user that runs the SQL Server Agent in Owner
- Make sure that the Job is marked as Enabled

Steps

Click New

- General
  - Enter a name for this step (e.g. Database Backup)
  - Select Transact-SQL script (T-SQL) as Type
  - Select the Database <database_name> as Database
  - Type the following in Command:

(The bold italic part is ONLY if mirroring is used.)

```
DECLARE @role VARCHAR(50)
SELECT @role = mirroring_role_desc
    FROM sys.database_mirroring sm
    INNER JOIN sys.databases d ON sm.database_id = d.database_id
        AND d.name =<database_name>
    IF @role ='PRINCIPAL'
    BACKUP DATABASE<database_name> TO <db_device_name> WITH INIT
```

Advanced

- On success action: Go to the next step

Schedules

- Click New
  - Enter a name for this Schedule (e.g. Database Backup)
  - Type should be Recurring
  - Make sure this Schedule is marked as Enabled
  - Under Frequency, select Daily and 1 day(s)
  - Under Daily frequency, select Occurs once at <the_time_for_this_backup>
  - Click Ok

Remember to create a full backup first – before the Transaction Log Job starts.

Right-click on Backup Devices and select New Backup Device

- Enter a Device name (e.g. <db_log_device_name>)
- Enter the path to the Database Backup file in File
- Click ok

Create Transaction Log backup

- Right-click on Jobs and select New Job

  General
  - Enter a name for the job (e.g. Transaction Log Backup)
  - Select the user that runs the SQL Server Agent in Owner
  - Make sure that the Job is marked as Enabled

  Steps

  Click New
  - General
    - Enter a name for this step (e.g. Transaction Log Backup)
    - Select Transact-SQL script (T-SQL) as Type
    - Select the Database <database_name> as Database
    - Type the following in Command:

  (The bold italic part is ONLY if mirroring is used.)

```
DECLARE @role VARCHAR(50)
SELECT @role = mirroring_role_desc
    FROM sys.database_mirroring sm
    INNER JOIN sys.databases d ON sm.database_id = d.database_id
        AND d.name = <database_name>
    IF @role ='PRINCIPAL'
    BACKUP LOG<database_name> TO <db_log_device_name>
```

  Advanced
  - On success action: Go to the next step

  Schedules

  Click New
  - Enter a name for this Schedule (e.g. Transaction Log Backup)
  - Type should be Recurring
  - Make sure this Schedule is marked as Enabled
  - Under Frequency, select Daily and 1 day(s)
  - Under Daily frequency, select Occurs every 60 minute(s)
  - Click Ok

Create copy task

- Create a bat file C:\Scripts\copylog.bat containing the following:

```
SET current_time=%time: =0%
copy <path_to_db_backup_file>\<database_name>_LOG.bak <path_to_destination>\<database_name>_LOG.%current_time:~0,2%.bak
```

- Right-click on Jobs and select New Job

  General
  - Enter a name for the job (e.g. Copy backup file)

- Select the user that runs the SQL Server Agent in Owner
- Make sure that the Job is marked as Enabled

Steps

Click New

General

- Enter a name for this step (e.g. Copy backup file)
- Select Operating system (CmdExec) as Type
- Select SQL Agent Service Account in Run as
- Type the following in Command: C:\Scripts\copylog.bat

Advanced

- On success action: Quit the job reporting success

Schedules

Click New

- Enter a name for this Schedule (e.g. Copy backup file)
- Type should be Recurring
- Make sure this Schedule is marked as Enabled
- Under Frequency, select Daily and 1 day(s)
- Under Daily frequency, select Occurs every <XX> hour(s). This should be matching the transaction log backup interval, so that it is copied every time a transaction log has been made. Due to network traffic this could be too often, so consult with the IS department. Another option is to back up the logs to time stamped files and only copy the latest file. To get an idea, consult the section above for SQL server Express on how to time stamp log files.
- Click Ok

Log Shipping

Essentially, log shipping is the process of automating the backup of database and transaction log files on a production SQL server, and then restoring them onto a standby server. But this is not all. The key feature of log shipping is that it will automatically backup transaction logs throughout the day (for whatever interval you specify) and automatically restore them on the standby server. This in effect keeps the two SQL Servers in synch.

Please see https://docs.microsoft.com/en-gb/sql/database-engine/database-mirroring/database-mirroring-and-log-shipping-sql-server?view=sql-server-ver15 for details on setting up log shipping.

SQL Server Summary: Restore

Restore

Manual

The user logged on to the database server must have “Create database” permissions to restore a database.

To check the permissions of the user or to give the user “Create database” permissions do the following:

Start “SQL Server Management Studio.

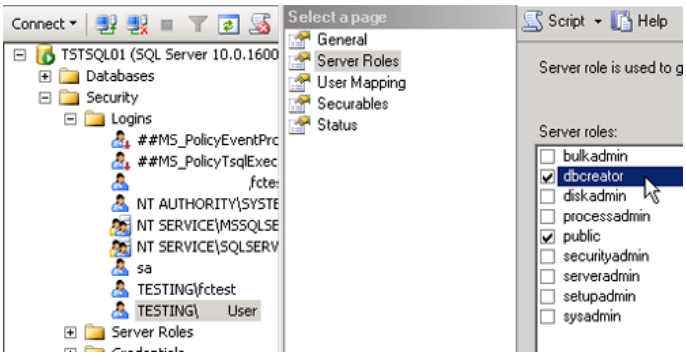
Explore “Security” and “Logins”

Double-click on the user

Select “Server Roles” in the menu on the left

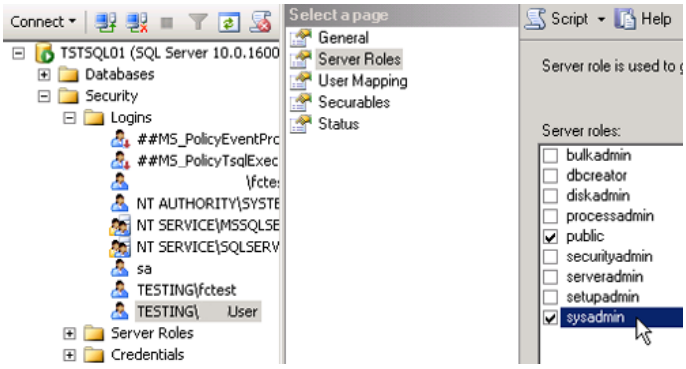
Check the permissions for the user.

It should be “dbcreator”...



dbcreator

or “sysadmin”.



sysadmin

Find the latest full Database Backup and all the Transaction Log Backups that are newer than the latest full database backup.

Open SQL Server Management Studio

Restore Database:

Right-Click on Databases, select Restore Database...

Mark "From Device"

Click on ...

Click Add

Find the full Database Backup file (including the transaction log backups)

Click OK

Click OK

To database: <database_name>

Mark "Restore" for the one just added

And mark all subsequent transaction logs.

Semiautomatic

There is a possibility to define some scripts to automate the restoration when transaction logs are stored in separate files:

Recover from Full Database Backup (no Transaction Logs):

Create a .bat file and a .sql file and save them on the SQL Server (e.g. C:\Scripts\restoredb.bat and C:\Scripts\ restoredb.sql)

The restoredb.bat file should contain the following line:

```
sqlcmd -S(local)\<INSTANCENAME> -E -irestoredb.sql #(remove \<INSTANCENAME> if no instance name)
```

The restoredb.sql file should contain the following:

```
DECLARE @name VARCHAR(50)       -- database name
DECLARE @path VARCHAR(256)      -- path for backup files
DECLARE @fileName VARCHAR(256)  -- filename for backup
SET @path ='C:\Backup\'SET @name ='<database_name>'
BEGIN
    SET @fileName = @path + @name +'.bak'
    RESTORE DATABASE @name FROM DISK= @fileName WITH NORECOVERY

END
```

Recover from Full Database Backup and Transaction Logs:

To be able to use master.sys.xp_cmdshell you must run the following commands:

```
EXEC master.dbo.sp_configure'show advanced options', 1
RECONFIGURE
EXEC master.dbo.sp_configure'xp_cmdshell', 1
RECONFIGURE
```

Create a .bat file and a .sql file and save them on the SQL Server (e.g. C:\Scripts\restore_db_log.bat and C:\Scripts\ restore_db_log.sql)

The **restore_db_log.bat** file should contain the following line:

```
sqlcmd -S(local)\<INSTANCENAME> -E -I restore_db_log.sql(remove \<INSTANCENAME> if no instance name)
```

The **restore_db_log.sql** file should contain the following:

```
USE [master]
GO
SET NOCOUNT ON

-- 1 - Variable declaration
DECLARE @dbName sysname
DECLARE @backupPath NVARCHAR(500)
DECLARE @cmd NVARCHAR(500)
DECLARE @fileList TABLE (backupFile NVARCHAR(255))
DECLARE @lastFullBackup NVARCHAR(500)
DECLARE @lastDiffBackup NVARCHAR(500)
DECLARE @backupFile NVARCHAR(500)
DECLARE @fullBackupPath NVARCHAR(500)

-- 2 - Initialize variables
SET @dbName ='<database_name>'
SET @backupPath ='C:\Backup\ '

-- 3 - get list of files
SET @cmd ='DIR /b '+ @backupPath
INSERT INTO @fileList(backupFile)
EXECmaster.sys.xp_cmdshell@cmd

-- 4 - Find latest full backup
SET @lastFullBackup = @dbName +'.bak'
SET @fullBackupPath = @backupPath + @lastFullBackup
SET @cmd ='RESTORE DATABASE '+ @dbName +' FROM DISK = '''
    + @backupPath + @lastFullBackup + ''' WITH NORECOVERY, REPLACE'
PRINT @cmd

BEGIN
    RESTORE DATABASE @dbName FROMDISK= @fullBackupPath WITHNORECOVERY,REPLACE
END

-- 5 - check for log backups
DECLARE backupFiles CURSOR FOR
    SELECT backupFile
    FROM @fileList
    WHERE backupFile LIKE'%_log.bak'
    AND backupFile LIKE @dbName +'%'

OPEN backupFiles
```

```
43
44  -- Loop through all the files for the database
45  FETCH NEXT FROM backupFiles INTO @backupFile
46  WHILE @@FETCH_STATUS= 0
47  BEGIN
48      SET @cmd ='RESTORE LOG '+ @dbName +' FROM DISK = '''
49          + @backupPath + @backupFile + ''' WITH NORECOVERY'
50      PRINT @cmd
51
52      SET @fullBackupPath = @backupPath + @backupFile
53      RESTORE LOG @dbName FROM DISK = @fullBackupPath WITH NORECOVERY
54
55      FETCH NEXT FROM backupFiles INTO @backupFile
56  END
57
58  CLOSE backupFiles
59  DEALLOCATE backupFiles
60
61  -- 6 - put database in a useable state
62  SET @cmd ='RESTORE DATABASE '+ @dbName +' WITH RECOVERY'
63  PRINT @cmd
64  RESTOREDATABASE @dbName WITHRECOVERY
```

To disable master.sys.xp_cmdshell again you must run the following command:

```
1  EXEC master.dbo.sp_configure'xp_cmdshell', 0
2  RECONFIGURE
```

Database Administors: Vault Database Administration

Overview

This page describes all the necessary steps required to create and configure a database needed to run a Vault.

- Overview
- Inextor Database Installation
  - DAC Package Installation
  - Vault Service Login Creation
- Certificates Import in Database
- Filling Admin Account Details in Database
- Geonames Database Installation
- Upgrade Scripts
- SQL Mirroring

Inextor Database Installation

Create a database for the Vault

Note 1

For SQL Server mirroring (i.e. multiple database (high availability) configuration), the database should only be created on the SQL server instance that should start with being principal.

Note 2 - Collations

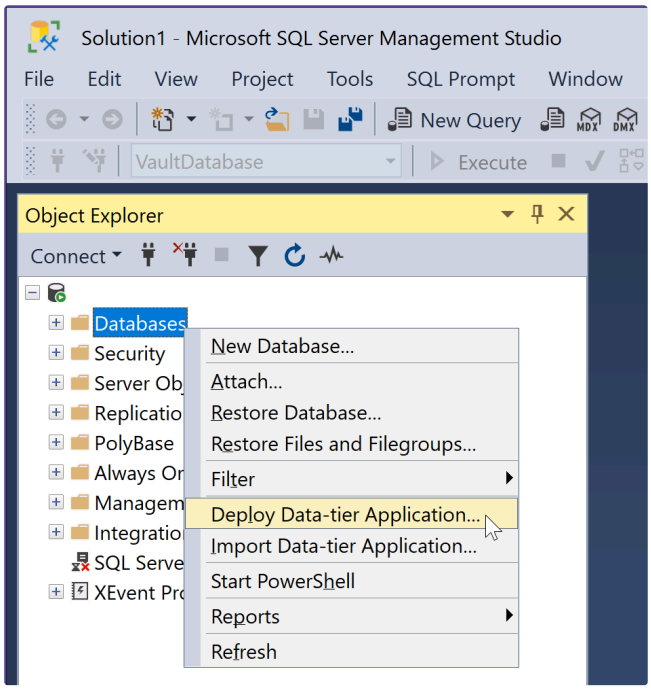
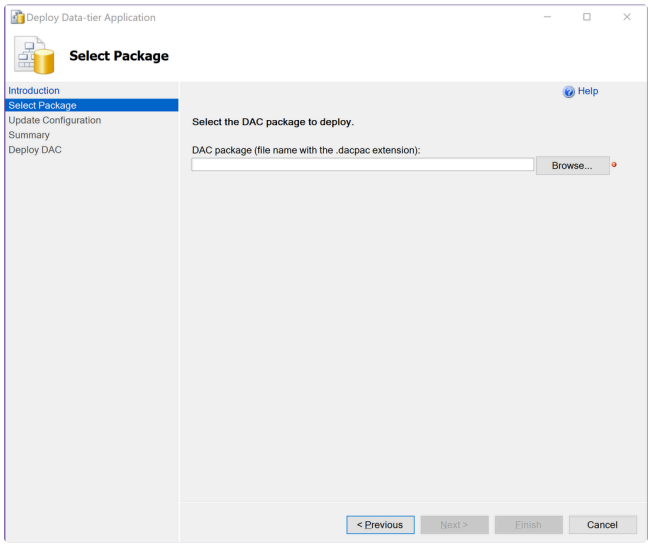
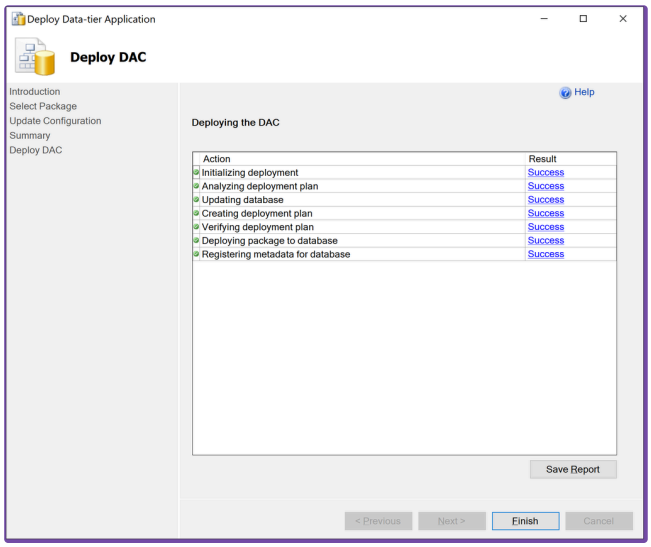
The Vault database dacpac uses collation **SQL_Latin1_General_CP1_CI_AS**. This is the only collation supported and should not be changed.

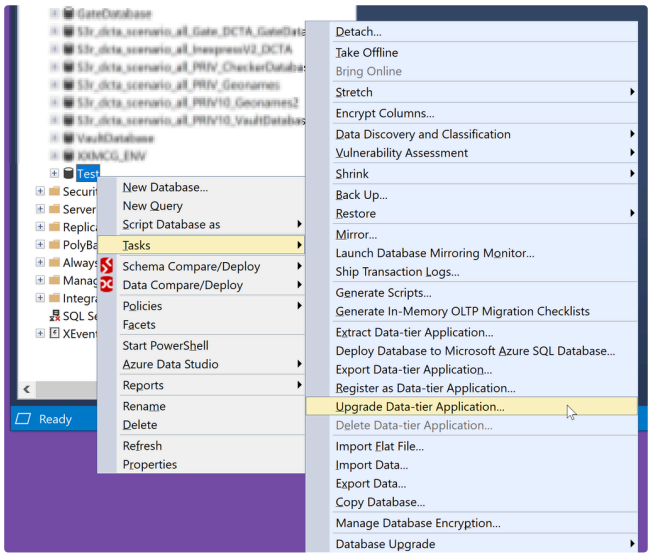
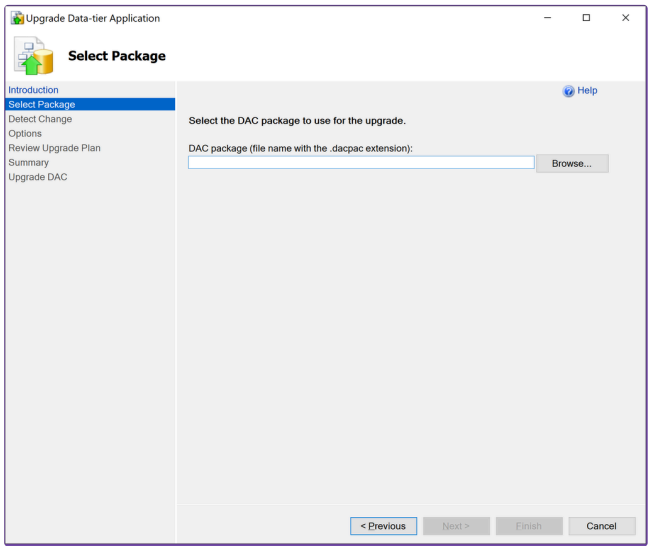
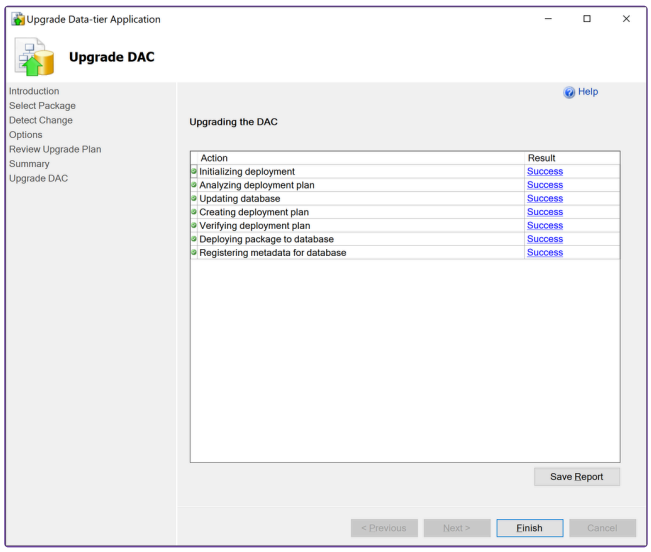
The SQL Server on which the Vault database is attached should have collation **SQL_Latin1_General_CP1_CI_AS**, or **Latin1_General_CI_AS**. Other collations have not been sufficiently tested and are not supported.

DAC Package Installation

Follow the steps below to complete this process

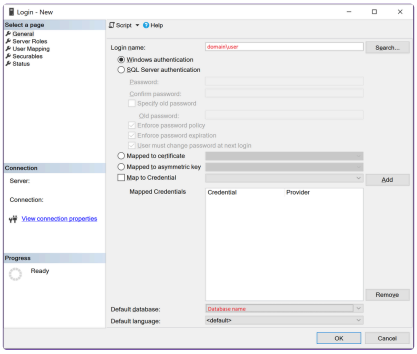
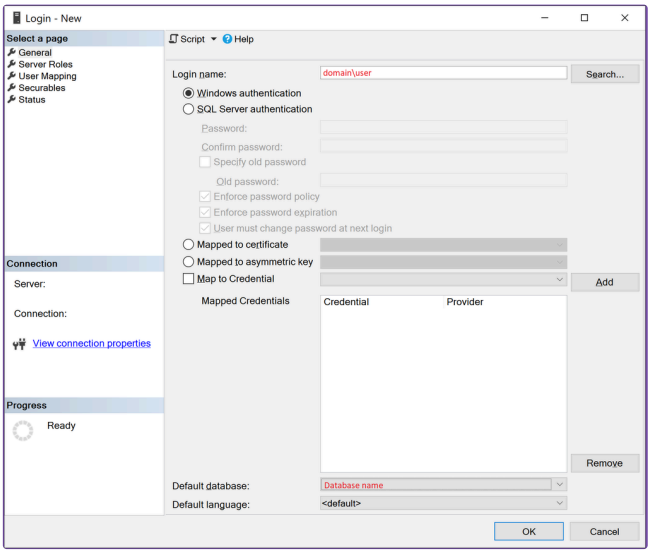
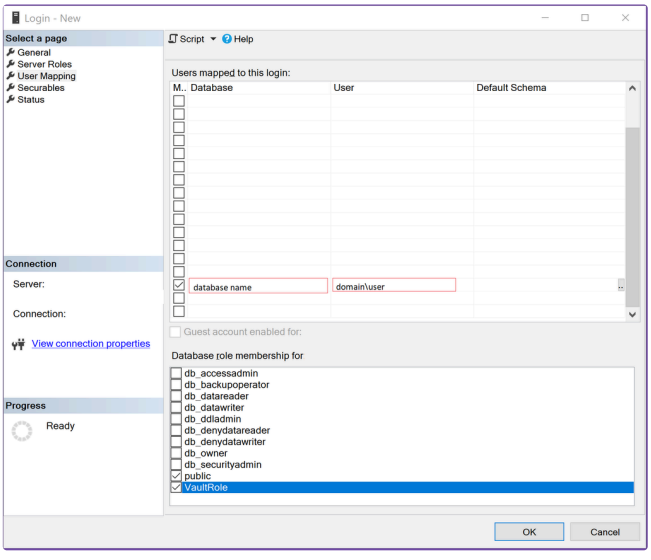
	Steps	Screenshots

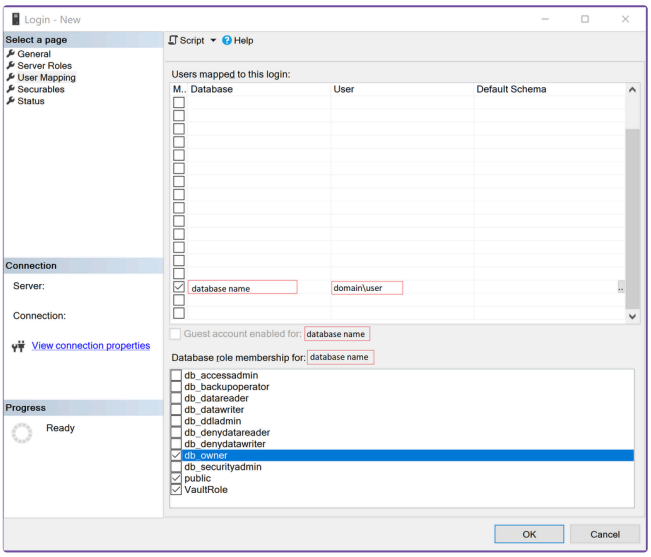
1	• Open the SQL Management Studio and connect to an SQL Instance • Create a Data-tier Application on the SQL Server	
2	• Select the DAC package of the Inextor database from the installation media • Enter a database name and click on **Next**	
3	• Deploy the DAC package and make sure all results are successful	

4	• Upgrade the newly deployed DAC package with the industry-specific DAC package (Tobacco, Pharma, etc.) **Note**: The Gate Software is often accompanied with the DAC package stored stored in a ZIP folder. If this file is not available for you, please ask the Support Desk.	
5	• Select the DAC package located in the database Vault folder	
6	• Click next until the upgrade is successful	

Vault Service Login Creation

Follow the steps below to complete this process

	Steps	Screenshots
1	• Create a Login on the Database Server for the service account that runs the Vault Service	
2	• Enter the service account that runs the Vault Service and select the Inextor database as the default database.	
3	If you want to limit the access rights for the Vault NT Service, the service account which accesses the database should be assigned the role VaultserviceRole. • Select User Mapping • Select the Vault database • Make sure that db_owner is not checked • Check the VaultserviceRole and press OK	

4	If not deployed with the limited access then: • Give DBO access to this database to the account under which the service will run.	

Certificates Import in Database

The next step of the database setup is to import the following certificates into the database:

- Root certificate
- TP certificate
- Vault certificate
- ADMIN certificate.

This is done using the Certificate Manager tool as described in Import Certificates into Vault Database.

First, edit the configuration file (CertificateManager.exe.config) and enter the correct password (if there is a password for the certificates)

> Note: If using data protection for the certificates with private keys, then you also need the following (see Create Keys for Data Protection):
>
> - Specific key pair ("Specific key.txt")
> - The global public key ("Public key for the Global key.txt")

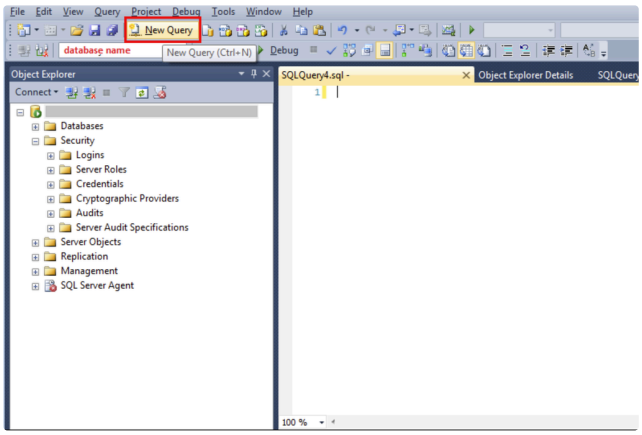
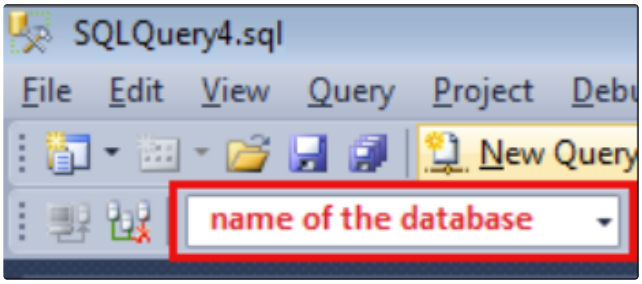
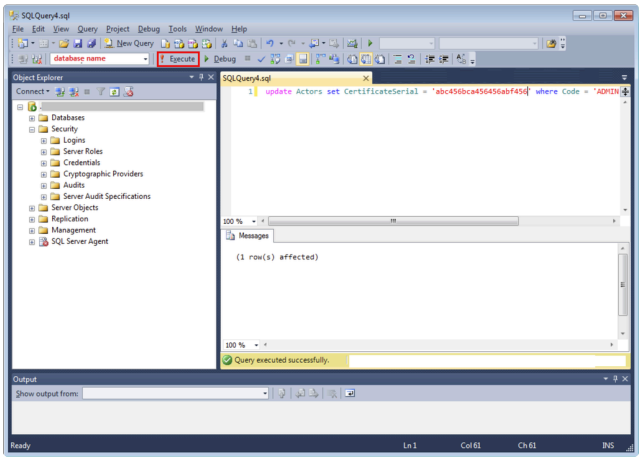
Filling Admin Account Details in Database

In order to be able to authenticate as an administrator, the serial number of the ADMIN certificate must be stored in the CertificateSerial field of the ADMIN Actor in the database.

Therefore it is important to retrieve the serial number of the certificate from the Certificates table and update the Actors table accordingly.

This can be done with the following procedure.

1	• In SQL Server Management Studio, Select New Query. This will open a new SQL query window on the right.	

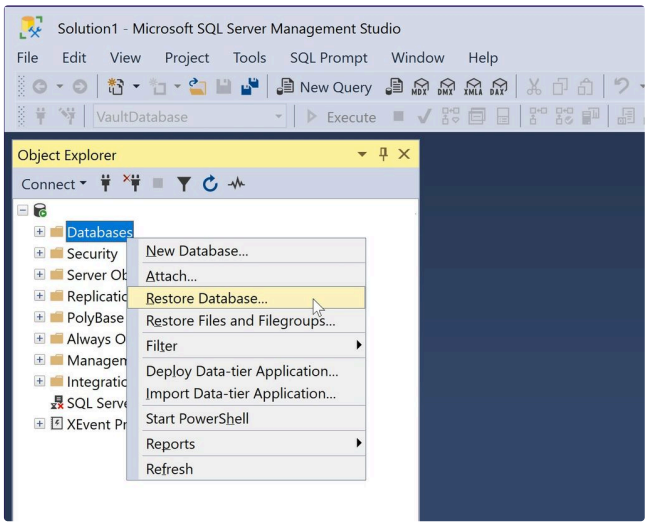
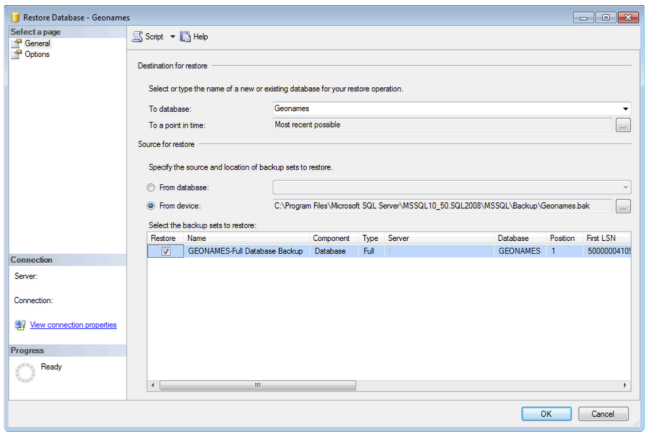
		
2	Select the Vault database	
3	Execute the query shown below (step 4) on the Vault database.	
4	SQL query to execute `update Actors set CertificateSerial = (select SerialNumber from Certificates where IssuedTo like 'CN=ADMIN%') where Code = 'ADMIN'`	

Geonames Database Installation

The Geonames database provides geographical data to associate locations to Inextor objects (entities, factories, code verifications).

This database needs to be installed along with the Inextor Vault database.

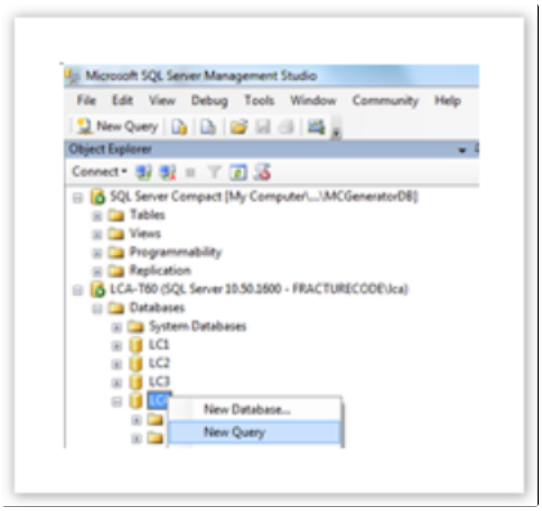
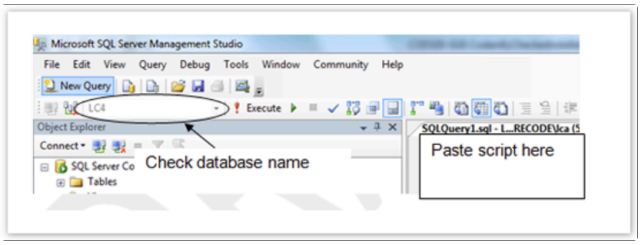
Steps	Screenshots

1	• Open SQL Server Management Studio • Connect to the database server planned to host the Inextor database • Right-click on the Databases folder and select "Restore Database".	
2	• Parameterize a full database restore from the device and select the backup file **Geonames.bak** from the installation media. • Confirm database backup by clicking OK. • Make sure that the service account that runs the Vault Service has read access (db_datareader role) on the Geonames Database.	

Upgrade Scripts

Note: This section is for upgrade only and not for new installation or disaster recovery.

To perform an upgrade, you may be given SQL scripts that need to be run. To run the scripts, follow the procedure below:

1	• Open SQL Management Studio and connect to your SQL instance • Select the database, right-click and select New Query	
2	• Open the script file (*.sql) in Notepad and copy all the content. • In SQL Management studio, paste the content to the right (see snapshot below) • Check that the correct database is selected • Click the Execute button • Check that the script ran without errors.	

SQL Mirroring

Note: This is only required if using SQL server mirroring, in the case of a multi-server installation.

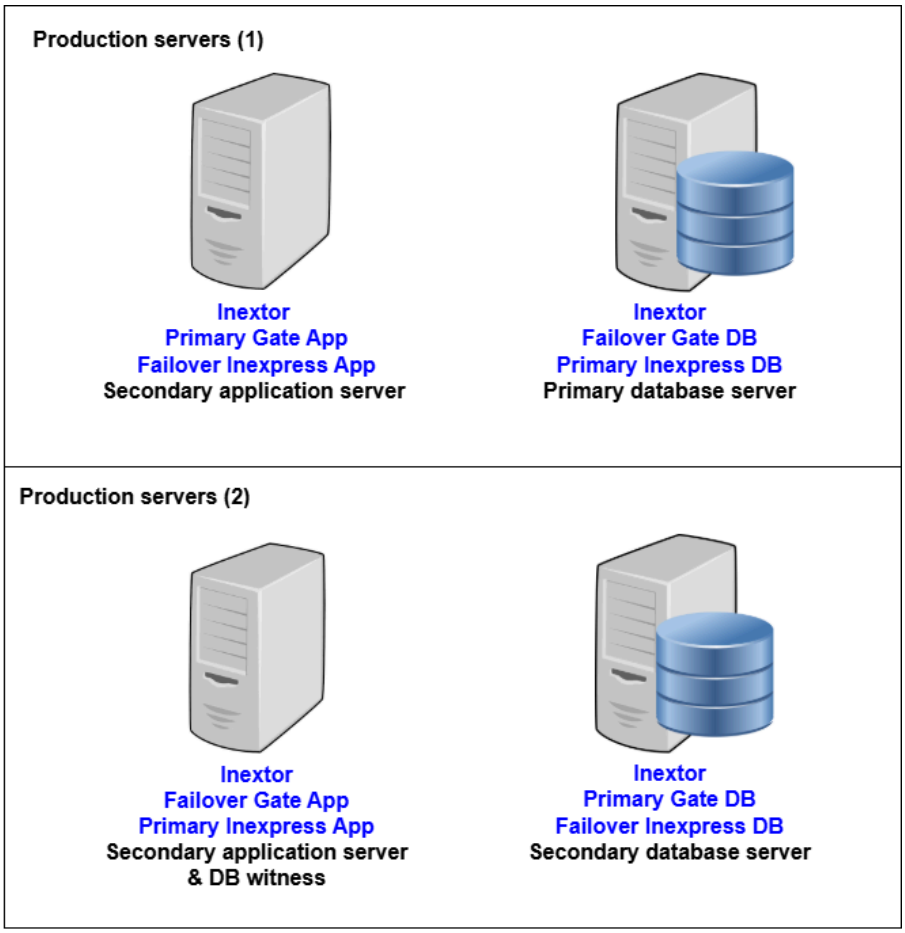
Setup the mirroring as described in Database mirroring with one of the following options:

- Without Witness using Windows Authentication
- Without Witness using Certificates
- With Witness using Windows Authentication
- With Witness using Certificates

Note: The security on the mirrored SQL Server is automatically configured to give DBO access to this database to the account under which the Vault Service will run.

Database administrators: Contingency plans

High availability can be attained by implementing redundancy for both Application Servers and Database Servers. To maximize the efficiency of machines and Microsoft licenses, the following configuration can be adopted:



Database Contigency Plans: Always On

Recommendation
We recommend to use **Always On Basic Availability Groups**, which provides a similar level of feature support as the deprecated **Database Mirroring** feature.

Only one license is required, as the inactive replica does not need one.

There is no witness role with Always On, as the quorum is determined among all the replicas. However since only 2 replicas are supported with **Basic Availability Groups**, it is not possible to have the equivalent of the witness role.

It is recommended to use synchronous mode (High Availability) rather than asynchronous mode (High Performance), unless the 2 replicas are located far away from each other.

There are three different things that are called **Always On** and this can cause some confusion.

The feature matrix below describes the significant differences that exist between them (SQL2016/SQL2017) :

	Standard	Enterprise	Comments
Basic availability groups	X (2 nodes)		**Can be used as replacement for database mirroring.**
Always On failover cluster instances	X (2 nodes)	X	This feature provides local high availability through redundancy at the SQL server-instance; However it does **not** provide redundancy at the storage level, so it is not as protective as database mirroring. This solution should be used in combination with a SAN (Storage Area Network). **Cannot be used as a replacement for database mirroring.**
Always On availability groups		X	**Can be used as replacement for database mirroring.**

High Availability Solutions: https://docs.microsoft.com/en-us/sql/sql-server/failover-clusters/high-availability-solutions-sql-server?view=sql-server-2017

Database mirroring

Microsoft has not determined when they will deprecate database mirroring: it will still exist in SQL Server 2017 and next version.

In conclusion, it can still be used for many years for companies who want to purchase only SQL Standard Edition, which does not support Always On availability groups.

https://docs.microsoft.com/en-us/sql/database-engine/database-mirroring/database-mirroring-operating-modes?view=sql-server-2017

Comparison of Failover Cluster Instances (FCI) and Availability Groups

This section is copied from https://docs.microsoft.com/en-us/sql/database-engine/availability-groups/windows/failover-clustering-and-always-on-availability-groups-sql-server?view=sql-server-2017

Regardless of the number of nodes in the FCI, an entire FCI hosts a single replica within an availability group. The following table describes the distinctions in concepts between nodes in an FCI and replicas within an availability group.

	Nodes within an FCI	Replicas within an availability group
Uses WSFC cluster	Yes	Yes
Protection level	Instance	Database
Storage type	Shared	Non-shared While the replicas in an availability group do not share storage, a replica that is hosted by an FCI uses a shared storage solution as required by that FCI. The storage solution is shared only by nodes within the FCI and not between replicas of the availability group.
Storage solutions	Direct attached, SAN, mount points, SMB	Depends on node type
Readable secondaries	No*	Yes
Applicable failover policy settings	WSFC quorum FCI-specific	WSFC quorum Availability group settings

	Nodes within an FCI	Replicas within an availability group
	Availability group settings**	
Failed-over resources	Server, instance, and database	Database only

*Whereas synchronous secondary replicas in an availability group are always running on their respective SQL Server instances, secondary nodes in an FCI actually have not started their respective SQL Server instances and are therefore not readable. In an FCI, a secondary node starts its SQL Server instance only when the resource group ownership is transferred to it during an FCI failover. However, on the active FCI node, when an FCI-hosted database belongs to an availability group, if the local availability replica is running as a readable secondary replica, the database is readable.

**Failover policy settings for the availability group apply to all replicas, whether it is hosted in a standalone instance or an FCI instance.

Database Contigency Plans: Database mirroring

In order to ensure high availability of the database for Gate or Vault, database mirroring can be set up.

The set-up varies for each version of SQL Server. The following table gives the appropriate links (at the time of this writing):

Topic	SQL Server 2019	SQL Server 2017	SQL Server 2016
Introduction to mirroring	https://docs.microsoft.com/en-gb/sql/database-engine/database-mirroring/database-mirroring-sql-server?view=sql-server-ver15	https://docs.microsoft.com/en-gb/sql/database-engine/database-mirroring/database-mirroring-sql-server?view=sql-server-2017	https://docs.microsoft.com/en-gb/sql/database-engine/database-mirroring/database-mirroring-sql-server?view=sql-server-2016
Setting up mirroring	https://docs.microsoft.com/en-gb/sql/database-engine/database-mirroring/setting-up-database-mirroring-sql-server?view=sql-server-ver15	https://docs.microsoft.com/en-gb/sql/database-engine/database-mirroring/setting-up-database-mirroring-sql-server?view=sql-server-2017	https://docs.microsoft.com/en-gb/sql/database-engine/database-mirroring/setting-up-database-mirroring-sql-server?view=sql-server-2016
Add a witness	https://docs.microsoft.com/en-gb/sql/database-engine/database-mirroring/add-or-replace-a-database-mirroring-witness-sql-server-management-studio?view=sql-server-ver15	https://docs.microsoft.com/en-gb/sql/database-engine/database-mirroring/add-or-replace-a-database-mirroring-witness-sql-server-management-studio?view=sql-server-2017	https://docs.microsoft.com/en-gb/sql/database-engine/database-mirroring/add-or-replace-a-database-mirroring-witness-sql-server-management-studio?view=sql-server-2016
Manual fail-over	https://docs.microsoft.com/en-gb/sql/database-engine/database-mirroring/manually-fail-over-a-database-mirroring-session-sql-server-management-studio?view=sql-server-ver15	https://docs.microsoft.com/en-gb/sql/database-engine/database-mirroring/manually-fail-over-a-database-mirroring-session-sql-server-management-studio?view=sql-server-2017	https://docs.microsoft.com/en-gb/sql/database-engine/database-mirroring/manually-fail-over-a-database-mirroring-session-sql-server-management-studio?view=sql-server-2016
Tearing down mirroring	https://docs.microsoft.com/en-gb/sql/database-engine/database-mirroring/remove-database-mirroring-sql-server?view=sql-server-ver15	https://docs.microsoft.com/en-gb/sql/database-engine/database-mirroring/remove-database-mirroring-sql-server?view=sql-server-2017	https://docs.microsoft.com/en-gb/sql/database-engine/database-mirroring/remove-database-mirroring-sql-server?view=sql-server-2016

Database administrators: Vault proxy database setup

Vault proxy database installation

Define a database name for the Vault Proxy database

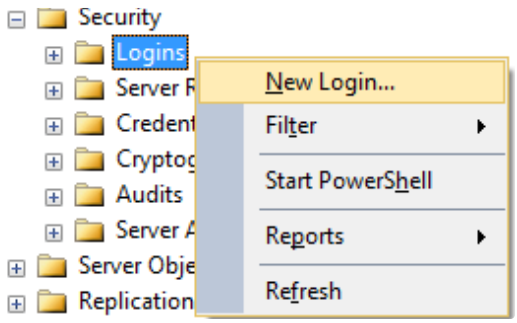
> When using SQL Server 2008 R2, it is possible that the SQL Server client (SSMS) doesn't provide the feature to deploy data-tier applications (dacpac) required for this installation. In such a case, it is necessary to install the client tools from the installation folder of your SQL Server 2008 R2 (see installation instructions).

DAC package installation

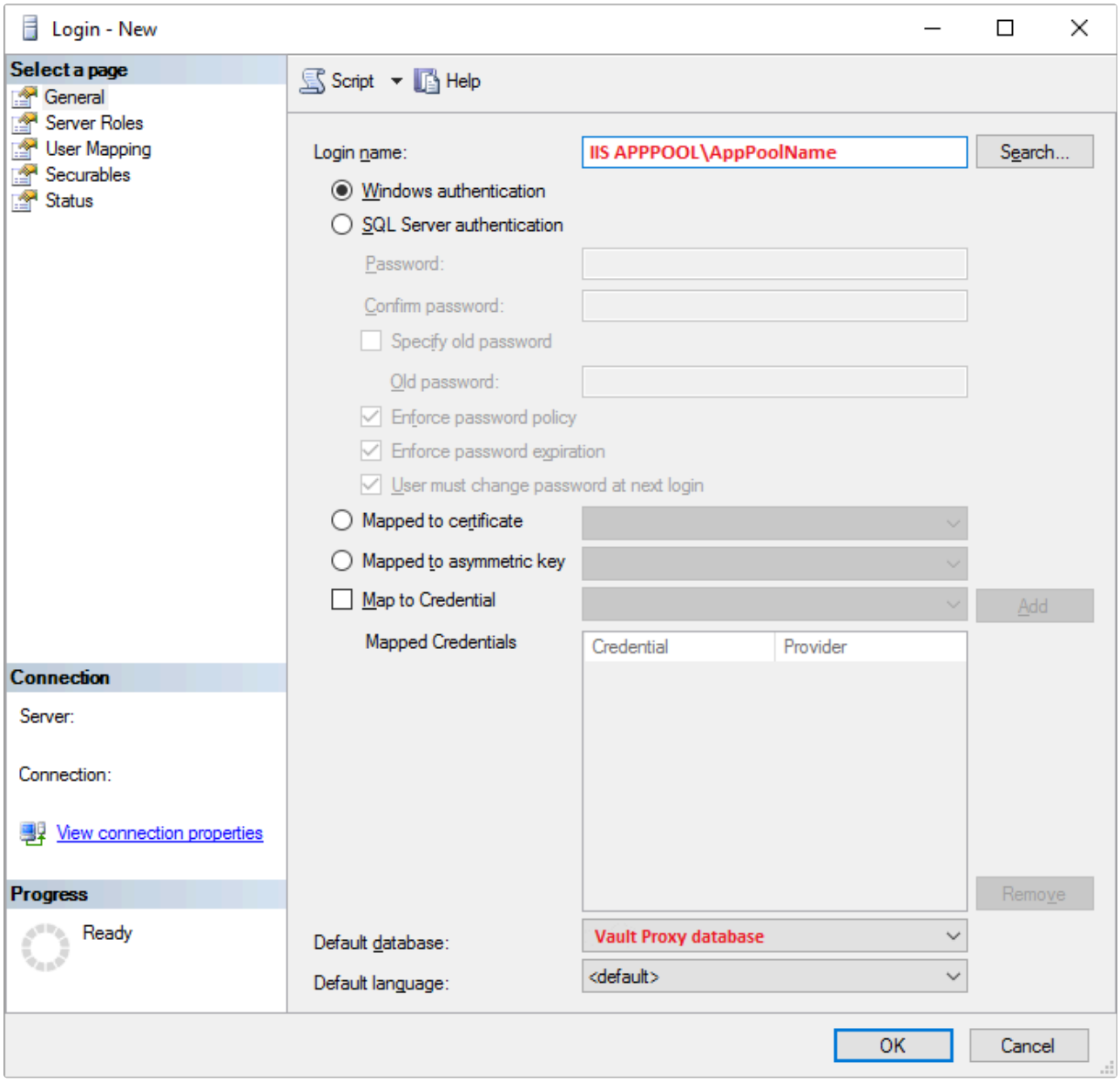
Refer to Vault DAC package Installation to understand how to deploy VaultProxyDatabase.dacpac file onto Vault Proxy database.

Vault service login creation

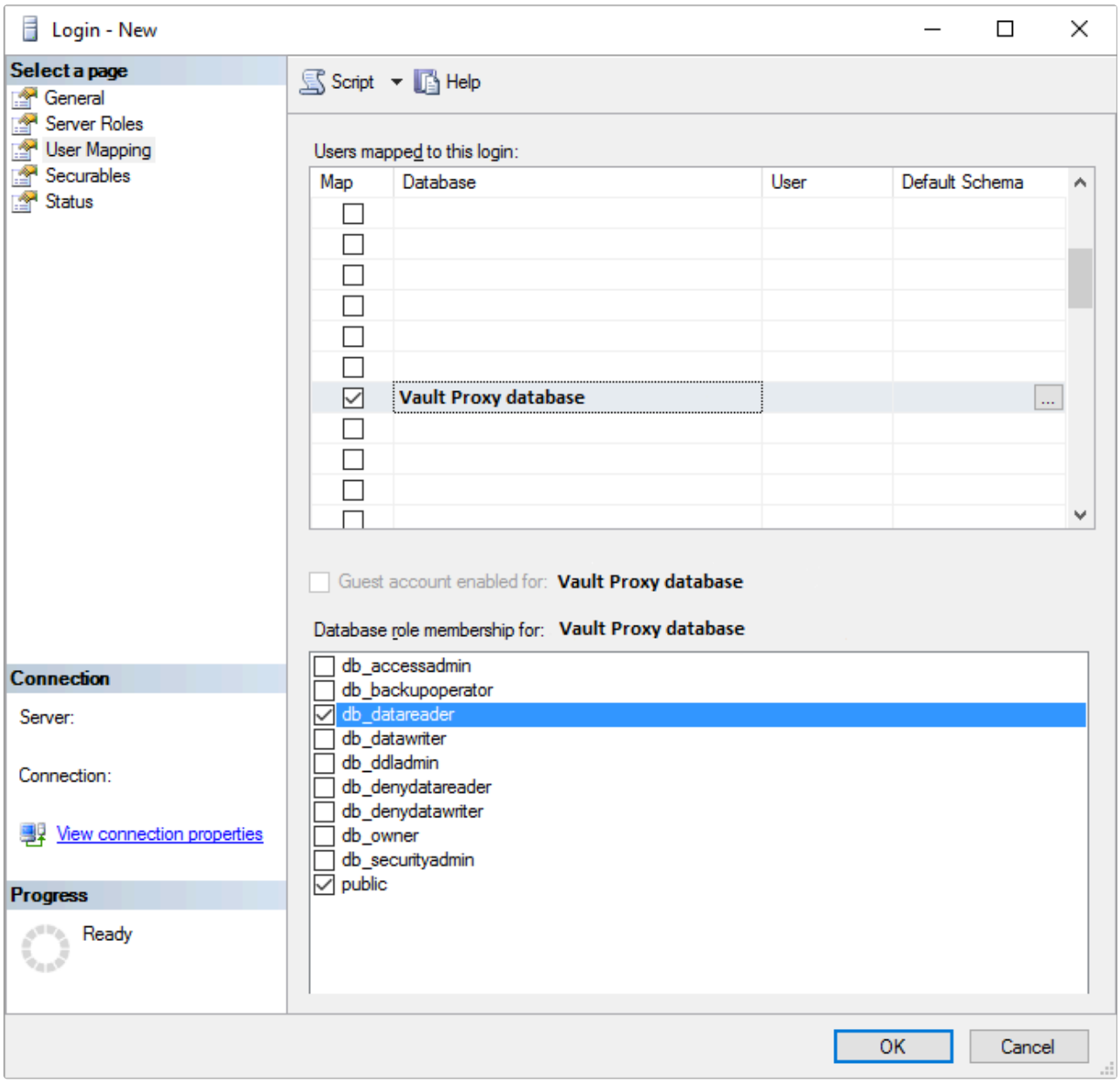
- Create a Login on the Database Server for the AppPool Identity that runs the InextoClientWS.



- Enter the Identity of AppPool which hosts the InextoClientWS and select the Vault Proxy database as default database



- Select User Mapping
- Select the Vault Proxy database
- Make sure that only db_datareader and public roles are checked



Import certificates to database

The next step of the database setup is to import the following certificates in the database:

- Proxy Root Certificate
- Vault Proxy Certificate

This is done using the Certificate Manager tool as described in Import Certificates into Vault Database.

First edit the configuration file (CertificateManager.exe.config) and enter the correct password (if there is a password for the certificates).

Note: If using data protection for the certificates with private keys, then you also need the following (see Create Keys for Data Protection):

- Specific key pair ("Specific key.txt")
- Global public key ("Public key for the Global key.txt")

Database administrators: Advanced maintenance and optimization

Snapshot isolation and row versioning

To enhance database performance and minimize contention—specifically to prevent queries from becoming deadlock victims—it is advisable to configure the database for Snapshot Isolation.

```
USE [master]
ALTER DATABASE GateDB SET SINGLE_USER
ALTER DATABASE GateDB SET ALLOW_SNAPSHOT_ISOLATION ON
ALTER DATABASE GateDB SET READ_COMMITTED_SNAPSHOT ON
ALTER DATABASE GateDB SET MULTI_USER
GO
```

Data Fragmentation

Over time, SQL databases can become fragmented, which directly affects application performance. To prevent this issue, it is essential to establish a proper maintenance plan. Below, we provide a script designed to defragment the databases. This stored procedure can be executed on both the Vault and the Gate.

The script **'Defragmentation.sql'** should be requested to your local IS support department, executed on the relevant database, then scheduled to **run every week** on:

- the Gate
- the Vault database
- potentially other external databases that are related to INEXTOR (like the database used for product integration).

The following parameters have to be used:

Parameter name	Type of value	Recommended value	Description
minFragmentation	Decimal	5.0	In percentage, will not defrag if fragmentation less than specified
rebuildThreshold	Decimal	30.0	In percentage, greater than the value will result in rebuild instead of reorganisation
executeSQL	True/False	1	1 = execute; 0 = print command only
database	Text	<database name>	Option to specify a database name; null will return all
tableName	Text	Null	Option to specify a table name; null will return all
onlineRebuild	True/False	1	1 = online rebuild; 0 = offline rebuild

maxDopRestriction	Integer	Null	Option to restrict the number of processors for the operation
printCommands	True/False	0	1 = print commands; 0 = do not print commands
printFragmentation	True/False	0	1 = print fragmentation prior to defrag; 0 = do not print
defragDelay	Time	00:00:05	Time to wait between defrag commands
debugMode	True/False	1	Display some useful comments to help determine if/where issues occur

And here is the command to be scheduled in SQL Agent:

```
EXECdbo.dba_indexDefrag_sp
    @minFragmentation = 5.0,
    @rebuildThreshold = 30.0,
    @executeSQL = 1,
    @database = N'Gate43_PRD',  -- replace with the
proper db name
    @tableName = Null,
    @onlineRebuild = 1,
    @maxDopRestriction = Null,
    @printCommands = 0,
    @printFragmentation = 0,
    @defragDelay = '00:00:05',
    @debugMode = 1
```

Database administrators: Database

This page outlines the procedure to follow for setting up the Gate database..

Note 1

None of the operations related to the database are automated in the Gate installer; these tasks must be completed before the service installation.

Note 2

When using SQL Server 2008 R2, it is possible that the SQL Server client (SSMS) doesn't provide the feature to deploy data-tier applications (dacpac) required for this installation. In such a case, it is necessary to install the client tools from the installation folder of your SQL Server 2008 R2 (see installation instructions).

Note 3

When using SQL Server 2016 (SP2) (KB4052908), we have encountered that SQL Server client (SSMS) doesn't deploy the (dacpac) for this installation, as that it obtained a session hangs (Known issue). In such a case, it is necessary to install a newer version of the SQL Build. (see MS SQL Builds).

Note 4 - Collations

The Gate database dacpac uses collation **SQL_Latin1_General_CP1_CI_AS**. This is the only collation supported and should not be changed.

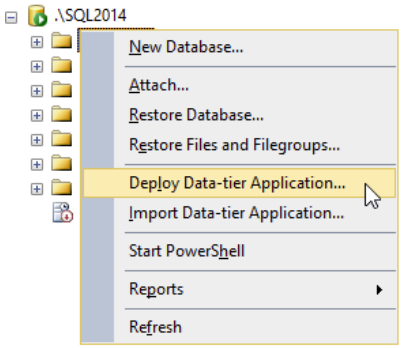
The SQL Server on which the Gate database is attached should have collation **SQL_Latin1_General_CP1_CI_AS**, or **Latin1_General_CI_AS**. Other collations have not been sufficiently tested and are not supported.

The information is organized as follows:

1. DAC Package Installation (in case of a new clean installation or a side-by-side installation)
2. Upgrade Gate Database (in case of an upgrade)
3. Complete the SQL Configuration.

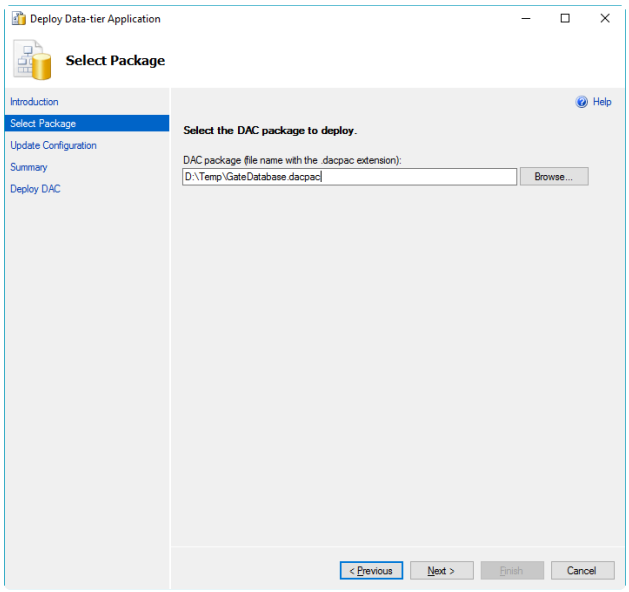
DAC package Installation

- Open SQL Management Studio and connect to your SQL instance
- Create a Data-tier Application on the SQL server.



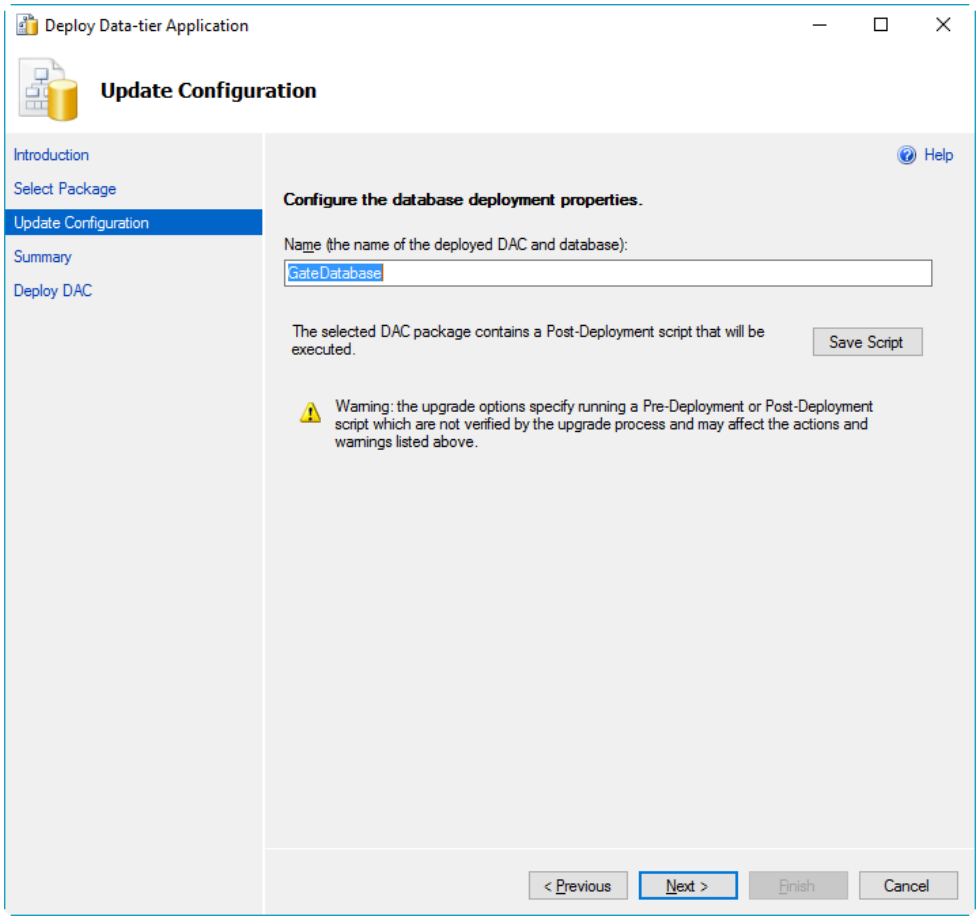
Deploy Data-tier Application

- Select the DAC package of the Inextor database from the installation media



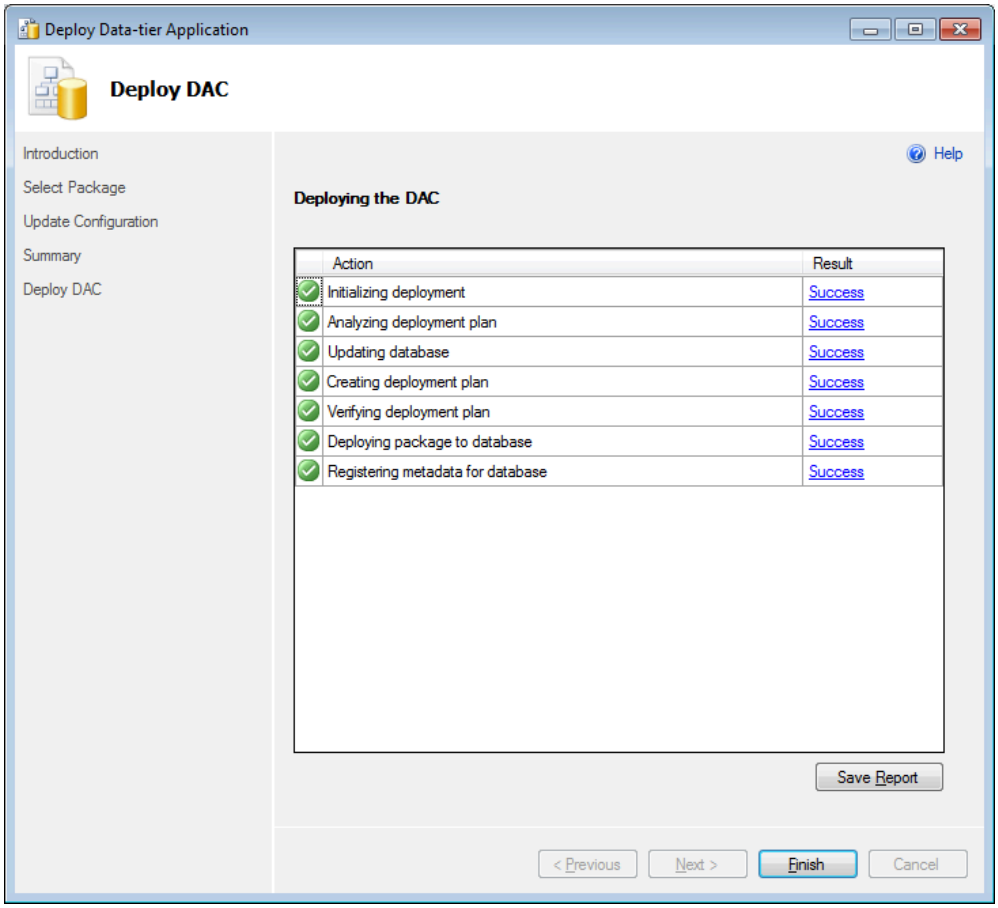
Select Vault's DAC package

- Enter database name



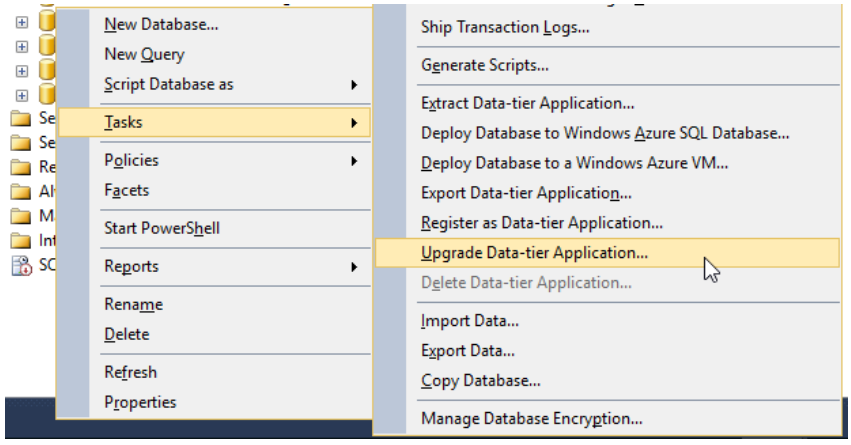
Enter a Database name

- Deploy the DAC package



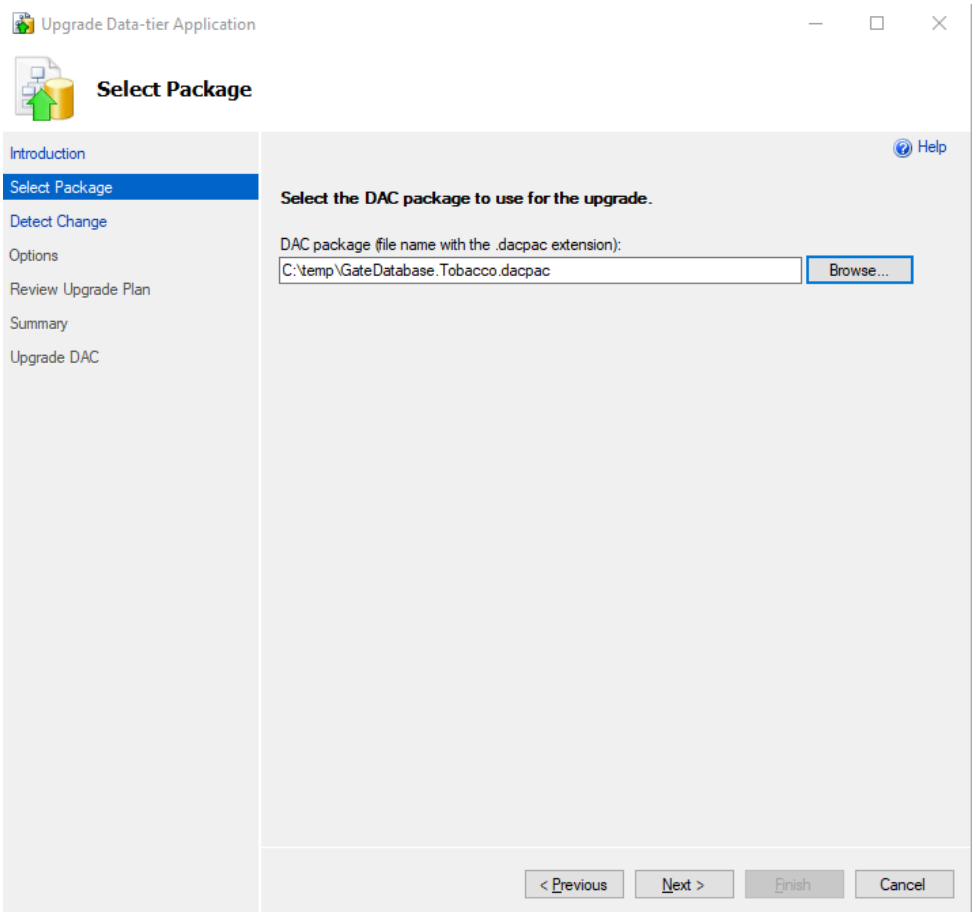
Deploy DAC package

- Upgrade the Gate database



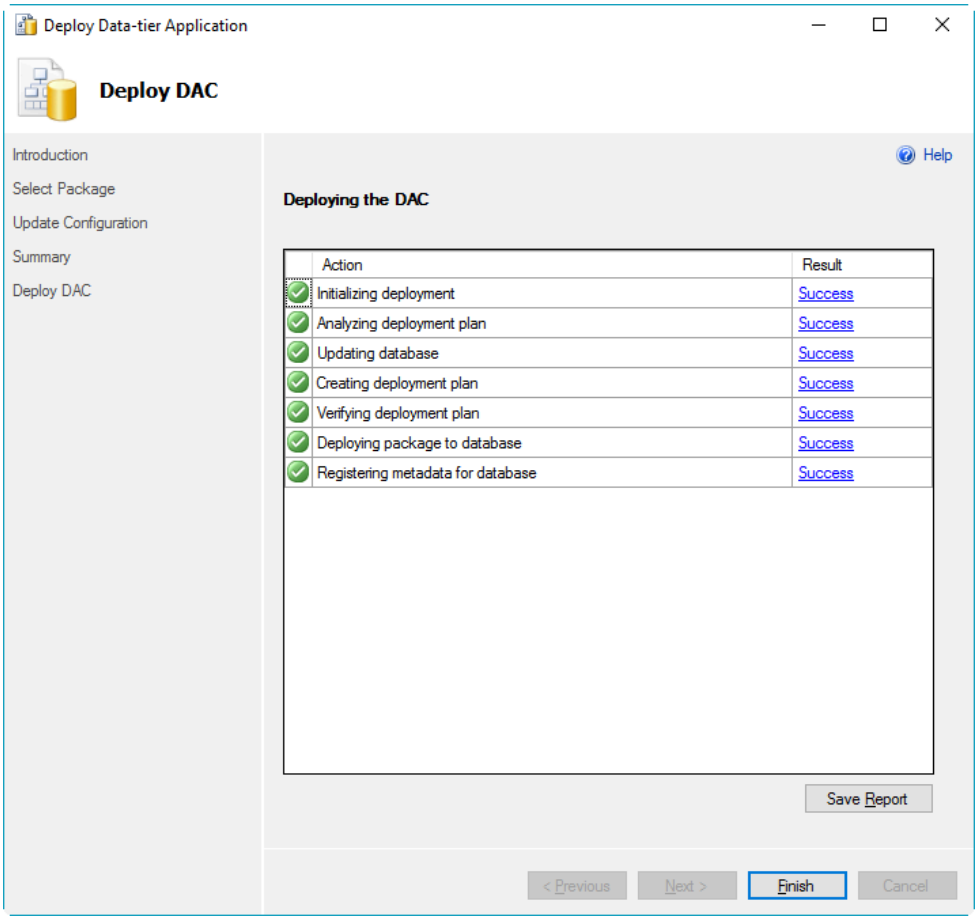
Gate database upgrade

- Select the upgrade dacpac



Select upgrade dacpac

- Click Next until the upgrade is done



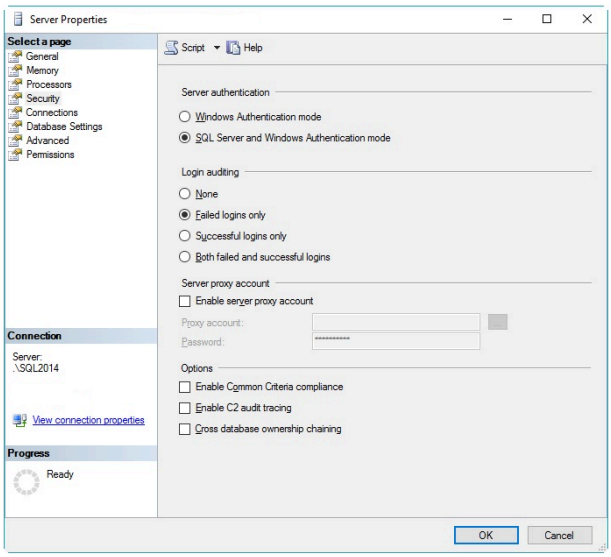
Gate database upgrade successful

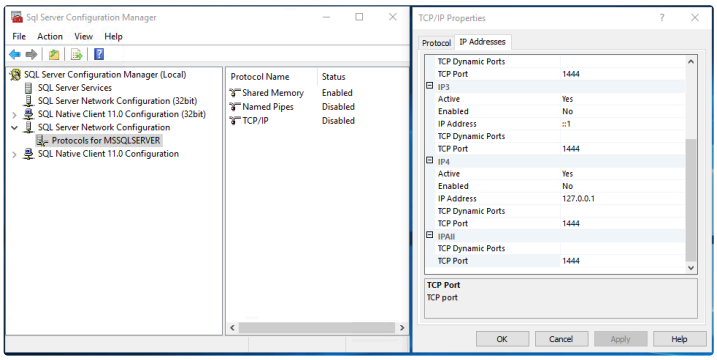
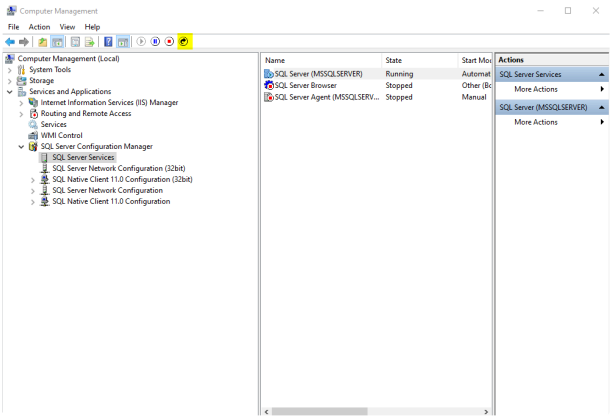
Upgrade Gate Database

The database updates automatically upon the service's startup.

Complete the SQL Configuration

1. In SQL Server Management Studio, right-click on the server name and click Properties then Security. Verify that SQL server is configured in the mixed authentications mode.



2.	Open the SQL Server Configuration Manager and change TCP port number as shown in the picture aside (the default 1433 is known to be vulnerable to network attack). Also, set IP1 and IP2 as Active then Press OK.	
3.	Restart the **SQL Server** Service	

General installation guide: Integration developers

- Integration developers: Inextor Gate
- Integration Developers: Web Services List
- Integration Developers: Vault

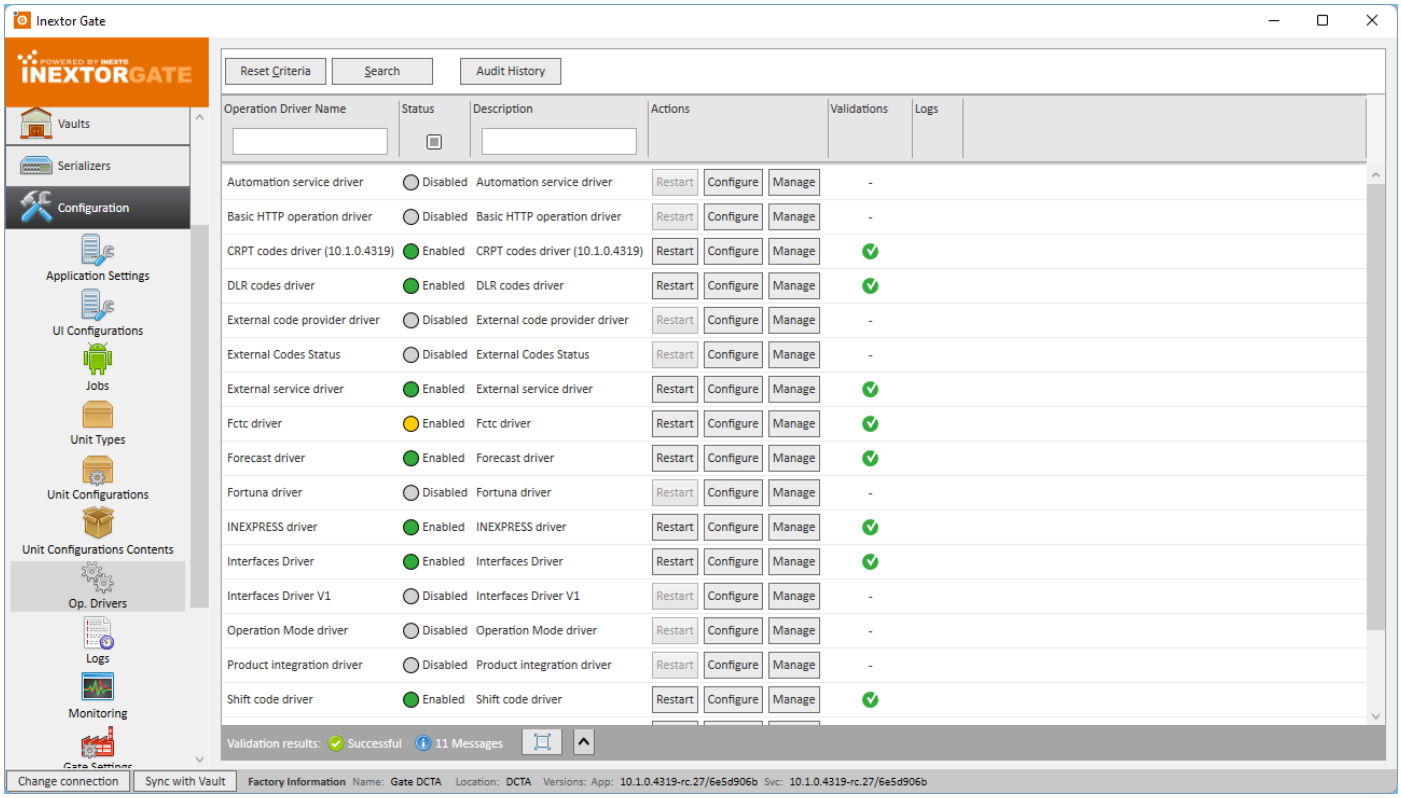
Integration developers: Inextor Gate

- Gate Integration Developers: Operation Drivers
- Gate Integration Developers: Data sources and flows
- Gate Integration Developers: Product integration for the BasicHttpWcf operation driver
- How to configure an operation driver for https
- Dynamic Attributes Editor
- Creating or Editing Reports
- Decompose codes with Ait Service (Gate)
- Product Catalog Integration Views

Gate Integration Developers: Operation Drivers

This chapter explain how to configure the Operation Drivers.

Configure Gate on First Run - Configure Operation Drivers



Gate Configuration - Op. Drivers - Operation Drivers

This screen manages the operation drivers that are custom components. These drivers expose either web services or tasks that run in the background.

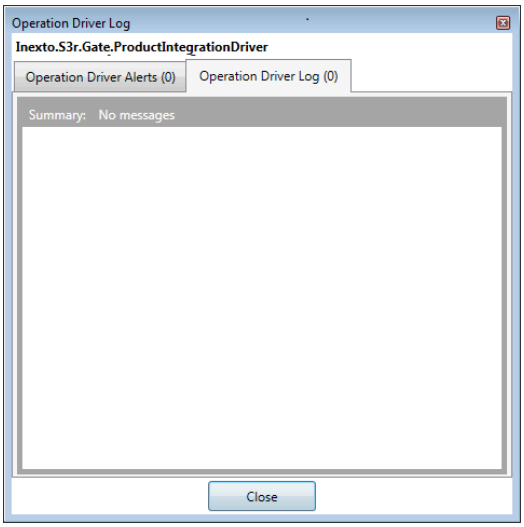
On this screen you have a summary status for the different operation drivers. For each operation driver you can:

- Globally enable or disable the operation driver.
- Enable or disable part of it.

> Disabling and enabling an operation driver will restart it (can be useful to troubleshoot).

By clicking on **Configure** button, you can configure a specific operation driver. The configuration screen of an operation driver is specific to this operation driver.

By clicking on **Log button, you can view the last logs related to this operation driver.**



Gate Configuration - Op. Drivers - Operation Drivers - View Log

Supported Operation Drivers

The list of existing operation drivers depends on the industry.

Gate Integration Developers: Data sources and flows

Manufactures can choose 3 approaches to automatically import their product catalogs into the Gate:

- Develop some software client to push the product data using the Gate web-service **ExternalServiceDriver**.
- Develop some SQL views to extract the product data from an existing database.
- Use the Vault to import product from Master Data Services (MDS).

Using Web services

Manufacturers who want to use a web-service are encouraged to use the new ExternalServiceDriver.ProductInterface.

Using SQL Views

The SQL integration views can be redefined to retrieve the data from a Corporate database.

Using MDS

Products are stored in MDS. The Vault has an operational driver to get the product from MDS and push them to the Gate. See Vault Op. Drivers: INX Driver

A service exists to extract product from a customer's SQL database and populate MDS. See MDS Extractor

Data flow

See the following diagram and table to understand the data flow accomplished during product integration

Gate Integration Developers: Product integration for the BasicHttpWcf operation driver

This page explains how to integrate products into INEXTOR using the Basic HTTP Operation Driver.

Setup

As illustrated on the page Gate Integration Developers: Data sources and flows, you need to:

- create some additional tables into the Gate database.
- redirect the views for the Product Integration catalog.

In the Gate SQL scripts, under the folder **ProductIntegration\WS**, you can find the SQL Script **WsCatalogOperationDriverIntegration.sql** that creates the following tables :

WSCatalogSkus	
Code	VARCHAR (20) NOT NULL
ActorCode	VARCHAR (20) NOT NULL
Variant	SMALLINT NOT NULL
VaultCode	VARCHAR (20) NOT NULL
Description	VARCHAR (80) NOT NULL
Activity	NVARCHAR (50) NOT NULL
Market	NVARCHAR (50) NOT NULL
StaticAttributesSettingsAndExtendedFieldsXml	XML NOT NULL
TaxLevelCode	VARCHAR (20) NULL
StartDate	DATETIME NOT NULL
EndDate	DATETIME NULL
Origin	TINYINT NOT NULL

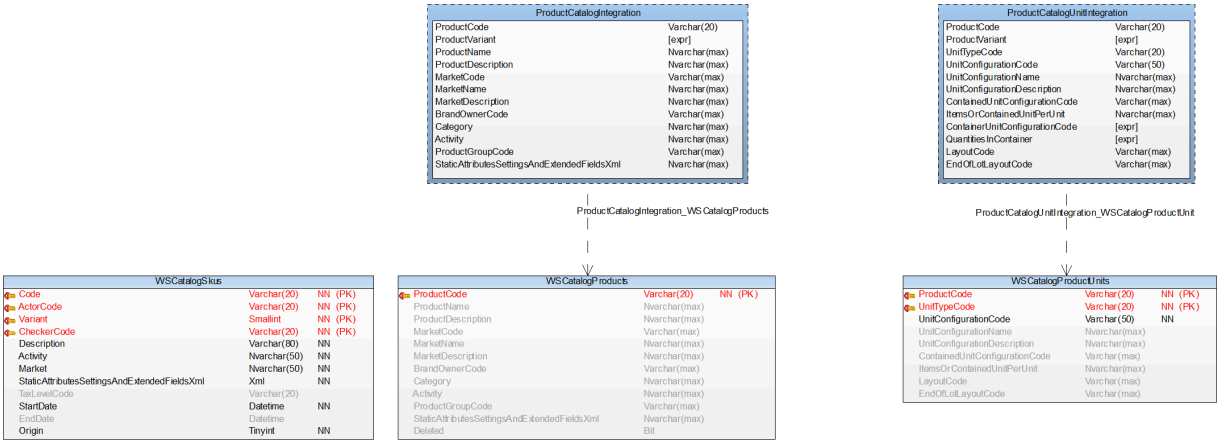
WSCatalogProducts	
ProductCode	VARCHAR (20) NOT NULL
ProductName	NVARCHAR (max) NULL
ProductDescription	NVARCHAR (max) NULL
MarketCode	VARCHAR (max) NULL
MarketName	NVARCHAR (max) NULL
MarketDescription	NVARCHAR (max) NULL
BrandOwnerCode	VARCHAR (max) NULL
Category	NVARCHAR (max) NULL
Activity	NVARCHAR (max) NULL
ProductGroupCode	VARCHAR (max) NULL
StaticAttributesSettingsAndExtendedFieldsXml	NVARCHAR (max) NULL
Deleted	BIT NULL

WSCatalogProductUnits	
ProductCode	VARCHAR (20) NOT NULL
UnitTypeCode	VARCHAR (20) NOT NULL
UnitConfigurationCode	VARCHAR (50) NOT NULL

UnitConfigurationName	NVARCHAR (max) NULL
UnitConfigurationDescription	NVARCHAR (max) NULL
ContainedUnitConfigurationCode	VARCHAR (max) NULL
ItemsOrContainedUnitPerUnit	NVARCHAR (max) NULL
LayoutCode	VARCHAR (max) NULL
EndOfLotLayoutCode	VARCHAR (max) NULL

The tables **WSCatalogProducts** and **WSCatalogProductUnits** are the equivalent of the product integration views **ProductCatalogIntegration** and **ProductCatalogUnitIntegration** respectively.

The table **WSCatalogSkus** stores the information of the SKUs before they are used in a manufactured product.



There is no physical foreign key between WSCatalogProducts and WSCatalogSkus. However the relationship between them is stored in the field StaticAttributesSettingsAndExtendedFieldsXml:

StaticAttributesSettingsAndExtendedFieldsXml Example

```
<Set ns="">
  <Set ns="Attributes">
      <P n="BPRICE" t="x" v="CHF 70.00" u="HttpOperationDriver" />
      <P n="PPRICE" t="x" v="CHF 7.00" u="HttpOperationDriver" /></Set>
  <Set ns="ExtendedFields">
      <P n="SkuActorCode" t="x" v="LE_TEST" u="HttpOperationDriver" />
      <P n="SkuVaultCode" t="x" v="TEST" u="HttpOperationDriver" />
      <P n="SkuCode" t="x" v="SKU0001" u="HttpOperationDriver" />
      <P n="SkuVariant" t="s" v="0" u="HttpOperationDriver" />
  </Set></Set>
```

1. The SKUs and Manufactured Products created using the webservices will be only visible in the Gate UI once the ProductIntegrationDriver has copied them. If you want to speed up the process, you should go to the Configuration - Op. Drivers section of the Gate and restart the ProductIntegrationDriver.
2. If the Market associated with the SKU does not exist when the SKU is going to be created using the SKU Interface, then it can be defined in two ways:
   a. Using the ExtendedFields of the SKU:
   For Example if we define <ExtendedFields>MarketCode,JP;MarketName,JAPAN</ExtendedFields> the Product Integration Driver will create the Market automatically. (MarketCode =JP, MarketName=Japan and MarketDescription=WSCatalogSkus.Market)
   b. If we don't provide the MarketCode and MarketName in the ExtendedFields, the Product Integration Driver will use for all of them the value WSCatalogSkus.Market (MarketCode =WSCatalogSkus.Market, MarketName=WSCatalogSkus.Market and MarketDescription=WSCatalogSkus.Market)

Product integration rules

Once the Manufactured product and the SKU are added to the WS tables, it is possible to create them on the Gate with the product integration rules.

Since the Basic HTTP driver supports passing SKUs that are named differently from the manufactured product, the following extended fields are added to the product:

- SkuCode
- SkuActorCode
- SkuVariant
- SkuVaultCode
- *Plus all the extended fields sent to the SKU interface*

Those extended fields are then available in the rules.

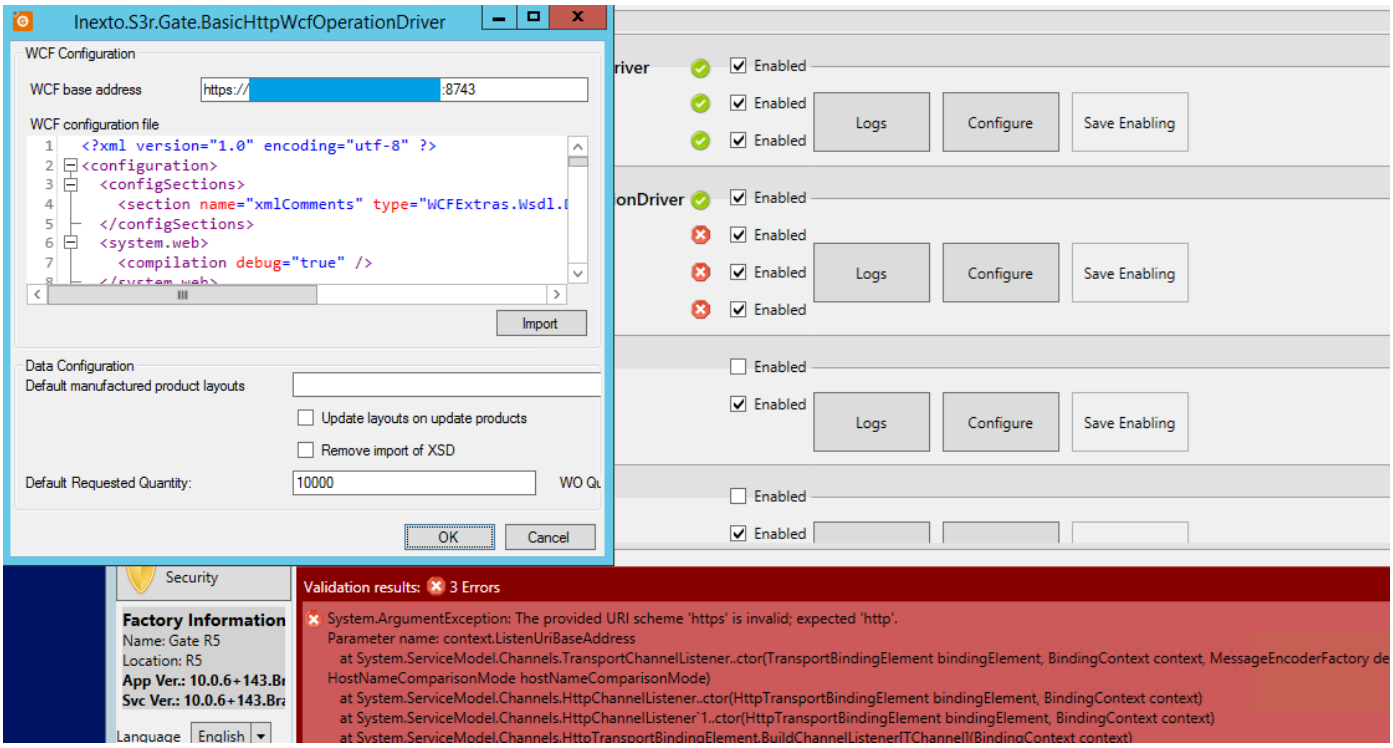
For example:

- To create a SKU with the code coming from a SKU sent to the SKUInterface, you have to put the following value in the *Sku code format* field of the product integration rule : "{SkuCode}".
- To create a SKU with the code and the variant coming from a SKU sent to the SKUInterface, you have to put the following value in the *Sku code format* field of the product integration rule : "{SkuCode}.{SkuVariant}".

How to configure an operation driver for https

A Windows Administrator must register a server certificate against the port of your web-service (see How to Register a Server Certificate for WCF Operation Drivers).

After you configure your WCF base address for https, you may get this error:



The base address applies to all endpoints that are defined in the WCF configuration file.

Consequently, **all these endpoints** must use a binding that supports the https scheme. The binding should be defined as follows:

```
<bindings>
 <basicHttpBinding>
   <binding name="secureBasicHttpBinding">
     <security mode="Transport">
        <transport clientCredentialType="None"/>
     </security>
   </binding>
</bindings>
```

In addition, you must set the httpsGetEnabled and httpGetEnabled properly for WSDL browsing:

```
    <serviceBehaviors>
        <behavior name="Cvs.MCGateway.OperationDrivers.ProductionInterfaceBehaviour">
          <serviceMetadata httpsGetEnabled="True" httpGetEnabled="False" />
          <serviceDebug includeExceptionDetailInFaults="True" />
        </behavior>
...
    </serviceBehaviors>
```

Dynamic Attributes Editor

Configuring Attribute Editors

It is possible to attach custom property set editors to some entities for editing their attributes and extended fields.

To enable this functionality, the **DynamicFormsFolderPath** setting in the Gate Service configuration file must be set to point to a folder on the file system that is accessible to the Gate.

Inside that folder can be placed xml documents that specify the appearance and operation of dynamic forms that can facilitate editing the attributes and extended fields of certain entities.

Currently Available Attribute Editors

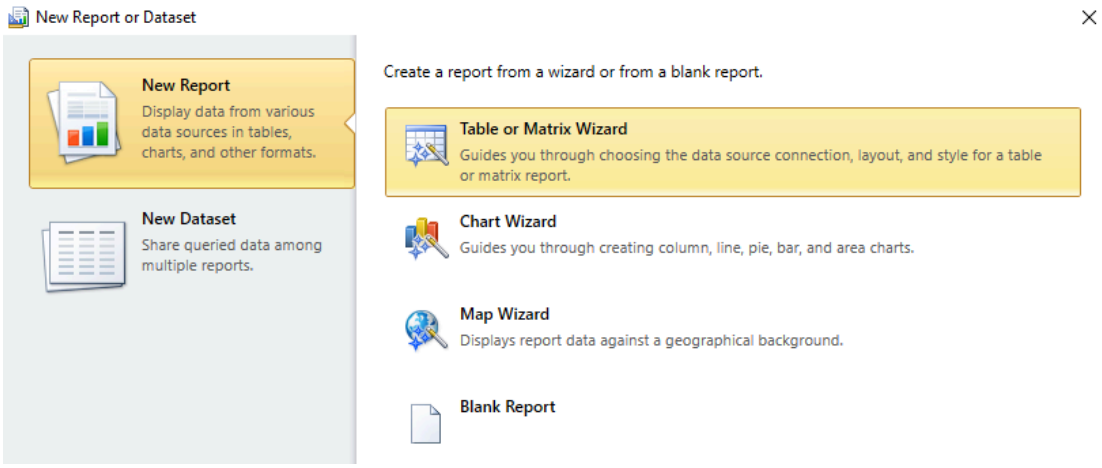
- INEXPRESS attribute editor in Production Order form
  Gate PO Tab: INEXPRESS Attributes Editor

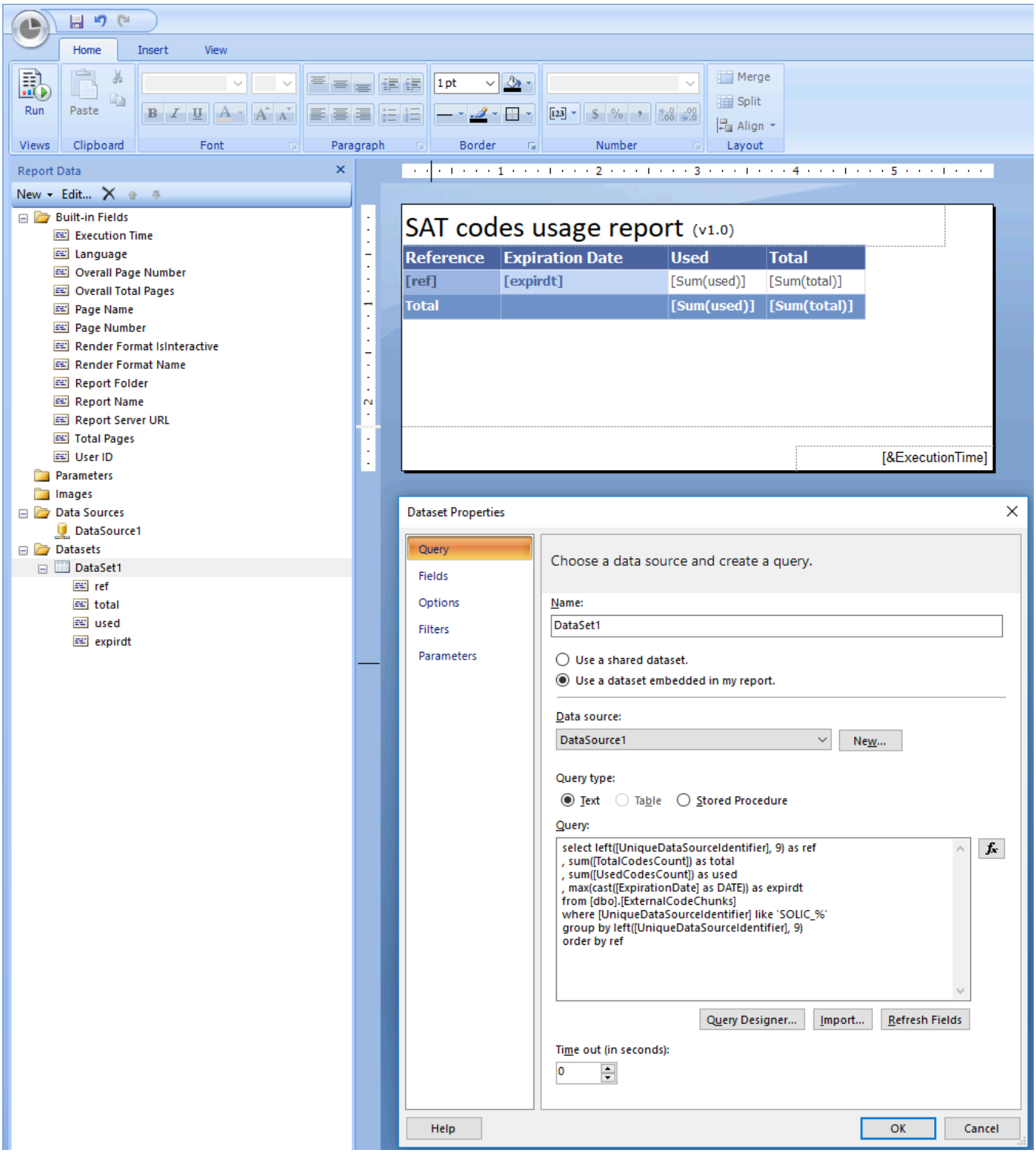
Creating or Editing Reports

You need to install **Microsoft® SQL Server® 2014 Report Builder** on your PC: https://www.microsoft.com/en-us/download/details.aspx?id=42301

You can use the wizards to create your report from an existing database:

When editing and modifying an existing report, it is important to increase the version number of the report.




Home
Insert
View
Run
Paste
Views
Clipboard
Font
Paragraph
Border
Number
Merge
Split
Align
Layout
Report Data
New
Edit...
Built-in Fields
Execution Time
Language
Overall Page Number
Overall Total Pages
Page Name
Page Number
Render Format IsInteractive
Render Format Name
Report Folder
Report Name
Report Server URL
Total Pages
User ID
Parameters
Images
Data Sources
DataSource1
Datasets
DataSet1
ref
total
used
expirdt
SAT codes usage report (v1.0)
Reference
Expiration Date
Used
Total
[ref]
[expirdt]
[Sum(used)]
[Sum(total)]
Total
[Sum(used)]
[Sum(total)]
[&ExecutionTime]
Dataset Properties
Query
Fields
Options
Filters
Parameters
Choose a data source and create a query.
Name:
DataSet1
Use a shared dataset.
Use a dataset embedded in my report.
Data source:
DataSource1
New...
Query type:
Text
Table
Stored Procedure
Query:
select left([UniqueDataSourceIdentifier], 9) as ref
, sum([TotalCodesCount]) as total
, sum([UsedCodesCount]) as used
, max(cast([ExpirationDate] as DATE)) as expirdt
from [dbo].[ExternalCodeChunks]
where [UniqueDataSourceIdentifier] like 'SOLIC_%'
group by left([UniqueDataSourceIdentifier], 9)
order by ref
Query Designer...
Import...
Refresh Fields
Time out (in seconds):
0
Help
OK
Cancel

Decompose codes with Ait Service (Gate)

See Decompose codes with Ait Service (Vault).

Product Catalog Integration Views

This page provides a comprehensive guide on defining the Views necessary for the Product Catalog Integration feature.

Legacy Product Catalog Integration Views (Pre-2022)

> **Important**: This documentation is only relevant for Gate versions prior to 2022. Current Gate versions no longer require these database views when using the External Code Service Provider.

Historical Context

Before v10.1.0, the Gate required two database views to enable product catalog integration:

1. **ProductCatalogIntegration View**
   - Stored basic product information (one record per product)
   - Required for catalog synchronization
   - Contained fields like product codes, variants, market info
2. **ProductCatalogUnitIntegration View**
   - Managed unit configurations (one record per product/unit type)
   - Handled packaging configurations (packs, bundles, cartons, etc.)
   - Stored layout information for each unit type

Legacy Implementation Guide

If you must maintain an older Gate installation (pre-2022), here's where to implement the required views:

Create the ProductCatalogIntegration view:

```
CREATE VIEW[dbo].[ProductCatalogIntegration]
  AS SELECT
    [ProductCode] as                                [ProductCode]        --Expect VARCHAR(20)
    ,[ProductVariant] as                             [ProductVariant]     -- Expect VARCHAR(2)
    ,coalesce([ProductName], [ProductCode]) as       [ProductName]        -- Expect NVARCHAR(80)
    ,coalesce([ProductDescription], [ProductCode]) as [ProductDescription] -- Expect NVARCHAR(80)
    ,[MarketCode] as                                 [MarketCode]         -- Expect VARCHAR(20)
    ,coalesce([MarketName], [MarketCode]) as         [MarketName]         -- Expect NVARCHAR(50)
    ,coalesce([MarketDescription], [MarketCode]) as  [MarketDescription]  -- Expect NVARCHAR(50)
    ,NULL as                                         [BrandOwnerCode]     -- Expect VARCHAR(20)
    ,[Category] as                                   [Category]           -- Expect NVARCHAR(20)
    ,[Activity] as                                   [Activity]           -- Expect NVARCHAR(50)
    ,NULL as                                         [ProductGroupCode]   -- Expect VARCHAR(20)
    ,[PropertiesXml] as                              [StaticAttributesSettingsAndExtendedFieldsXml]      --
Expect XML
FROM
    [dbo].[Products];
```

- **ProductCode (Mandatory)**
  - Description: The Manufactured Product Code without the variant.

- **ProductVariant (Mandatory)**
  - Description: The variant on two digits.
- **ProductName (Mandatory)**
  - Description: The name of the Manufactured Product.
- **ProductDescription (Mandatory)**
  - Description: The Description of the Manufactured Product.
- **MarketCode (Mandatory)**
  - Description: The Code of the Manufacturing Market.
- **MarketName (Mandatory)**
  - Description: The Name of the Manufacturing Market.
- **MarketDescription (Mandatory)**
  - Description: The Description of the Manufacturing Market.
- **BrandOwnerCode (Optional)**
  - Description: In case of TPM or CMO, an identifier of the entity owning the product.
    - Note: This information can be ignored if all products belong to the same brand owner for a certain market.
- **Category (Mandatory)**
  - Description: The Category of the Manufactured Product (used to organize the Manufactured Products in Gate).
- **Activity (Mandatory)**
  - Description: The Activity related to the Manufactured Product.
- **ProductGroupCode (Optional)**
  - Description: The Code of the Product Group or NULL.
- **StaticAttributesSettingsAndExtendedFieldsXml (Optional)**
  - Description: XML data that can be used to preset Static Attributes of a Manufactured Product.

Create the ProductCatalogUnitIntegration view:

```
CREATE VIEW [dbo].[ProductCatalogUnitIntegration]
AS
SELECT
     [ProductCode] as                    [ProductCode]
                                        -- Expect VARCHAR(20)
  ,[ProductVariant] as                  [ProductVariant]
                                        -- Expect smallint
    ,[UnitTypeCode] as                    [UnitTypeCode]
                                        -- Expect VARCHAR (20)
    ,[UnitConfigurationCode] as           [UnitConfigurationCode]
                                        -- Expect VARCHAR(50)
    ,[UnitConfigurationName] as           [UnitConfigurationName]
                                        -- Expect NVARCHAR(80)
    ,[UnitConfigurationDescription] as    [UnitConfigurationDescription]
                                        -- NVARCHAR(200)
    -- first strategy (standard approach)
    ,[ContainedUnitConfigurationCode] as [ContainedUnitConfigurationCode]
                                        -- Expect VARCHAR (20)
    ,[ItemsOrContainedUnitPerUnit] as     [ItemsOrContainedUnitPerUnit]
                                        -- Expect smallint
    -- second strategy (heterogenous packaging)
    ,[ContainerUnitConfigurationCode] as [ContainerUnitConfigurationCode]
                                        -- Expect VARCHAR (20)
```

```
    ,[QuantitiesInContainer] as          [QuantitiesInContainer]
                                        -- Expect smallint
    ,[LayoutCode] as                     [LayoutCode]
                                        -- Expect VARCHAR (50)
    ,[EndOfLotLayoutCode] as             [EndOfLotLayoutCode]
                                        -- Expect VARCHAR (50)
FROM
    [dbo].[XXX]
```

- **ProductCode (Mandatory)**
  - Description: The Manufactured Product Code without the variant.
- **ProductVariant (Mandatory)**
  - Description: The variant on two digits.
- **UnitTypeCode (Mandatory)**
  - Description: The Code of the Unit Type (e.g., Bundle) as defined in the Gate.
- **UnitConfigurationCode (Mandatory)**
  - Description: The Code of the Unit Configuration (e.g., StandardBundle_10) related to that UnitType.
- **UnitConfigurationName (Mandatory)**
  - Description: The Name of the Unit Configuration.
- **UnitConfigurationDescription (Mandatory)**
  - Description: The Description of the Unit Configuration.
- **ContainedUnitConfigurationCode (Optional)**
  - Description: The Code of the Unit Configuration that is contained in the Unit Configuration above (e.g., PACK/CARTON).
- **ItemsOrContainedUnitPerUnit (Mandatory)**
  - Description: The quantity of the Item that is contained in the Unit Configuration above (e.g., 10).
- **ContainerUnitConfigurationCode (Optional)**
  - Description: For reserved use.
- **QuantitiesInContainer (Optional)**
  - Description: For reserved use.
- **LayoutCode (Optional)**
  - Description: The Code of the Layout (as defined in the Gate).
- **EndOfLotLayoutCode (Optional)**
  - Description: The Code of the end of lot Layout (as defined in the Gate).

Static Attributes and Extended Fields

Below is an example demonstrating the creation of the `StaticAttributesSettingsAndExtendedFieldsXml` field:

```
CAST(N'<Set ns="">
  <Set ns="Attributes">
  </Set>
  <Set ns="Settings">
  </Set>
  <Set ns="ExtendedFields">
    <P n="X_INDICATOR" t="x" v="'+[X_INDICATOR]+'" u="" />
    <P n="Y_INDICATOR" t="x" v="'+[Y_INDICATOR]+'" u="" />
  </Set>
</Set>' AS xml) as [StaticAttributesSettingsAndExtendedFieldsXml]
```

Explanation of Elements:

- **ns**: Represents the namespace.
- **P**: Stands for Property.
- **n**: Denotes the name.
- **t**: Indicates the type.
  - **x**: Represents a C# string / SQL nvarchar.
  - **i**: Represents a C# int / SQL int.
- **v**: Represents the value.
- **u**: Indicates the user who has set the value.

> The `ExtendedFields` can be utilized as tokens in the Product Integration Filtering (see Product Integration).

Create the helper stored procedures:

```
CREATE PROCEDURE [dbo].[BeginProductCatalogScan]
    @productCount int out,
    @productUnitCount int out,
    @message VARCHAR(max) out,
    @result bit out
AS
BEGIN
    SET @productCount = 0;
    SET @productUnitCount = 0;
    SET @message = 'OK';
    SET @result = 1;
END
```

```
CREATE PROCEDURE [dbo].[EndProductCatalogScan]
    @message VARCHAR(max) out,
    @result bit out
AS
BEGIN
    SET @message = 'OK';
    SET @result = 1;
END
```

Performance Considerations

To enhance performance, it is important to ensure that the tables used by the views are indexed appropriately. For the **ProductCatalogIntegration**, the recommended indexing is on **(ProductCode, ProductVariant)**, while for the **ProductCatalogUnitIntegration**, the indexing should be on **(ProductCode, ProductVariant, UnitTypeCode)**.

Additionally, stored procedures can significantly optimize the synchronization process of products. By executing these procedures before and after each synchronization round, you can create snapshots (materializing the views with the necessary indexed tables), thereby boosting the performance of the Product Catalog Integration.

Important Notes

1. These views are only required for older version of the Gate (released before 2022)
2. Current Gate versions handle product catalog integration differently
3. If you're setting up a new installation, use the current Gate version instead

4. No migration is needed - newer versions work without these views

Integration Developers: Web Services List

Currently Used Web Services

For each Web Service in this chapter, clicking on the name will take you to a page which provides the documentation for that service.

Vault

On the Vault, Web Services are always exposed.

Web Service Name	Contract Name	Type	Authentication Type	Description
AIT (V3)	Deserialization	SOAP	Certificate, Windows authentication	Used to decompose codes into identifiers and production identification.
	ProductionTT	SOAP	Certificate, Windows authentication	Used to get production information from production identification.
	Serialization	SOAP	Certificate, Windows authentication	Used to rebuild codes.
	ActivationAndPairingService	SOAP	Certificate, Windows authentication	Provides an interface for code activation and pairing.
	VaultSyndicationService	SOAP	Certificate, Windows authentication	Used to allow sharing information between Vault and INEXTRACE in order to allow INEXTRACE to route the code verification correctly.
VaultWS	MasterDataManager	SOAP	Certificate, Windows authentication	Provides an interface in order to manage entities for the TPD (economic operators, Facilities, Machines), as well as Actors and SKUs.
	Actor	REST	Anonymous	Provides an interface in order to manage Actors.
	Code	REST	Anonymous	REST equivalent of AITv3 SOAP Services.
	EconomicOperator	REST	Anonymous	Provides an interface in order to manage economic operator for the TPD.
	Facility	REST	Anonymous	Provides an interface in order to manage Facilities for the TPD.
	Gate	REST	Anonymous	Provides an interface in order to retrieve information on Gates configured on the Vault.
	Inextend	REST	Anonymous	???
	InextracePortal	REST	Anonymous	Used to allow INEXTRACE to query codes.
	Machine	REST	Anonymous	Provides an interface in order to manage Machines for the TPD.
	Materialization	REST	Anonymous	???
	Metric	REST	Anonymous	Provides an interface in order to retrieve metrics stored on the Vault.
	Public	REST	Anonymous	Provides an interface in order to ping the Vault and check the Vault health.
	Sku	REST	Anonymous	Provides an interface in order to manage SKUs
EPCIS Query	EPCISQueryService	SOAP	Certificate, Windows authentication	???
TrackingInterfacesService	CodeDecomposerWs	SOAP	Certificate, Windows authentication	Used by INEXTRACK to decompose codes into identifiers and production identification.
	CodeRebuilderWs	SOAP	Certificate, Windows authentication	Used to rebuild codes.
	CodeActivatorWs	SOAP	Certificate, Windows authentication	???
	InterfaceInformationWs	SOAP	Certificate, Windows authentication	Used to return the basic universe information
WCFCheckerService	CheckerWCFService	SOAP	Certificate, Windows authentication	Used to get codes information and generate SUPIs
PortalQueryService	QueryInterface	SOAP	Certificate, Windows authentication	Used to decompose codes into identifiers and production identification.
	CheckerSyndicationWcfService	SOAP	Certificate, Windows authentication	

Gate

On the Gate, Web Services are exposed by operation drivers, which must be activated.

Authentication types can be chosen in the driver configuration, so they are not shown here.

Operation Driver Name	Web Service Name	Contract Name	Type	Description
External Code Provider	ExternalCodeProviderWsProxy	ExternalCodeProvider	SOAP	Used to allow an external system to reserve, download and update usage of the external code chunks.
External Service	ExternalStockManagerProxy	ExternalStockManager	SOAP	
	ExternalProductionOrderManagerProxy	ExternalProductionOrderManager	SOAP	Used to transmit POs for external code ordering and to check code availability. 1 version REST, 1 version SOAP.
	ExternalProductManagerProxy	ExternalProductManager	SOAP	See page Gate Integration Developers: Data sources and flows (this service is the modern version of IManufacturedProductInterface). 1 version REST, 1 version SOAP.

Operation Driver Name	Web Service Name	Contract Name	Type	Description
	ExternalLizManagerProxy	ExternalLizManager	SOAP	
	GateMasterData See Web Services List: External Service Operation Driver	Lizs	REST	
		ProductionManager	REST	
		Products	REST	
		Stocks	REST	
Basic Http Operation	ManufacturedProductInterface	ManufacturedProductInterface	SOAP	See page Gate Integration Developers: Data sources and flows.
	ProductionInterface	ProductionInterface	SOAP	Used to drive production (start, stop, get status). If the Production Order does not exist, it will create it.
	SKUInterface	SKUInterface	SOAP	Used to create or update SKUs
	GateProduction	GateProduction	REST	Used to start and stop Production Orders
Interfaces Driver	InterfacesGateServiceWsProxy	SerializationService	SOAP	
		DeserializationService	SOAP	
		DataService	SOAP	
		GateInterfaceWs	SOAP	

Liz

On the Liz, Web Services are exposed by operation drivers, which must be activated.

Authentication types can be chosen in the driver configuration, so they are not shown here.

Operation Driver Name	Web Service Name	Contract Name	Type	Description
PMI MES	PmiMesServiceWsProxy	IPmiMesServiceWs	SOAP	Used to start and stop production from PMI MES.
Print On Demand	PrintOnDemand	PrintOnDemand	REST	Used to manage print jobs for a limited amount of prints and copies. Typically used by INEXLINE in scenarios involving lots of manual operations.

Unused and Obsolete Web Services

Web Services in this chapter are not further documented (with a few exceptions). Should you need more explanations on one of the elements of this list, please contact Inexto Support.

Vault

Web Service Name	Contract Name	Type	Authentication Type	Description
Actor	ActorController	REST	Anonymous	Used to manage actors.
AIT (V1)	CheckerSyndication	SOAP	Certificate, Windows authentication	Used to allow sharing information between Vault and INEXTRACE in order to allow INEXTRACE to route the code verification correctly.
	Deserialization	SOAP	Certificate, Windows authentication	Used to decompose codes into identifiers and production identification.
	ProductionTT	SOAP	Certificate, Windows authentication	Used to get production information from production identification.
	Serialization	SOAP	Certificate, Windows authentication	Used to rebuild codes.
EconomicOperator	EconomicOperatorController	REST	Anonymous	Used to manage economic operators.
Facility	FacilityController	REST	Anonymous	Used to manage facilities.
Gate	GateController	REST	Anonymous	Used to manage Gates.
Machine	MachineController	REST	Anonymous	Used to manage machines on the line.
MasterDataService	MasterDataManager	SOAP	Certificate, Windows authentication	Used to manage Master Data (economic operator registration, etc.).
Navisat	NaviSatCodeService	SOAP	Certificate, Windows authentication	Used to associate custom information to codes, to query codes and to rebuild codes.
SKU	SkuController	REST	Anonymous	Used to manage SKUs
Stock	StockController	REST	Anonymous	Used to manage stocks.
VaultWS	Stocks	REST	Anonymous	provides an interface in order to manage Stocks.

Gate

Operation Driver Name	Web Service Name	Contract Name	Type	Description
Automation Service	EditionAutomationServiceWsProxy	EditionAutomationServiceWs	SOAP	Used to configure the Gate.
	ProductionAutomationServiceWsProxy	ProductionAutomationServiceWs	SOAP	Used to implement a new GUI to operate the Gate (Remark: Has never been used) (Also on Liz).
		GateStock	REST, SOAP	Used to manage stocks. 1 version REST, 1 version SOAP (replaced by ExternalStockManager)
Interfaces Driver v1	InterfacesGateServiceWsV1Proxy	SerializationTTService	SOAP	
		IDeserializationTTService	SOAP	

		IProductionTTService	SOAP	

Liz

Operation Driver Name	Web Service Name	Contract Name	Type	Description
Code Activation Driver	CodeInformationService	CodeInformation	SOAP	Used to report code status events or code pairings from external systems (Remark: Has never been used).
Automation Service	ProductionAutomationServiceWsProxy	CodeServerWS	SOAP	Used to tell Liz to generate codes without printing them (Remark: Has never been used).
		LineService	SOAP	Used to orchestrate equipment services on the line (Remark: Has never been used).
		IProductionAutomationServiceWs	SOAP	Used to implement a new GUI to operate the Liz (Remark: Has never been used) (Also on Gate).

External Web Services

External Web Services are not developed within the INEXTOR product, but have been specified by INEXTOR in order to be used by it. They are documented in page External Web Services.

Vault web services

- Vault Web Service: AIT Web Service, V3 (Vault)
- Vault Web Service: VaultWS Web Service (Vault)
- Vault Web Service: EPCIS Query Web Service (Vault)
- Vault Web Service: TrackingInterfacesService Web Service (Vault)
- Vault Web Service: WCFCheckerService Web Service (Vault)
- Vault Web Service: PortalQueryService Web Service (Vault)
- Vault Web Service: AIT Web Service, V1 (Vault)

Vault Web Service: AIT Web Service, V3 (Vault)

This is the current version, AIT V3.

> This Web Service is also known as Interfaces Service.

Deserialization Contract

File	Modified
IDeserializationService.wsdl	Jun 30, 2022 by Nathalie Chiva
IDeserializationService.html	Jun 30, 2022 by Nathalie Chiva

Download All

IDataService Contract

File	Modified
IDataService.wsdl	Jun 11, 2024 by Jean-Marie Pirelli
IDataService.html	Jun 11, 2024 by Jean-Marie Pirelli

Download All

Serialization Contract

File	Modified
ISerializationService.html	Jun 30, 2022 by Nathalie Chiva
ISerializationService.wsdl	Jun 30, 2022 by Nathalie Chiva

Download All

ActivationAndPairingService Contract

Provides an interface for code activation and pairing.

File	Modified
IActivationAndPairingService.html	Jun 29, 2022 by Nathalie Chiva
IActivationAndPairingService.wsdl	Jun 29, 2022 by Nathalie Chiva

Download All

VaultSyndicationService Contract

File	Modified
IVaultSyndicationService.html	Jul 13, 2022 by Jean-Marie Pirelli
IVaultSyndicationService.wsdl	Jul 13, 2022 by Jean-Marie Pirelli

Download All

Vault Web Service: VaultWS Web Service (Vault)

TPD and Master Data

MasterDataManager Contract

Provide interfaces specific for managing entities for the TPD (Economic Operators, Facilities, and Machines).

It provides also an interface for Actors and SKUs (master data).

File	Modified
IMasterDataManager.html	Jun 29, 2022 by Nathalie Chiva
IMasterDataManager.wsdl	Jun 29, 2022 by Nathalie Chiva

Download All

TPD

EconomicOperator Contract

This contract provides an interface in order to manage economic operator for the TPD.

File	Modified
Vault.EconomicOperator.html	Jul 06, 2022 by Jean-Marie Pirelli
Vault.EconomicOperator.json	Jul 06, 2022 by Jean-Marie Pirelli

Download All

Facility Contract

This contract provides an interface in order to manage facilities for the TPD.

File	Modified
Vault.Facility.html	Jun 29, 2022 by Nathalie Chiva
Vault.Facility.json	Jun 29, 2022 by Nathalie Chiva

Download All

Machine Contract

This contract provides an interface in order to manage machines for the TPD.

File	Modified
Vault.Machine.json	Jun 29, 2022 by Nathalie Chiva
Vault.Machine.html	Jun 29, 2022 by Nathalie Chiva

Download All

Master Data

Actor Contract

This contract provides an interface in order to retrieve and get information on actors.

File	Modified
Vault.Actor.html	Jun 29, 2022 by Nathalie Chiva
Vault.Actor.json	Jun 29, 2022 by Nathalie Chiva

Download All

Sku Contract

This contract provides an interface in order to manage SKUs.

File	Modified
Vault.Sku.html	Jun 29, 2022 by Nathalie Chiva
Vault.Sku.json	Jun 29, 2022 by Nathalie Chiva

Download All

Gate Contract

This contract provides an interface in order to retrieve and get information on Gates as configured on the Vault.

File	Modified
Vault.Gate.html	Jun 29, 2022 by Nathalie Chiva
Vault.Gate.json	Jun 29, 2022 by Nathalie Chiva

Download All

Stocks Contract

Deprecated. This contract provides an interface in order to manage stocks.

Code Checking

Code Contract

This contract is used by INEXTRACE. It is a REST equivalent of AITv3 SOAP Services.

File	Modified
Vault.Inextrace.CodeController.html	Jun 29, 2022 by Nathalie Chiva
Vault.Inextrace.CodeController.json	Jun 29, 2022 by Nathalie Chiva

Download All

InextracePortal Contract

This contract is a REST interface in order to check codes and return production information.

File	Modified
Vault.InextracePortal.html	Jun 29, 2022 by Nathalie Chiva
Vault.InextracePortal.json	Jun 29, 2022 by Nathalie Chiva

Download All

Other

Inextend Contract

File	Modified
Vault.Inextend.html	Jul 06, 2022 by Jean-Marie Pirelli
Vault.Inextend.json	Jul 06, 2022 by Jean-Marie Pirelli

Download All

Materialization Contract

File	Modified
Vault.Materialization.json	Jul 06, 2022 by Jean-Marie Pirelli
Vault.Materialization.html	Jul 06, 2022 by Jean-Marie Pirelli

Download All

Metric Contract

This contract provides an interface in order to retrieve metrics stored on the Vault.

File	Modified
Vault.Metric.html	Jul 06, 2022 by Jean-Marie Pirelli
Vault.Metric.json	Jul 06, 2022 by Jean-Marie Pirelli

Download All

Public Contract

This contract provides an interface in order to ping the Vault and check the Vault health.

File	Modified
Vault.Public.html	Jul 06, 2022 by Jean-Marie Pirelli
Vault.Public.json	Jul 06, 2022 by Jean-Marie Pirelli

Download All

Vault Web Service: EPCIS Query Web Service (Vault)

EPCISQueryService Contract

File	Modified
IEPCISQueryService.html	Jul 13, 2022 by Jean-Marie Pirelli
IEPCISQueryService.wsdl	Jul 13, 2022 by Jean-Marie Pirelli

Download All

Vault Web Service: TrackingInterfacesService Web Service (Vault)

Specificity

ProductVariant is returned as "-1" for RRP devices and Generic products.

CodeDecomposerWs Contract

This SOAP contract provides an interface for code decomposition.

File	Modified
ICodeDecomposerWs.html	Jun 29, 2022 by Nathalie Chiva
ICodeDecomposerWs.wsdl	Jun 29, 2022 by Nathalie Chiva

Download All

CodeRebuilderWs Contract

This SOAP contract provides an interface for code rebuild.

File	Modified
ICodeRebuilderWs.html	Jun 29, 2022 by Nathalie Chiva
ICodeRebuilderWs.wsdl	Jun 29, 2022 by Nathalie Chiva

Download All

CodeActivatorWs Contract

This SOAP contract provides an interface for code activation.

File	Modified
ICodeActivatorWs.html	Jun 29, 2022 by Nathalie Chiva
ICodeActivatorWs.wsdl	Jun 29, 2022 by Nathalie Chiva

Download All

InterfaceInformationWs Contract

This SOAP contract provides an interface for ???

File	Modified
IInterfaceInformationWs.html	Jun 29, 2022 by Nathalie Chiva
IInterfaceInformationWs.wsdl	Jun 29, 2022 by Nathalie Chiva

Download All

Vault Web Service: WCFCheckerService Web Service (Vault)

CheckerWCFService Contract

File	Modified
ICheckerWCFService.html	Jul 13, 2022 by Jean-Marie Pirelli
ICheckerWCFService.wsdl	Jul 13, 2022 by Jean-Marie Pirelli

Download All

Vault Web Service: PortalQueryService Web Service (Vault)

QueryInterface Contract

File	Modified
IQueryInterface.html	Jul 13, 2022 by Jean-Marie Pirelli
IQueryInterface.wsdl	Jul 13, 2022 by Jean-Marie Pirelli

Download All

CheckerSyndicationWcfService Contract

File	Modified
ICheckerSyndicationWcfService.html	Jul 13, 2022 by Jean-Marie Pirelli
ICheckerSyndicationWcfService.wsdl	Jul 13, 2022 by Jean-Marie Pirelli

Download All

Vault Web Service: AIT Web Service, V1 (Vault)

This is the previous version of the AIT Service, V1. It should not be used in any new installation.

CheckerSyndicationWcfService

File	Modified
ICheckerSyndicationWcfService.html	Jun 30, 2022 by Nathalie Chiva
ICheckerSyndicationWcfService.wsdl	Jun 30, 2022 by Nathalie Chiva

Download All

DeserializationTTService

File	Modified
IDeserializationTTService.wsdl	Jun 30, 2022 by Nathalie Chiva
IDeserializationTTService.html	Jun 30, 2022 by Nathalie Chiva

Download All

ProductionTTService

File	Modified
IProductionTTService.wsdl	Jun 30, 2022 by Nathalie Chiva
IProductionTTService.html	Jun 30, 2022 by Nathalie Chiva

Download All

SerializationTTService

File	Modified
ISerializationTTService.wsdl	Jun 30, 2022 by Nathalie Chiva
ISerializationTTService.html	Jun 30, 2022 by Nathalie Chiva

Download All

Web services list: External code provider operation driver

- Web Service: Gate: ExternalCodeProviderWsProxy

Web Service: Gate: ExternalCodeProviderWsProxy

ExternalCodeProvider Contract

This SOAP contract provides external code chunks to external systems (only for DLR-UK codes).

File	Modified
IExternalCodeProvider.wsdl	Jun 27, 2022 by Nathalie Chiva
IExternalCodeProvider.html	Jun 27, 2022 by Nathalie Chiva

Download All

Web services list: Basic Http operation driver

- Web Service: Gate: ManufacturedProductInterface Web Service (Gate)
- Web Service: Gate: ProductionInterface Web Service
- Web Service: Gate: SKUInterface Web Service
- Web Service: Gate: GateProduction Web Service

Web Service: Gate: ManufacturedProductInterface Web Service (Gate)

ManufacturedProductInterface Contract

This SOAP contract provides interfaces to manage Manufactured Products. (This interface is a legacy one).

File	Modified
IManufacturedProductInterface.html	Jun 29, 2022 by Nathalie Chiva (Deactivated)
IManufacturedProductInterface.wsdl	Jun 29, 2022 by Nathalie Chiva (Deactivated)

Download All

Web Service: Gate: ProductionInterface Web Service

ProductionInterface Contract

This SOAP contract provides interfaces to manage production. (This interface is a legacy one).

File	Modified
IProductionInterface.html	Jun 29, 2022 by Nathalie Chiva
IProductionInterface.wsdl	Jun 29, 2022 by Nathalie Chiva

Download All

Web Service: Gate: SKUInterface Web Service

SKUInterface Contract

This SOAP contract provides interfaces to manage SKUs. (This interface is a legacy one).

File	Modified
ISKUInterface.html	Jun 29, 2022 by Nathalie Chiva (Deactivated)
ISKUInterface.wsdl	Jun 29, 2022 by Nathalie Chiva (Deactivated)

Download All

Web Service: Gate: GateProduction Web Service

This is a REST service.

File	Modified
Gate.Production.json	Jul 11, 2022 by Nathalie Chiva (Deactivated)
Gate.Production.html	Jul 11, 2022 by Nathalie Chiva (Deactivated)

Download All

Web services list: Interfaces driver operation driver

- Operation Driver Interfaces: InterfacesGateServiceWsProxy

Operation Driver Interfaces: InterfacesGateServiceWsProxy

SerializationService Contract

This SOAP contract provides an interface for ???.

File	Modified
ISerializationService.html	Jun 29, 2022 by Nathalie Chiva
ISerializationService.wsdl	Jun 29, 2022 by Nathalie Chiva

Download All

DeserializationService Contract

This SOAP contract provides an interface for ???.

File	Modified
IDeserializationService.html	Jun 29, 2022 by Nathalie Chiva
IDeserializationService.wsdl	Jun 29, 2022 by Nathalie Chiva

Download All

DataService Contract

This SOAP contract provides an interface for code queries.???

File	Modified
IDataService.html	Jun 29, 2022 by Nathalie Chiva
IDataService.wsdl	Jun 29, 2022 by Nathalie Chiva

Download All

GateInterfaceWs Contract

File	Modified
IGateInterfaceWs.html	Jul 13, 2022 by Jean-Marie Pirelli
IGateInterfaceWs.wsdl	Jul 13, 2022 by Jean-Marie Pirelli

Download All

Web services list: PMI MES operation driver

PMI MES Operation Driver Web Service: PmiMesServiceWsProxy

IPmiMesServiceWs Contract

This SOAP contract provides an interface with the PMI MES system.

File	Modified
IPmiMesServiceWs.html	Jun 29, 2022 by Nathalie Chiva (Deactivated)
IPmiMesServiceWs.wsdl	Jun 29, 2022 by Nathalie Chiva (Deactivated)

Download All

External Web Services

These Web Services are not developed within the INEXTOR product, but have been specified by INEXTOR.

Vault	Notification (TPD)	SOAP REST	Implemented inside Vault "TPD Notification Driver" Operation Driver (see Vault Op. Drivers: TPD Notification Driver). Provides interfaces for an external system to receive notifications from Vault: • Approval and rejection of registration, update and deregistration of EO, Facilities and Machines by an ID Issuer • UID order approval or rejection by an ID Issuer • UID order download complete • UID chunk download complete including the downloaded codes

File	Modified
INotificationService.html	Jul 06, 2022 by Nathalie Chiva (Deactivated)
INotificationService.wsdl	Jul 06, 2022 by Nathalie Chiva (Deactivated)
Tools.Notification.html	Jul 06, 2022 by Nathalie Chiva (Deactivated)
Tools.Notification.json	Jul 06, 2022 by Nathalie Chiva (Deactivated)

Download All

Web Services List: Print On Demand (Webservices, Liz)

Print On Demand Controller

This controller is used to manage print jobs with a limited amount of items to print and copies. It is exposed by the Print On Demand Operation Driver".

Exposed methods

Method	Endpoint	Description
GET	api/PrintOnDemand/Ping	Pings the controller and returns a boolean whether it is connected or not
GET	api/PrintOnDemand/Units	Returns the hierarchy of Units configured to work in Print On Demand mode and their devices: Unit⇒Device
GET	api/PrintOnDemand/Status?unit={Unit}	Returns the printer status
GET	api/PrintOnDemand/Quantities?unit={Unit}	Returns the quantities of generated and printed codes
GET	api/PrintOnDemand/Logs	Returns the validation messages currently displayed on the Liz. If the order number is not running, then nothing is returned
POST	api/PrintOnDemand/Print?unit={Unit}&items={NbOfItems}&copies={NbOfCopies=1}	Sends the specified number of codes to the printer buffer. Returns the number of codes successfully pushed to the printer. In case of success, it returns HTTP 200 and the result is expected to be NbOfItems * NbOfCopies. But if the TIMax limit is hit, the result can be a positive value (0 < x < NbOfItems * NbOfCopies). In case of bad request, it returns HTTP 400, and the reasons (e.g. invalid or missing parameters). If a request is already running, HTTP 400 is also returned. In case of error, it returns HTTP 500, and the result is usually 0.
PUT	api/PrintOnDemand/Amend?unit={Unit}&items={NbOfItems}	This method can be called only when a request is already running (printer is connected) to modify (either reduce or increase) it. The call changes the number of printed items with a new value. When the request gets reduced (new value < old value), the buffer does not get flushed (unless the NbOfItems = 0). This means that the codes that were already pushed to the printer buffer will anyway be printed. In case of success, it returns HTTP 200 and the result is expected to be NbOfItems * NbOfCopies. But if the TIMax limit is hit, the result can be a positive value (0 < x < NbOfItems * NbOfCopies). In case of bad request, it returns HTTP 400, and the reasons (e.g. invalid or missing parameters). If no request is already running, HTTP 400 is also returned. In case of error, it returns HTTP 500, and the result is usually 0.
DELETE	api/PrintOnDemand/Cancel	Sends a signal to cancel the current POD request. The printer will therefore stop printing. The PO is not stopped. In case of success, it returns HTTP 200. In case of error, it returns HTTP 500, and the error description.

File	Modified
PODv1.json	Oct 10, 2023 by Daniel Sgroi

Web Services List: External Service Operation Driver

IMPORTANT

The External Service Driver exposes not only its own REST services, but also all the REST services of the Gate, in order to configure only one URL for them.

The general documentation on the External Service Operation Driver can be found on page External Service Driver.

Component	HTTP	Method	Description
Lizs	GET	/api/lizs	Get a list of all the Lizs. List contains details like: status, last connection, quantity of codes inside, etc...
	GET	/api/lizbylizid/{lizId}	Get a specific Liz. Found by its Liz ID. Answer contains details like: status, last connection, quantity of codes inside, etc...
	GET	/api/lizbymachineId/{machineId}	Get a specific Liz. Found by its Machine ID. Answer contains details like: status, last connection, quantity of codes inside, etc...
Production Manager	POST	/api/mes/start	Start a PO on a specific machine. The answer contains a Message ID (GUID) that can be used with the methods below.
	POST	/api/mes/stop	Stop (pause) a PO on a specific machine.
		/api/stop	
	POST	/api/mes/change	Close the running PO and start a new one on a specific machine. The answer contains a Message ID (GUID) that can be used with the methods below.
	GET	/api/mes/production/{messageID}	Used to check that a production has correctly started. STATUS 0: Processing, 1: Success, 2: Failed.
	GET	/api/mes/terminate/{messageID}	Used to check that a production has correctly terminated. STATUS 0: Processing, 1: Success, 2: Failed.
	POST	/api/mes/production/terminate	Terminate the production running on a specific machine. An option allows closing the PO after termination.
		/api/production/terminate	
	GET	/api/mes/liz/{messageID}	Get info about production running on the Liz. Brand and available code quantity.
Production Orders	GET	/api/productionorders	Get list of all POs in the Gate. The list contains all the details of each PO.
	POST	/api/productionorders	Create a new PO.

	DELETE	/api/productionorders/{orderNumber}	Delete a PO.
	GET	/api/productionorders/{orderNumber}	Get a specific PO with all the details.
	PUT	/api/productionorders/{orderNumber}	Update an existing PO.
	POST	/api/productionorders/{overwrite}	Create a list of POs. Existing PO are updated. If the option overwrite is True, then the method will delete all the POs that are not in the request.
	GET	/api/productionorders/{orderNumber}/areRequestedQuantitiesAvailable	Return true if the PO has all the codes it has requested.
	GET	/api/productionorders/{orderNumbers}/areRequestedQuantitiesAvailableByKey	Send a list of PO separated by a ",". Return true for the POs that have all the codes they have requested.
	GET	/api/productionorders/{orderNumber}/quantities	Get details of PO's quantities. Available, produced, requested, etc...
	GET	/api/productionorders/{orderNumbers}/quantitiesByKey	Send a list of PO separated by a ",". Return, for each POs, the quantities. Available, produced, requested, etc...
	PUT	/api/productionorders/{orderNumber}/close	Close a PO.
Products	POST	/api/products/{overwrite}	Create a list of products. If overwrite is true, then the Gate starts by deleting all the products. Products are then transformed into Manufactured Products and SKU by a product integration rule.
	DELETE	/api/products/{productCode}/{productVariant}	Delete a product. Does not delete the Manufactured Products nor SKU.
	GET	/api/products/{productCode}/{productVariant}	Get a product. It only gets the products before the integration rule transform them into Manufactured Products and SKU.
	POST	/api/products/products	Create a product. Products are then transformed into Manufactured Products and SKU by a product integration rule.
	PUT	/api/products/{productCode}	Update an existing product. Products are then transformed into Manufactured Products and SKU by a product integration rule.
Stocks			Obsolete. Do not use.

PO: Production Order

External Service Operation: Driver ExternalStockManagerProxy

ExternalStockManager Contract

File	Modified
IExternalStockManager.html	Jun 29, 2022 by Nathalie Chiva
IExternalStockManager.wsdl	Jun 29, 2022 by Nathalie Chiva

Download All

External Service Operation: ExternalProductionOrderManager

See also Production Management

ExternalProductionOrderManager Contract

File	Modified
IExternalProductionOrderManager.html	Jun 29, 2022 by Nathalie Chiva
IExternalProductionOrderManager.wsdl	Jun 29, 2022 by Nathalie Chiva

Download All

External Service Operation: ExternalProductManager

See also Gate - Product Integration

ExternalProductManager Contract

File	Modified
IExternalProductManager.html	Jun 29, 2022 by Nathalie Chiva
IExternalProductManager.wsdl	Jun 29, 2022 by Nathalie Chiva

Download All

External Service Operation: ExternalLizManagerProxy

ExternalLizManager Contract

File	Modified
IExternalLizManager.wsdl	Sep 28, 2022 by Jeremy Samin
IExternalLizManager.html	Sep 28, 2022 by Jeremy Samin

Download All

External Service Operation: GateMasterData

Lizs Contract

This REST contract provides an interface for managing Lizs.

File	Modified
Gate.Liz.json	Sep 30, 2022 by Jeremy Samin
Gate.Liz.html	Sep 30, 2022 by Jeremy Samin

Download All

ProductionManager Contract

This REST contract provides an interface for managing production.

File	Modified
Gate.Production.html	Oct 04, 2022 by Jean-Marie Pirelli
Gate.Production.json	Oct 04, 2022 by Jean-Marie Pirelli

Download All

ProductionOrders Contract

This REST contract provides an interface for managing production orders.

File	Modified
Gate.ProductionOrder.json	Jun 29, 2022 by Nathalie Chiva
Gate.ProductionOrder.html	Jun 29, 2022 by Nathalie Chiva

Download All

Products Contract

This REST contract provides an interface for managing products.

File	Modified
Gate.Product.html	Jun 29, 2022 by Nathalie Chiva
Gate.Product.json	Jun 29, 2022 by Nathalie Chiva

Download All

Stocks Contract

This REST contract provides an interface for managing stocks (and demand planning as stock snapshots).

File	Modified
Gate.Stock.html	Jun 29, 2022 by Nathalie Chiva
Gate.Stock.json	Jun 29, 2022 by Nathalie Chiva

Download All

Integration Developers: Vault

Vault Integration Development: Supported Browsers

Supported browsers include:

- Microsoft Internet Explorer (versions 8, 9, 10 and 11)
- Google Chrome (version 23)
- Mozilla Firefox (version 17)
- Opera (version 12)

Vault Integration Development: Service HealthCheck

Overview

The *PublicController* methods now allow the User to verify the status of the service (Vault & Database) by using *Ping()* and *GetSystemHealth()*.

By using *GetSystemHealth()* the returned value will be a *SystemHealthInformation* object containing the *ComponentStatus* of the Vault and of the Database (coupled with an HTTP 200 status code) or an HTTP 503 status code.

GetSystemHealth()

If the Vault is operational the method will try contacting the Database through a query.

If the Database responds to this query then its status is marked as *Operational* (StatusDetail = "Database is working") otherwise its status is set to *MajorOutage* (StatusDetail = "Database is unresponsive").

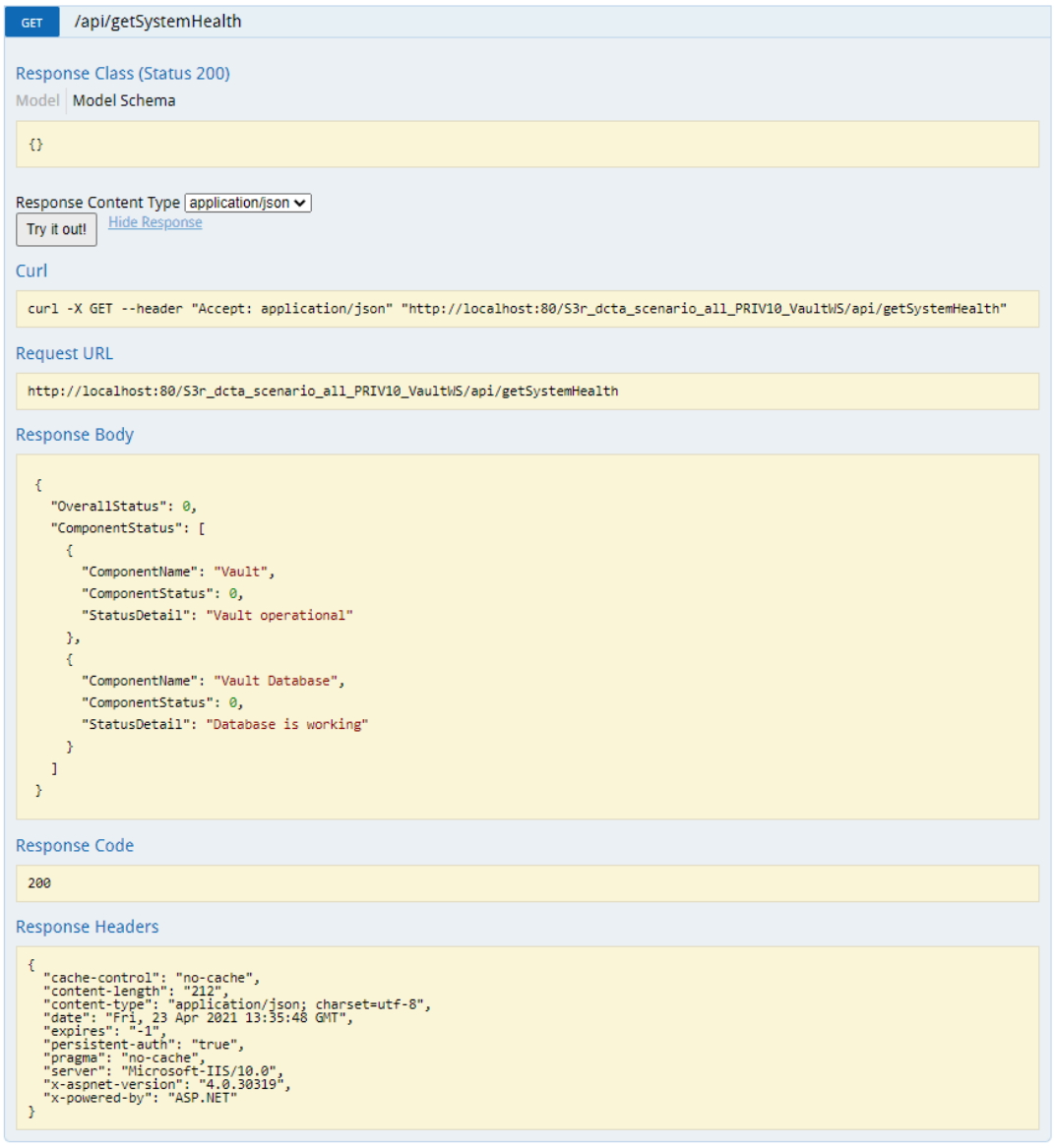


fig.1 Response from calling GetSystemHealth() with both Vault and Database up and running.

When the Vault is not responsive then an HTTP 503 status code is returned.

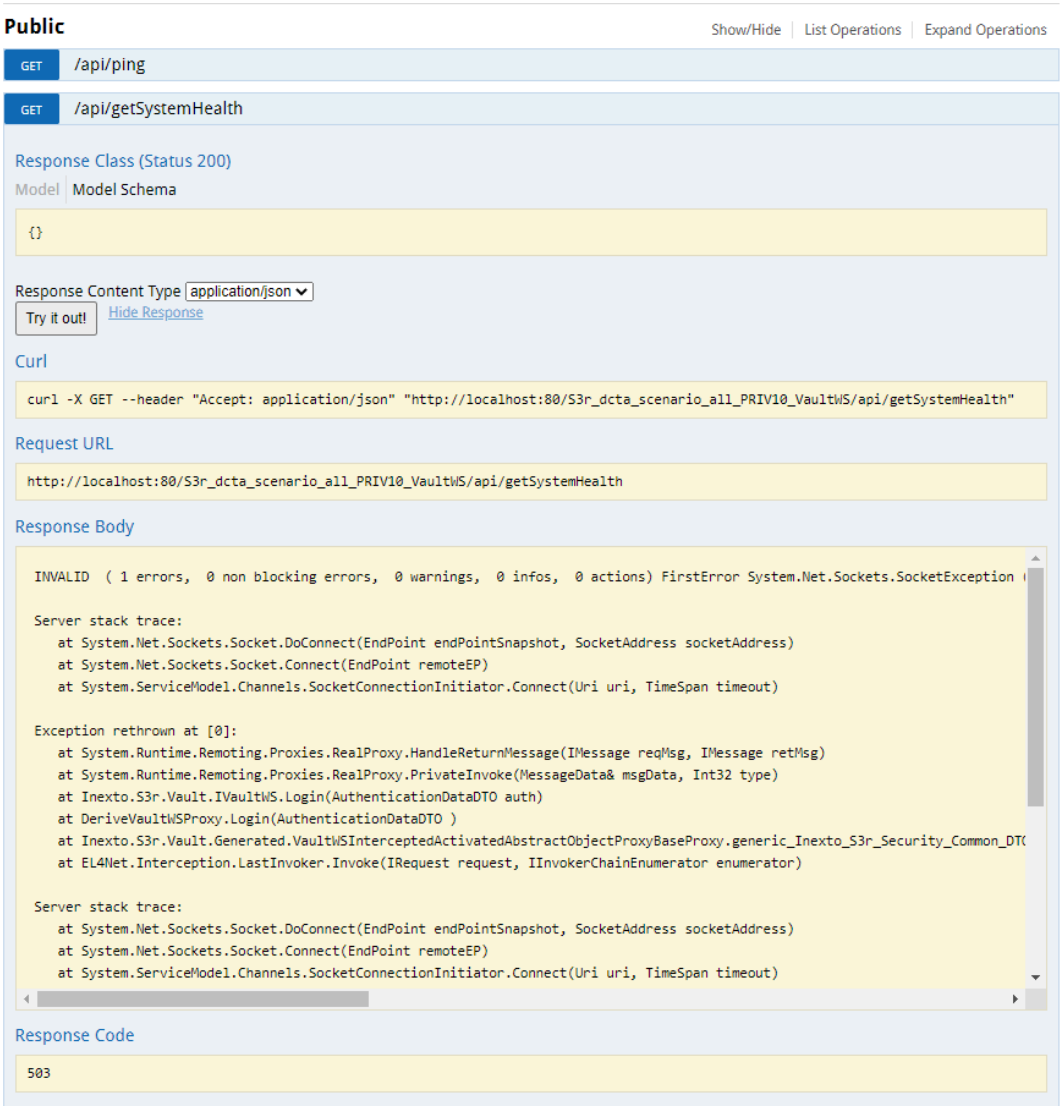


fig.2 Response from calling GetSystemHealth() when the Vault is down.

Inexto.Common.Healthcheck Status

The provided *GetSystemHealth()* service makes use of objects defined in the Inexto HealthCheck package (*SystemHealthInformation*, *ComponentStatus*, *Status Operational/MajorOutage*)

The code-status correspondence in the Inexto HealthCheck package is the following:

- [0] Operational
- [1] DegradedPerformance
- [2] Undermaintenance
- [3] PartialOutage
- [4] MajorOutage

Vault Integration Development: Decompose code with InterfacesService

The interfaces service can be accessed either via SOAP or REST (see Vault Web Service: AIT Web Service, V3 (Vault)).

Request parameters

Typically a SOAP request would be like

Interfaces service request

```
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
    xmlns:a="http://www.w3.org/2005/08/addressing">
    <s:Header>
        <a:Action s:mustUnderstand="1">DecomposeCodes</a:Action>
        <a:MessageID>urn:uuid:7f57e967-7838-4f7a-b1f0-445667154cb1</a:MessageID>
        <a:ReplyTo>
            <a:Address>http://www.w3.org/2005/08/addressing/anonymous</a:Address>
        </a:ReplyTo>
        <a:To s:mustUnderstand="1">https://staging-vault-api.inextor.biz/InterfacesService.svc/HttpsCertificate</a:
    </s:Header>
    <s:Body>
        <DecomposeCodes xmlns="http://inextor.com/tracking/2016/08">
            <queries xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
                <DecomposeQuery>
                    <AllowEarlyCheck>true</AllowEarlyCheck>
                    <RequestedCheckOptions>CheckCode</RequestedCheckOptions>
                    <RequestedCodeCheckActions>None</RequestedCodeCheckActions>
                    <RequestedCodeDisclosure>CodeInClear</RequestedCodeDisclosure>
                    <CodeTypeId i:nil="true"></CodeTypeId>
                    <Codes xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
                        <b:string>KPL2FCYnv3z6JfkqV8</b:string>
                        <b:string>LEDTCO6X7uV0R1tasZcY</b:string>
                    </Codes>
                </DecomposeQuery>
            </queries>
            <decomposeQueryOptions xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
                <AllOrOnlyIncludedCodeTypeIds i:nil="true"
                    xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"></AllOrOnlyIncludedCodeType
                <IncludeAlternateCodes>false</IncludeAlternateCodes>
                <IncludeGeoLocations>false</IncludeGeoLocations>
                <Location i:type="LocationIdQuery">
                    <Language i:nil="true"></Language>
                    <LocationId>0</LocationId>
                </Location>
                <QueryReason>DecomposeForTests</QueryReason>
                <Requestor>InterfacesServiceTester</Requestor>
                <ReturnContent>false</ReturnContent>
            </decomposeQueryOptions>
        </DecomposeCodes>
    </s:Body>
</s:Envelope>
```

Here is a list of parameters that can be entered in the request

Parameter	Description	Type
AllowEarlyCheck	This option allow to decompose codes even if the production data has not been received on the Vault yet.	boolean

<table>
<tr><th>Parameter</th><th>Description</th><th>Type</th></tr>
<tr><td>RequestedCheckOptions</td><td>Informs how the verification and check will be performed on the Vault (Checking, Flagging or not the code).</td><td>falgs<table><tr><th>Value</th><th>Name</th></tr><tr><td>0</td><td>None</td></tr><tr><td>1</td><td>GetCheckStatus</td></tr><tr><td>3</td><td>CheckCode</td></tr><tr><td>5</td><td>CountCheckHistory</td></tr><tr><td>13</td><td>GetCheckHistory</td></tr><tr><td>15</td><td>CheckCodeAndGetCheckHistory</td></tr></table></td></tr>
<tr><td>RequestedCodeCheckActions</td><td>Informs the action applied on check status.</td><td>flags<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>None</td></tr><tr><td>1<<0</td><td>Used</td></tr><tr><td>1<<16</td><td>RemoveUsed</td></tr><tr><td>1<<1</td><td>MarkAsCounterfeited</td></tr><tr><td>1 << 17</td><td>UnmarkAsCounterfeited</td></tr><tr><td>1<<2</td><td>DeclareAsWaste</td></tr><tr><td>1<<18</td><td>UndeclareAsWaste</td></tr><tr><td>1<<3</td><td>DeclareAsStolen</td></tr><tr><td>1 << 19</td><td>UndeclareAsStolen</td></tr><tr><td>1<<4</td><td>MarkAsEndOfLife</td></tr><tr><td>1 << 20</td><td>UnmarkAsEndOfLife</td></tr><tr><td>1<<5</td><td>MarkAsNonRepairableDamaged</td></tr><tr><td>1 << 21</td><td>UnmarkAsNonRepairableDamaged</td></tr><tr><td>1 << 6</td><td>MarkAsExported</td></tr><tr><td>1 << 22</td><td>UnmarkAsExported</td></tr><tr><td>1<<7</td><td>UnauthorizedCheck</td></tr></table></td></tr>
<tr><td>RequestedCodeDisclosure</td><td>This option allow to recompose partially the codes</td><td>enum<table><tr><th>Value</th><th>Name</th></tr><tr><td>0</td><td>NotAvailable</td></tr><tr><td>1</td><td>NoCode</td></tr><tr><td>2</td><td>IdentifyingPart</td></tr><tr><td>3</td><td>PartiallyObfuscated</td></tr><tr><td>4</td><td>CodeInClear</td></tr></table></td></tr>
<tr><td>CodeTypeId</td><td>This option allows to set the code type (see Standard Code Type IDs)</td><td>int</td></tr>
<tr><td>Codes</td><td>the codes</td><td>string array</td></tr>
<tr><td>AllOrOnlyIncludedCodeTypeIds</td><td>If not empty, provide the list of possible alternate codes the client system is</td><td>int array</td></tr>
</table>

<table>
<tr><th>Parameter</th><th>Description</th><th>Type</th></tr>
<tr><td></td><td>interested in to be returned.</td><td></td></tr>
<tr><td>IncludeAlternateCodes</td><td>Indicates if AlternateCodes have to be included</td><td>boolean</td></tr>
<tr><td>IncludeGeoLocations</td><td>Indicate if GetLocations are returned</td><td>boolean</td></tr>
<tr><td>Location</td><td>Indicates the origin of the verification</td><td>Language : ISO 639-1 Code for searching and returning information.
LocationId : Identifier of the location Id</td></tr>
<tr><td>QueryReason</td><td>Indicate the reason of the query being requested</td><td>int<table><tr><th>Title</th><th>Possible values</th></tr><tr><td>For normal code checks</td><td>CheckFromCustomer = 1000,
CheckFromAuthorities = 1001,
CheckFromInternal = 1002,
CheckFromQA = 1003,
CheckFromTrade = 1004,
CheckFromMedia = 1005,
CheckFromRetailer = 1006,
CheckFromManufacturer = 1007,
CheckFromDestroyManager = 1008,
CheckForTest = 1099</td></tr><tr><td>When Flagging the code</td><td>MarkAsCounterfeited = 1300,
DeclareAsWaste = 1301,
DeclareAsStolen = 1302,
MarkAsEndOfLife = 1303,
MarkAsNonRepairableDamaged = 1304,</td></tr><tr><td>When testing</td><td>DecomposeForTests = 1999</td></tr></table></td></tr>
<tr><td>Requestor</td><td>Indicates the name of the person or system who requested to check the code</td><td>string</td></tr>
<tr><td>ReturnContent</td><td>Indicate whether the codes have to be returned as they appear in the FormattedContent</td><td>boolean</td></tr>
</table>

Vault Integration Development: Installing the QTSP Signing Proxy

Introduction

> This page describes the installation of the QTSP Signing Proxy in a dedicated server owned and managed by the tobacco manufacturer. If you are an Inexto agent, and need to install the proxy on Inexto environment, please rather refer to Vault Integration Development: Installing the QTSP Signing Proxy - Inexto managed.

The QTSP Signing Proxy allows requests to be signed by an Italian Qualified Trust Service Provider (INTESI) before being sent to the Italian ID Issuer (ADM). By "requests" we mean those related to the registration/update/deletion of Economic Operators, Facilities & Machines, and to Code Ordering/Order cancelling.

In order to be signed, a request has to pass through the Proxy (hosted by the Manufacturers), where their signing credentials are stored in encrypted form and cannot be accessed by Inexto.

Prerequisites

This guide is a detailed explanation of one of the steps to be taken in the AAMS plugin setup", you'll need both to complete the installation.

- Windows Server 2016 (or more recent);
- IIS (Internet Information Service) installed [version 10.0.14393.0 or more recent];
- A client certificate to be provided to INEXTO should be generated for authentication from Vault application to the QTSP signing Proxy.
  This client certificate should be issued from a Root certificate trusted by the server where the proxy application is installed. It needs to be imported in the Vault and the proxy configuration needs it's *Thumbprint* (see step #3).
- A hostname should be defined to access the server remotely. This hostname should be declared in the DNS.
- The *ApplicationQtsp* solution (your can download it from here → ApplicationQTSP.zip).

The QTSP SigningProxy communicates with Intesi services, once the contract with Intesi Group is established you should have received the following pieces of information.

- Username of Intesi account;
- Password of Intesi account;
- Alias for the PkBox;
- PIN for the PkBox;
- Url of the PkBox.

These elements will be used later (see step #3 "Setup folder & app settings" and step #6 "Encrypt sensitive data").

Installation procedure

To install and have the SigningProxy correctly configured for production the following steps, each with its own section, need to be performed:

1. First IIS installation/configuration;
2. IIS Application Pool;
3. Setup project folder & app settings;
4. Create QTSP Web Application on IIS;
5. Verify that the QTSP Signing Proxy can be accessed;
6. Encrypt sensitive data;
7. Test QTSP Signing Proxy
8. Provide information (URL & Authentication Certificate)

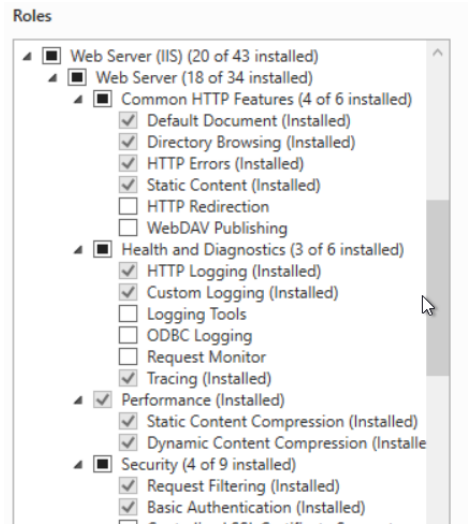
1 - First IIS configuration/installation

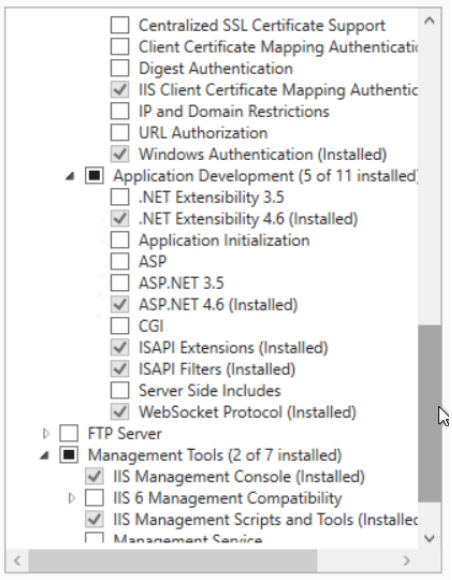
IIS Manager should already be installed and enabled on your server, if it's not the case follow this Microsoft guide:
https://docs.microsoft.com/en-us/dynamics-nav/how-to--install-and-configure-internet-information-services-for-microsoft-dynamics-nav-web-client#WS2012

Make sure SSL is enabled in IIS. If it's not the case, you can follow this guide:
https://learn.microsoft.com/en-us/iis/manage/configuring-security/how-to-set-up-ssl-on-iis

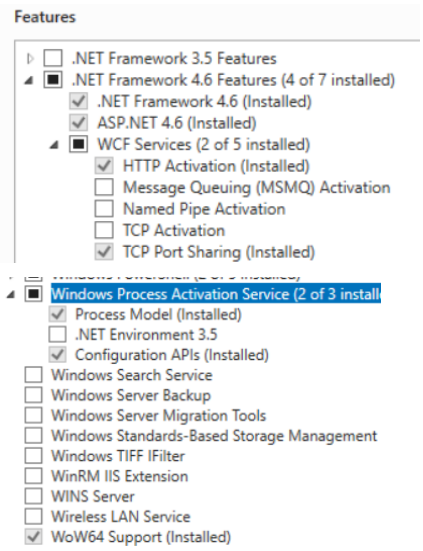
> The version of the .NET Framework and ASP.NET in the screenshots below reference 4.6, but depending on your version of Windows Server, the version can be higher (e.g. 4.8).

Open the Server Manager and enable the following Roles on the server:





The following features should be enabled on the server:



2 - IIS Application Pool

Open *IIS Manager* and under Connections right click on Application Pools, then select "*Add Application Pool...*" (admin permissions are needed to see the configuration).

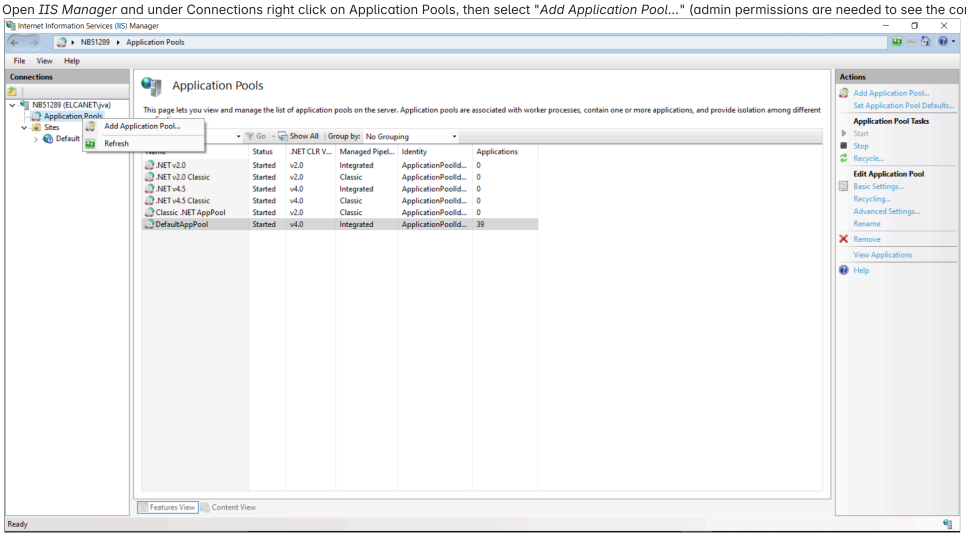


fig.1 Adding Application Pool

Insert the name of your application pool (ex. QtspSigningProxyAppPool), set *.NET CLR Version* to "*No Managed code*", leave managed pipeline mode to Integrated and click on OK.

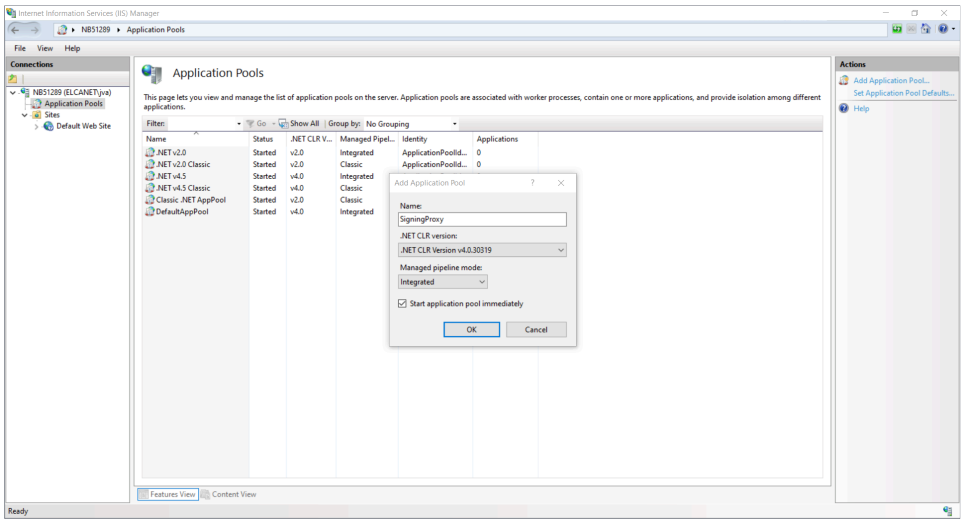


fig.2 Specifying Application Pool name and managed preference.

Now right click on the application pool, select advanced settings, scroll down to ***Process Model*** and set *Load User Profile* to *True*.

3 - Setup project folder & app settings

Extract the compiled project and place it in a sub-folder on your main disk (e.g. C:\Components).

Open *appsettings.Production.json* then:

- ensure that *EncryptSensitiveData* is set to false;
- Vault authenticates to the QTSP signing proxy using certificate authentication. Set *CertificateThumbprint* with the thumbprint of the certificate you will provide *Inexto* to communicate with the Proxy (this requires having a Trusted Root Certificate you can share already installed or installing one);
- fill *Username* & *Alias* with the values received by Intesi. These values are used to connect to Intesi system to apply the digital signature to messages sent to the Italian ID Issuer.
- update the *PKBoxUrl* with the url Intesi provided to the customer.
- if needed, update the path to the log files to a location of your preference (e.g. C:/Components/ApplicationQTPS/Logs/QtspSigningProxy-.log).

Save your changes and close the file.

4 - Create QTSP Web Application on IIS

Launch Internet Information Service.

In *Connections*, under the server root right click on *Sites* and select *"Add Website.."*.

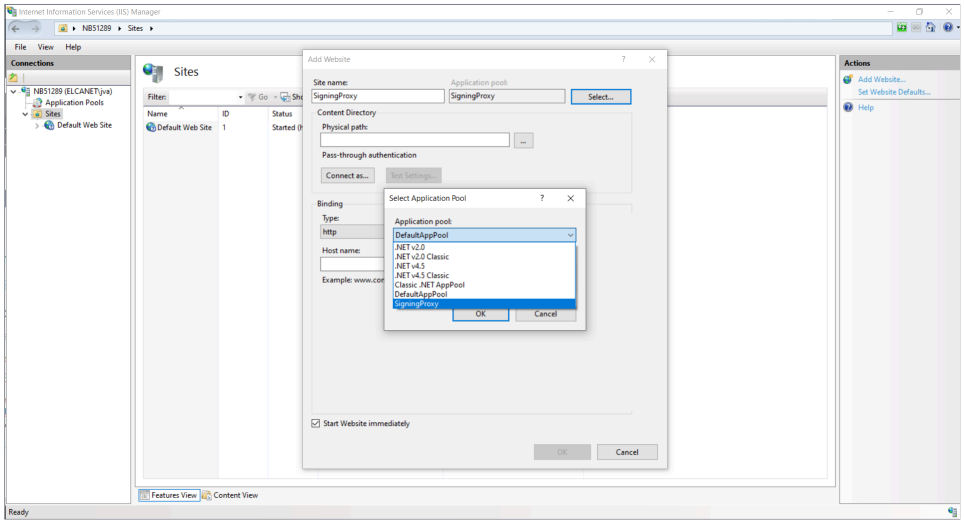


fig.3 Selecting the Application Pool defined earlier.

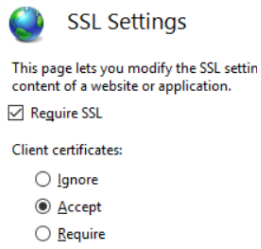
Define the site name (ex. *SigningProxy*), select the application pool defined in step #2.

Select the folder where you placed the SigningProxy files in step #3.

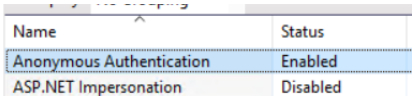
Set the binding type to *https*, then choose a hostname (ex. *signing-proxy.<company-name>.com*) then finally click on OK. **The hostname should be declared in your DNS.**

Give rights to "IIS AppPool\<Name of the AppPool>" (e.g. IIS AppPool\SigningProxyPool): right click on the site you just created, select edit permissions, click on the *Security* tab then "*Edit...*", then "*Add*" and give all permissions to the app pool account (location is the current server, the object name is "*IIS AppPool\QtspSigningProxyAppPool*", hit "*Check Names*" to ensure that it's recognized).

Select the site you just created and Edit the SSL settings according to the screenshot below (*Require SSL* is enabled and *Accept* option is selected):



Since certificate authentication is used, ensures that Anonymous Authentication is enabled for this site:



5 - Verify that QTSP Signing Proxy can be accessed

To perform this test, you should:

- Install the client certificate (see step #3) in the user certificate store of the local machine that you will use to perform this test and continue the procedure.
- Declare the URL to access the server in your DNS.

Now it should be possible to access the SigningProxy using the link: *https://qtsp-api.<hostname>/swagger/index.html* (use the hostname defined in step #4).

> When navigating to the Swagger web page, if you don't get prompted with the dialog to select the client certificate, ensure that client certificate negotiation is enabled in IIS on the server.
>
> To verify this and enable this feature in IIS, please follow the instructions from this link: https://stackoverflow.com/questions/2518314/make-iis-require-ssl-client-certificate-during-initial-handshake

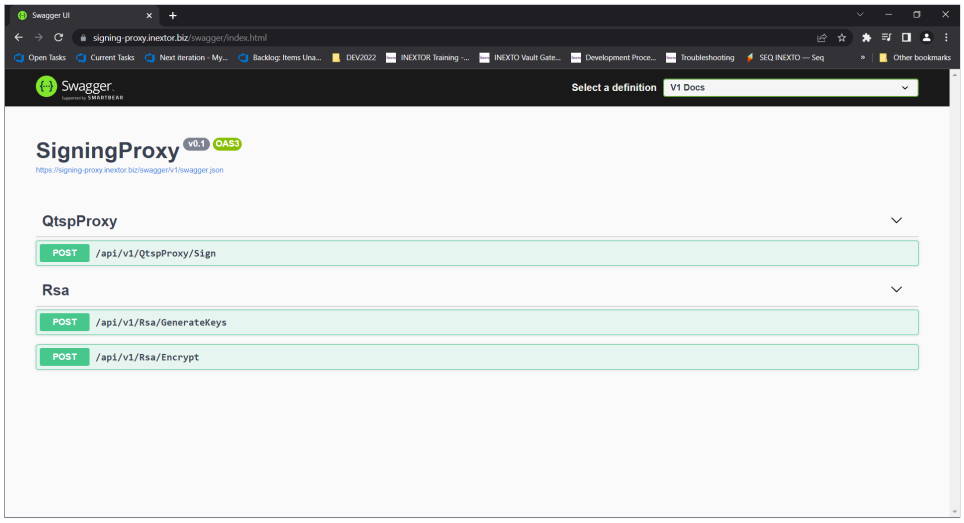


fig.4 Swagger interface of the SigningProxy

6 - Encrypt sensitive data

Generate the private and public keys by simply expanding the *GenerateKeys* POST method, clicking on "*Try it out!* and then "*Execute*";

Expand the *Encrypt* POST method, copy-paste the Intesi account Password in the clearText field, click on "*Execute*" and then copy-paste the resulting string in a temporary file.

The same must be done for the Intesi account PIN.

Now that both PIN & Password have been encrypted, open the appsettings.Production.json, paste the encrypted strings in the corresponsing fields (under *QtspSigningProxyConfiguration*) and set *EncryptSensitiveData* to *true*.

Save the file and close it.

Restart web app in IIS.

Here's an example of the *appsettings.Production.json* after encryption:

appsettings.Production.json

```
{
    "Serilog": {
        "WriteTo": [
            {
                "Name": "File",
                "Args": {
                    "path": "C:/Components/ApplicationQTSP/logs/QtspSigningProxy-.log",
                    "rollingInterval": "Month",
                    "outputTemplate": "[{Timestamp:yyyy-MM-dd HH:mm:ss.fff zzz} {CorrelationId} {Level:u3}] {Username} ({SourceContext}) {Message:lj
                }
            }
        ]
    },
    "QtspSigningProxyConfiguration": {
        "PKBoxUrl": "<https://your-pkbox-url-by-intesi>"
        "EncryptSensitiveData": true,
        "Username": "<your-username-by-intesi>",
```

```
        "Password": "fsIFAfooP4Z1FkoKAKyGYIyWpRZ8Rx/W5b6qHGbnNFbJv2HVZKLfxKtIcT4sZK6mIhvgOlwEFQXwIekFmOtiivjUFVhbekKdzHCVmKrNlDefLumCd4bmnfjY+DHmyqo(
        "Pin": "vQ5W45cyaNx809WhNg+UccpSwNJzuCBI7Qhrq5ExuaqLYEID17rOczwfwt3gGQr6FBaWwiJErzk3Lcyl/qkYw5/eIlLpJq9VKfmUMTMRH48bjptMiPUm+Iz3eFXUCyswBy2K/
        "Alias": "<your-alias-by-intesi>",
        "CertificateThumbprint": "<should-match-the-certificate-thumbprint>"
    }
}
```

7 - Test QTSP Signing Proxy installation

Reload the swagger page of the SigningProxy website in IIS.

Try signing a payload through the Swagger interface:

- Expand the Sign POST method
- The RequestTypes are the following:
    - 0 for *Economic Operators*
    - 1 for *Facilities*
    - 2 for *Machines*
    - 3 for *Code orders*
    - 4 for *Order Recalls*
    - 5 for ADM athentication Token

As an example you can set the requestType to 3 and use the payload provided here: order-clean-request.txt

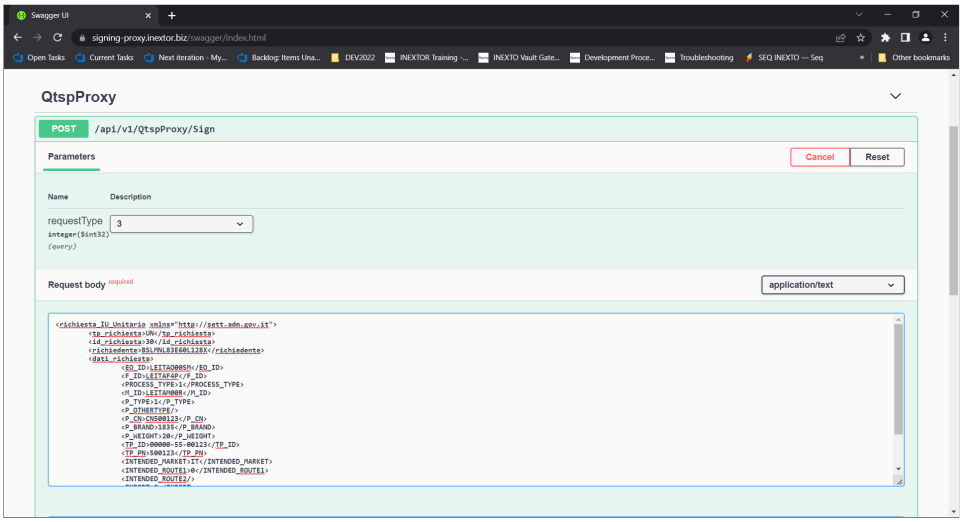


fig.5 Testing signing of a Code Order request.

If the content of the Request you are trying to sign respects the *XSD* for that *RequestType, you will receive the signed request in xml format the content response with a xml tag <Signature>.*

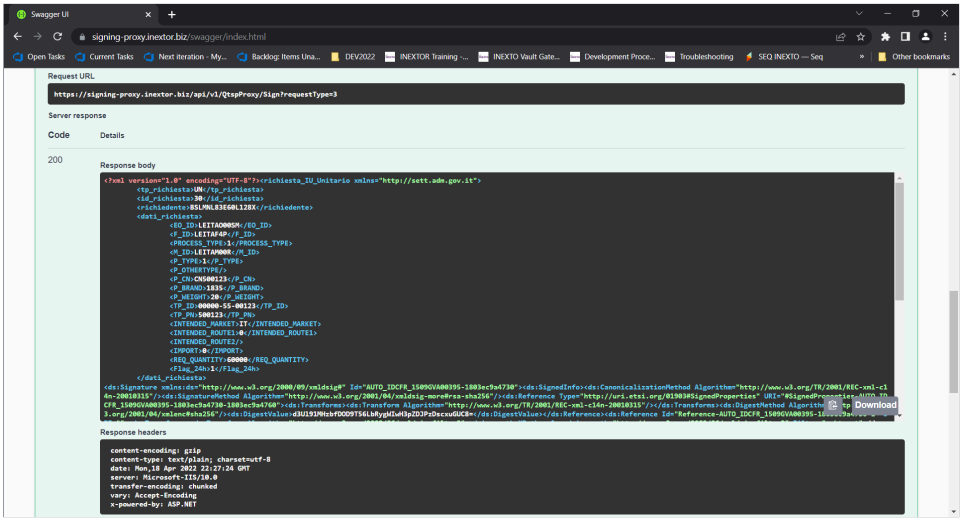
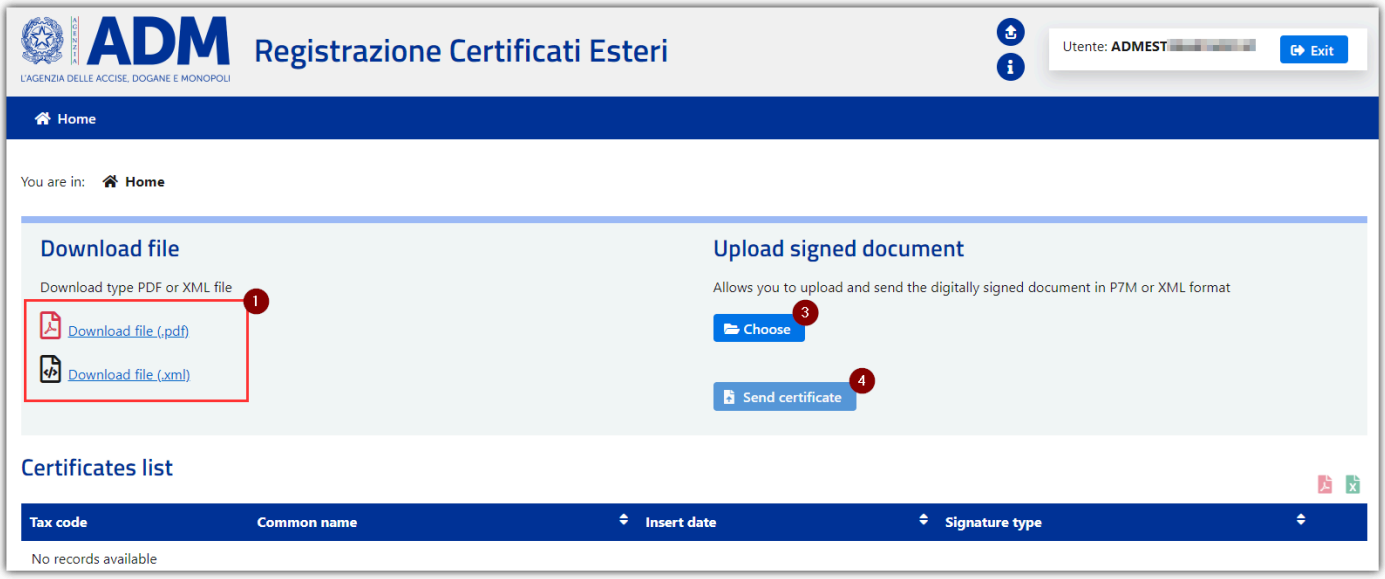


fig.6 Response with code 200 and the signed request in the Response Body.

8 - Make sure the signature was setup on ADM portal

Once the signing proxy is installed and ready to work, one last step needs to be performed in order for ADM/SOGEI to accept requests using your qualified signature. In fact, they need a signature reference which will be used to compare incoming requests' signature. This step is to be done from the ADM portal.

You don't necessarily need the Signing Proxy to perform this step if you have another way to access Intesi Group services to sign a xml of pdf document. You can use the another application to sign the token file, like for example ***IG Desk*** from Intesi Group.
Skip the procedure described in this step if you already did it or if you don't want to use Inexto Signing Proxy to do it.



1. Download a token file, either in xml format
2. You can use the Signing proxy swagger interface to sign the token **xml** file:



    1. select post /api/v1/QtspProxy/Sign, click "**Try it out**"
    2. set request type = 5
    3. paste the xml file content in the request body
3. Use the *Choose* button to select the signed document (result of step #2)
4. Use the *Send Certificate* button to upload the signed token to ADM's portal

> The token will expire if it's not re-uploaded quick enough. In this case ADM portal will display this error: 'Maximum elapsed time between download and upload has been exceeded'. You'll then need to start the procedure again from step 1.

9 - Vault configuration

In order to finalize the connection between the *SigningProxy* and the *Vault*, more actions are needed it in the *Edit view* of the *"AAMS Italy" Id Issuer* on the *Vault Administration* application.

If you don't have access to the Vault Administration, please provide the following information to Inexto once installation is done. You can then skip the final part of this guide.

- Client certificate with private key for authentication
- URL of your SigningProxy.

Go to the *Edit view* of the *"AAMS Italy" Id Issuer* on the *Vault Administration* application

- Add the *SigningProxy Url*
- Use the Import button to upload the authentication Certificate in *Proxy Authentication Certificate*
- Hit *Save* to validate

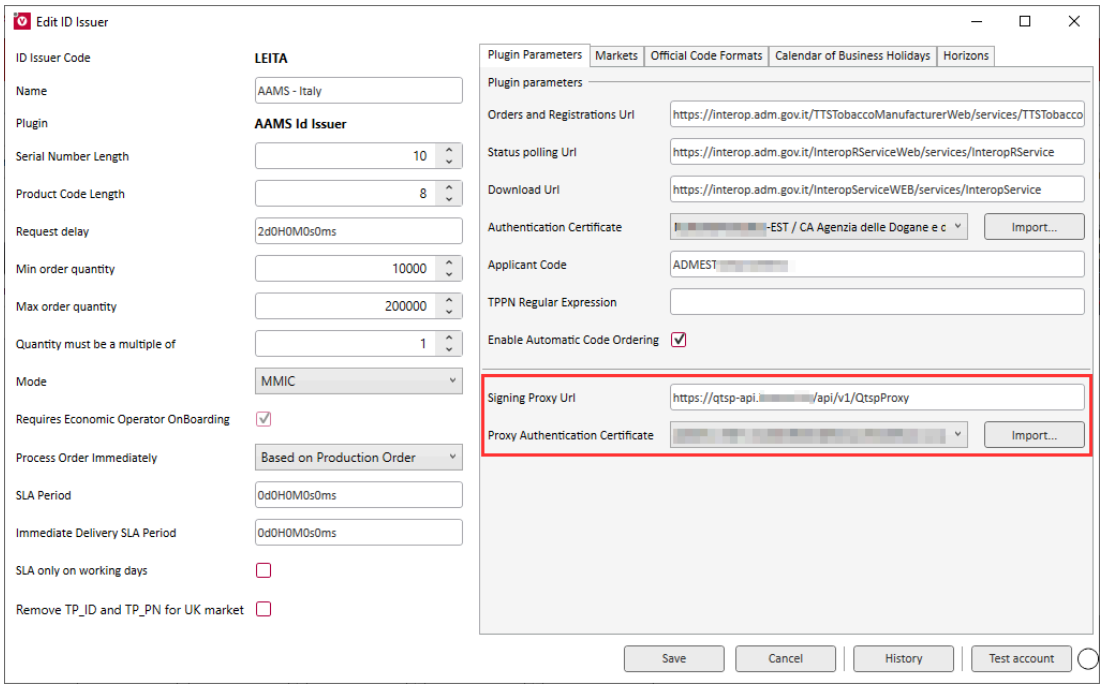


fig.7 AAMS Id Issuer configuration on the Vault Admin.

Vault Integration Development: Installing the QTSP Signing Proxy - Inexto managed

Introduction

> This page describes the installation of the QTSP Signing Proxy directly in a Vault server managed by Inexto. If you are not an Inexto agent, please rather refer to Installing the QTSP Signing Proxy - Manufacturer Managed.

The QTSP Signing Proxy allows requests to be signed by an Italian Qualified Trust Service Provider (INTESI) before being sent to the Italian ID Issuer (ADM). By "requests" we mean those related to the registration/update/deletion of Economic Operators, Facilities & Machines, and to Code Ordering/Order cancelling.

In order to be signed, a request has to pass through the Proxy where their signing credentials are stored in encrypted form. In this case, the proxy is hosted by Inexto and works alongside the Vault. You can see as an extension of the latter.

Prerequisites

This guide is a detailed explanation of one of the steps to be taken in the AAMS plugin setup", you'll need both to complete the installation.

- The Vault client certificate *Thumbprint* (needed at step #3). It is also the one you'll need to set in the Vault administration for the Proxy authentication.
- A hostname should be defined to access the server remotely. This hostname should be declared in the DNS.
- The *ApplicationQtsp* solution (your can download it from here → ApplicationQTSP.zip).

The customer should have received the following parameters (to be added in the *appsettings.Production.json*, see step #3). These pieces of information come from Intesi Group after a contract has been established with the manufacturer.

- Username of Intesi account;
- Password of Intesi account;
- Alias for the PkBox;
- PIN for the PkBox;
- Url of the PkBox.

These elements will be used later (see step #3 "Setup folder & app settings" and step #6 "Encrypt sensitive data").

Installation procedure

To install and have the SigningProxy correctly configured for production the following steps, each with its own section, need to be performed:

1. First IIS installation/configuration;
2. IIS Application Pool;
3. Setup project folder & app settings;
4. Create QTSP Web Application on IIS;
5. Verify that the QTSP Signing Proxy can be accessed;
6. Encrypt sensitive data;
7. Test QTSP Signing Proxy
8. Provide information (URL & Authentication Certificate)

> If there are several instances of the Vault running in several nodes, you will need to install the proxy in each node. This means that you must **run only one proxy at a time** while you go through the installation process. You can enable the proxies on all the nodes in parallel once they are all correctly installed and configured.

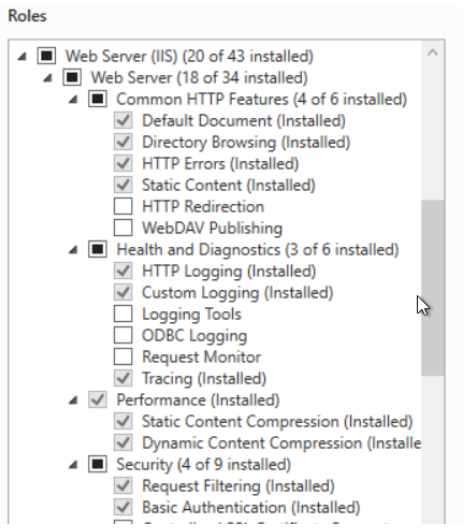
1 - First IIS configuration/installation

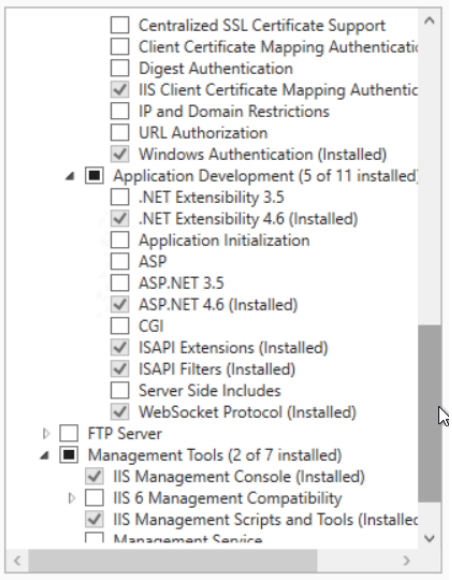
IIS Manager should already be installed and enabled on your server, if it's not the case follow this Microsoft guide:

https://docs.microsoft.com/en-us/dynamics-nav/how-to--install-and-configure-internet-information-services-for-microsoft-dynamics-nav-web-client#WS2012

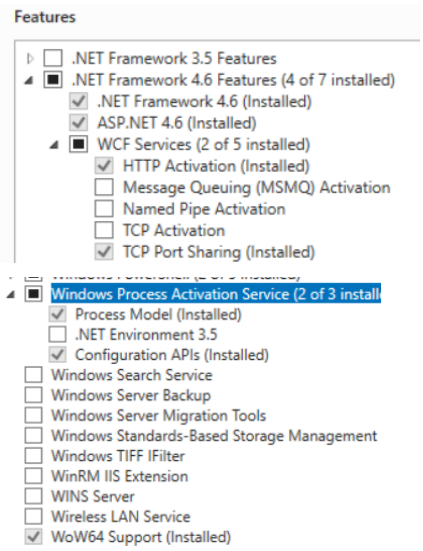
> The version of the .NET Framework and ASP.NET in the screenshots below reference 4.6, but depending on your version of Windows Server, the version can be higher (e.g. 4.8).

Open the Server Manager and enable the following Roles on the server:





The following features should be enabled on the server:



2 - IIS Application Pool

Open *IIS Manager* and under Connections right click on Application Pools, then select "*Add Application Pool...*".

You need to have admin permissions to see the configuration (ask access in Powerapps).

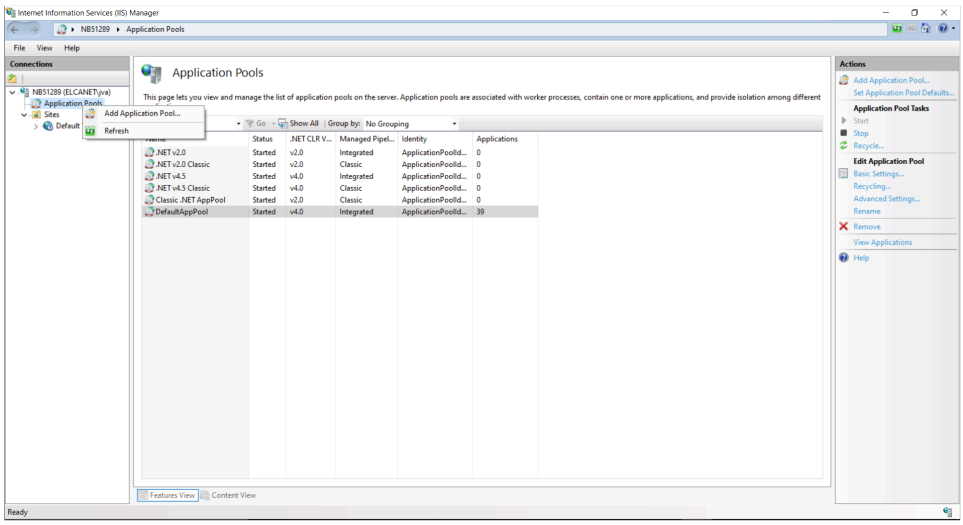


fig.1 Adding Application Pool

Insert the name of your application pool (ex. QtspSigningProxyAppPool), set *.NET CLR Version* to "*No Managed code*", leave managed pipeline mode to Integrated and click on OK.

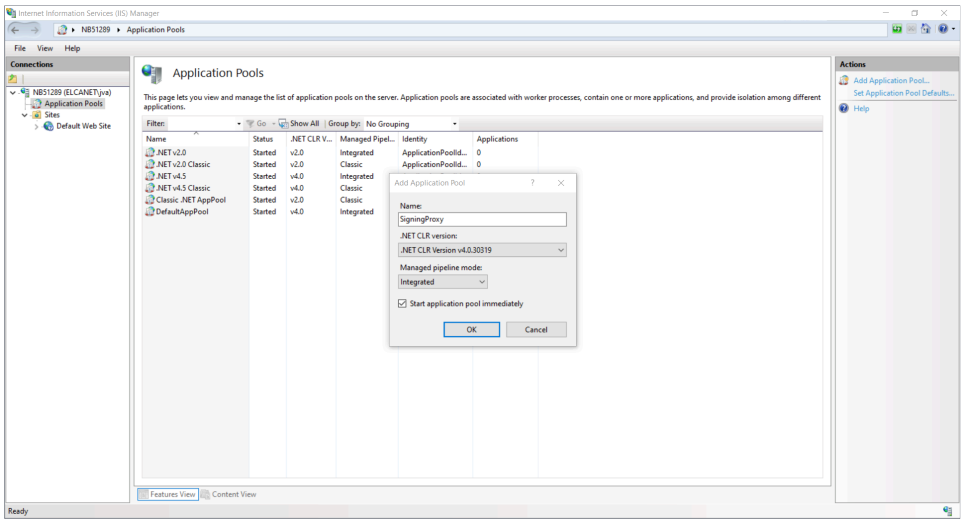


fig.2 Specifying Application Pool name and managed preference.

Now right click on the application pool, select advanced settings, scroll down to ***Process Model*** and set *Load User Profile* to *True*.

3 - Setup project folder & app settings

Extract the compiled project and place it in a sub-folder on your main disk (ex. C:\Components).

Open *appsettings.Production.json* then:

- ensure that *EncryptSensitiveData* is set to false;
- Vault authenticate to the QTSP signing proxy using certificate authentication. Set CertificateThumbprint to the thumbprint of the certificate you will provide *Inexto* to communicate with the Proxy (this requires having a Trusted Root Certificate you can share already installed or installing one);
- fill Username & Alias with the values received by Intesi. These values are used to connect to Intesi system to apply the digital signature to messages sent to the Italian ID Issuer.
- update the PKBoxUrl with the url Intesi provided to the customer.
- update the path to the log files to a location of your preference (ex. C:/Logs/QtspSigningProxy-.log).

Save your changes and close the file.

4 - Create QTSP Web Application on IIS

Launch Internet Information Service.

In *Connections*, under the server root right click on *Sites* and select *"Add Website..".*

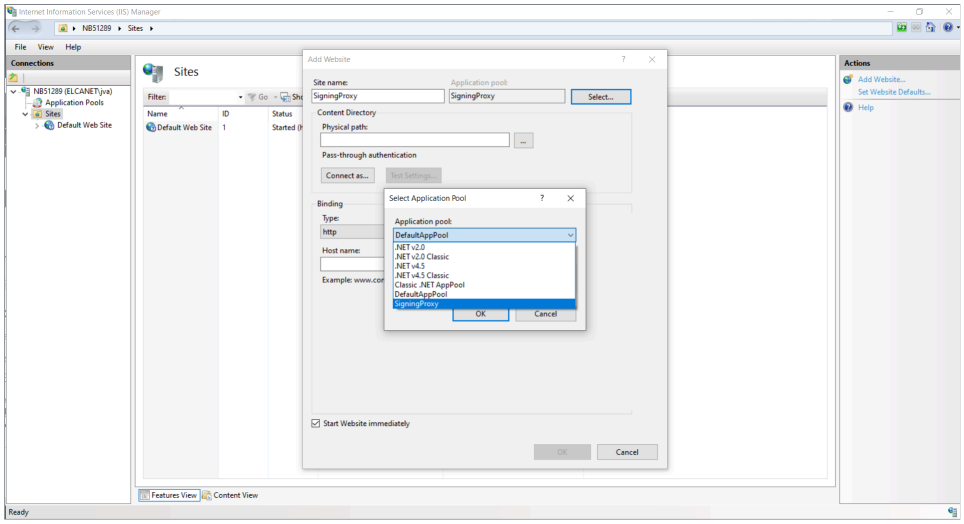


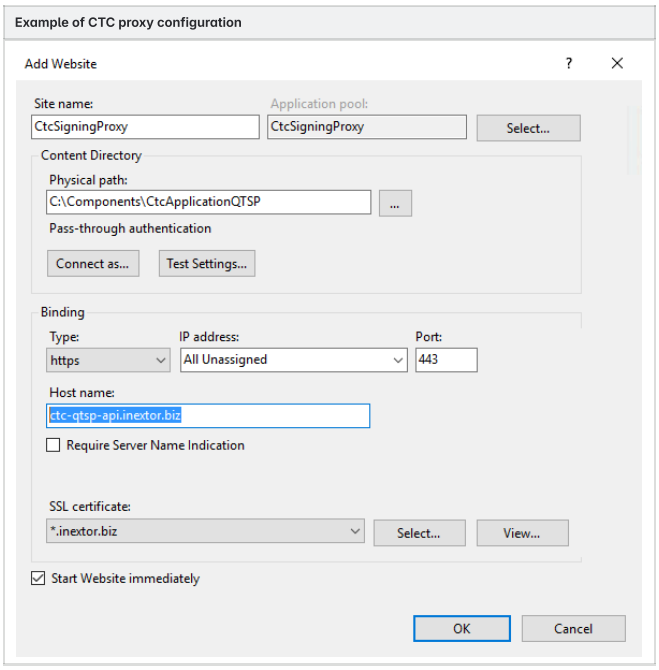
fig.3 Selecting the Application Pool defined earlier.

Define the site name (ex. *SigningProxy*), select the application pool defined in step #2.

Select the folder where you placed the SigningProxy files in step #3.

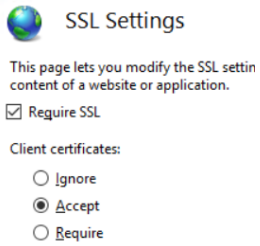
Set the binding type to *https*, then choose a hostname (ex. *signing-proxy.<company-name>.com*) then finally click on OK. **The hostname should be declared in the DNS**, ask the IT team to do that if not already done.

Example of CTC proxy configuration

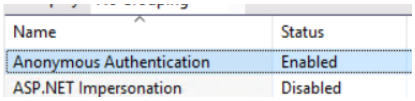


Give rights to "IIS AppPool\<Name of the AppPool>" (e.g. IIS AppPool\SigningProxyPool): right click on the site you just created, select edit permissions, click on the *Security* tab then "*Edit...*", then "*Add*" and give all permissions to the app pool account (location is the current server, the object name is "*IIS AppPool\QtspSigningProxyAppPool*", hit "*Check Names*" to ensure that it's recognized).

Select the site you just created and Edit the SSL settings according to the screenshot below (*Require SSL* is enabled and *Accept* option is selected):



Since certificate authentication is used, ensures that Anonymous Authentication is enabled for this site:



5 - Verify that QTSP Signing Proxy can be accessed

To perform this test, you should:

- Install the client certificate (see step #3) in the user certificate store of the local machine that you will use to perform this test and continue the procedure.
- Declare the URL to access the server in your DNS.

Now it should be possible to access the SigningProxy using the link: *https://<company-name>-qtsp-api.inextor.biz/swagger/index.html* (use the hostname defined in step #4).

When navigating to the Swagger web page, if you don't get prompted with the dialog to select the client certificate, ensure that client certificate negotiation is enabled in IIS on the server.

To verify this and enable this feature in IIS, please follow the instructions from this link: https://stackoverflow.com/questions/2518314/make-iis-require-ssl-client-certificate-during-initial-handshake

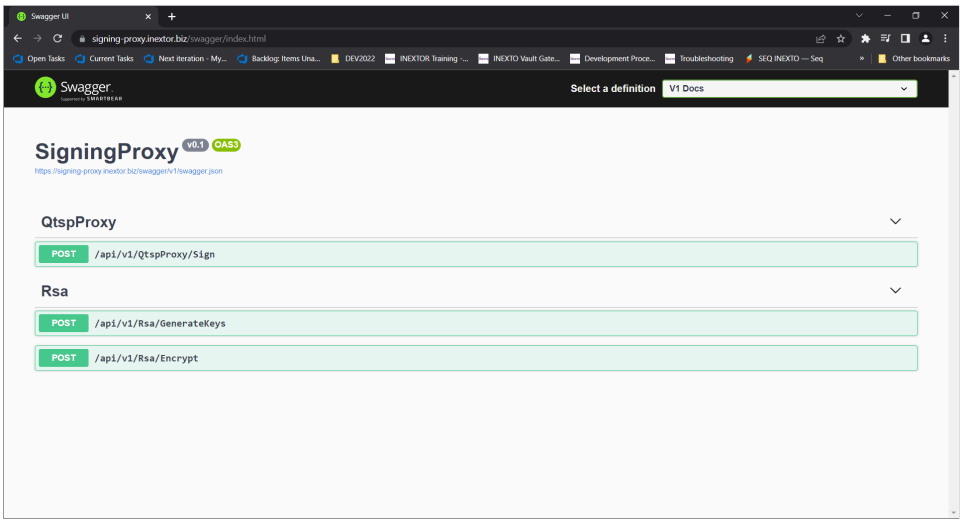


fig.4 Swagger interface of the SigningProxy

6 - Encrypt sensitive data

Generate the private and public keys by simply expanding the *GenerateKeys* POST method, clicking on "*Try it out!* and then "*Execute*";

Expand the *Encrypt* POST method, copy-paste the Intesi account Password in the clearText field, click on "*Execute*" and then copy-paste the resulting string in a temporary file.

The same must be done for the Intesi account PIN.

Now that both PIN & Password have been encrypted, open the appsettings.Production.json, paste the encrypted strings in the corresponsing fields (under *QtspSigningProxyConfiguration*) and set *EncryptSensitiveData* to *true.*

Save the file and close it.

Restart web app in IIS.

Here's an example of the *appsettings.Production.json* after encryption:

appsettings.Production.json

```
{
    "Serilog": {
        "WriteTo": [
            {
                "Name": "File",
                "Args": {
                    "path": "C:/Components/ApplicationQTSP/logs/QtspSigningProxy-.log",
                    "rollingInterval": "Month",
                    "outputTemplate": "[{Timestamp:yyyy-MM-dd HH:mm:ss.fff zzz} {CorrelationId} {Level:u3}] {Username} ({SourceContext}) {Message:lj
                }
            }
        ]
    },
    "QtspSigningProxyConfiguration": {
        "PKBoxUrl": "<https://your-pkbox-url-by-intesi>"
        "EncryptSensitiveData": true,
        "Username": "<your-username-by-intesi>",
        "Password": "fsIFAfooP4Z1FkoKAKyGYIyWpRZ8Rx/W5b6qHGbnNFbJv2HVZKLfxKtIcT4sZK6mIhvgOlwEFQXwIekFmOtiivjUFVhbekKdzHCVmKrNlDefLumCd4bmnfjY+DHmyqo
        "Pin": "vQ5W45cyaNx809WhNg+UccpSwNJzuCBI7Qhrq5ExuaqLYEID17rOczwfwt3gGQr6FBaWwiJErzk3Lcyl/qkYw5/eIlLpJq9VKfmUMTMRH48bjptMiPUm+Iz3eFXUCyswBy2K/
        "Alias": "<your-alias-by-intesi>",
        "CertificateThumbprint": "<should-match-the-certificate-thumbprint>"
    }
}
```

7 - Test QTSP Signing Proxy installation

Reload the swagger page of the SigningProxy website in IIS.

Try signing a payload through the Swagger interface:

- Expand the Sign POST method
- The RequestTypes are the following:
    - 0 for *Economic Operators*
    - 1 for *Facilities*
    - 2 for *Machines*
    - 3 for *Code orders*
    - 4 for *Order Recalls*
    - 5 for ADM athentication Token

As an example you can set the requestType to 3 and use the payload provided here: order-clean-request.txt

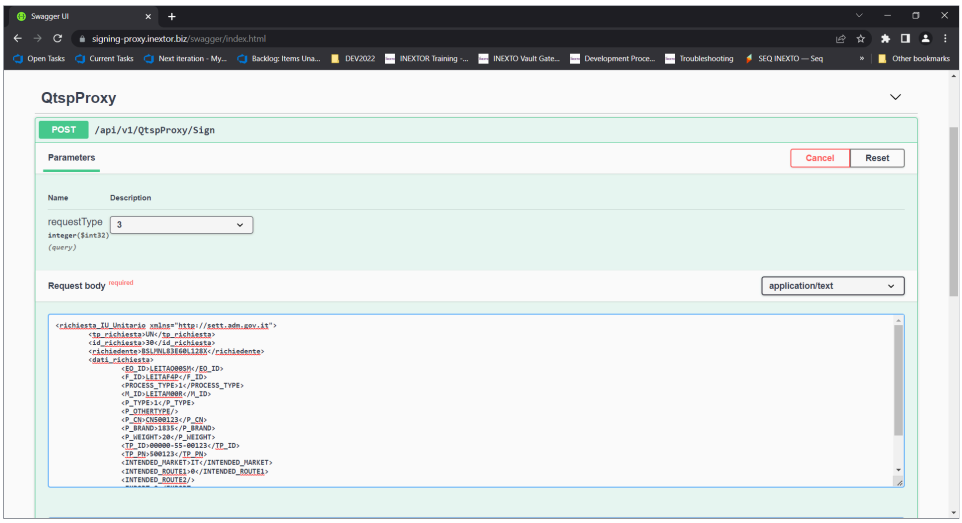


fig.5 Testing signing of a Code Order request.

If the content of the Request you are trying to sign respects the *XSD* for that *RequestType, you will receive the signed request in xml format the content response with a xml tag <Signature>.*

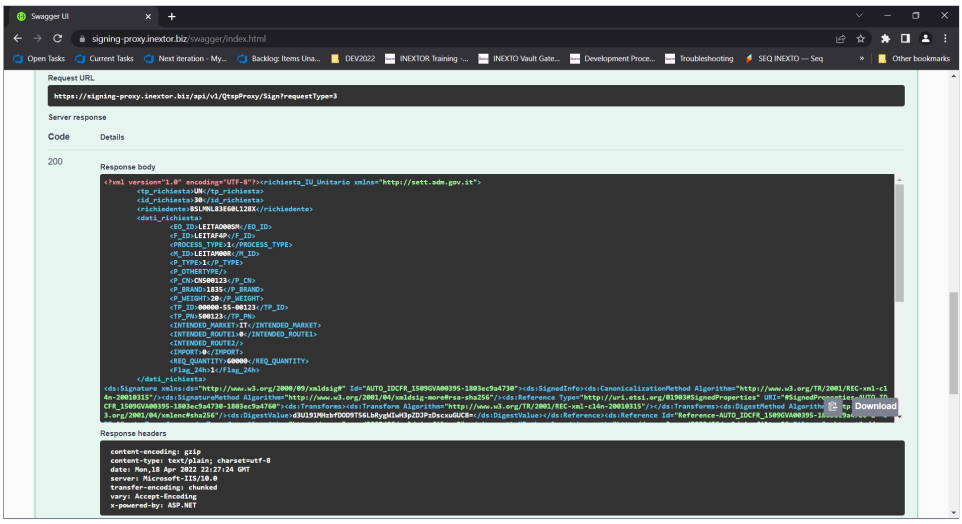
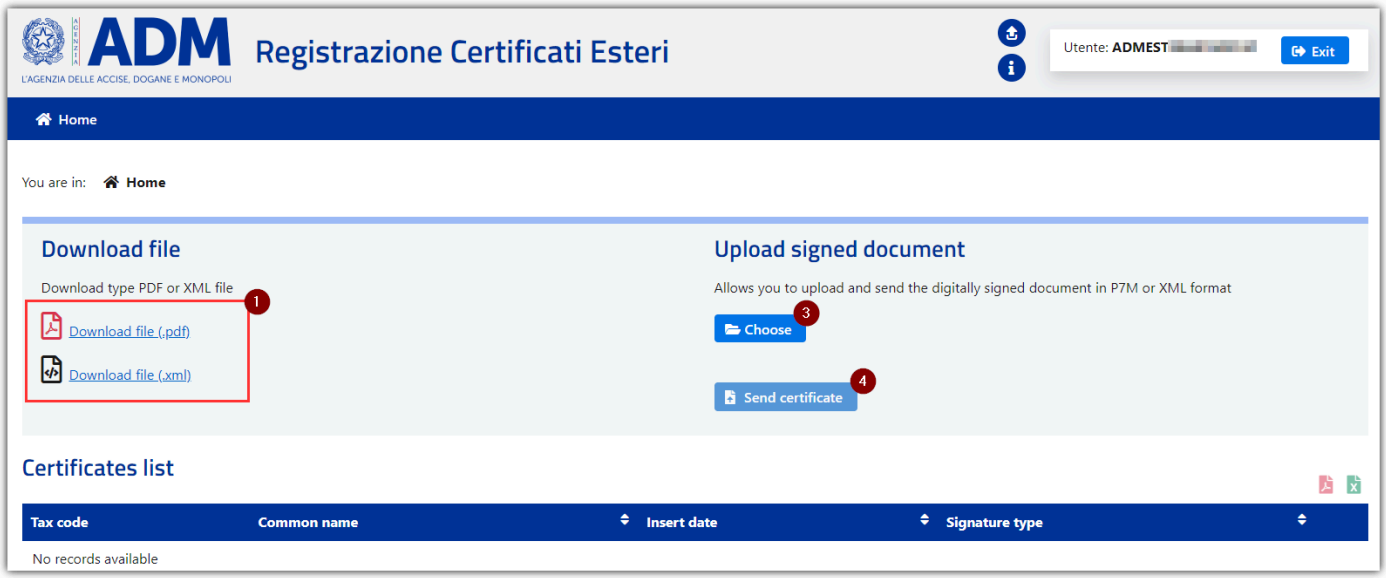


fig.6 Response with code 200 and the signed request in the Response Body.

8 - Make sure the signature was setup on ADM portal

Once the signing proxy is installed and ready to work, one last step needs to be performed in order for ADM/SOGEI to accept requests using the manufacturer's qualified signature. In fact, they need a signature reference which will be used to compare incoming requests' signature. This step is to be done from the ADM portal, so you'll need to do it together with the manufacturer (you don't have access to ADM portal).

This step is not dependent on the signing proxy installation and may already have been performed by the customer using other signature procedures (like Intesi Group ***IG Desk*** application), make sure that it was not already done before continuing. Else this step can be skipped.



1. Ask the manufacturer to download the token file in xml format

2. Use the Signing proxy swagger interface to sign the token xml file:



1. select post /api/v1/QtspProxy/Sign, click on "**Try it out**"
2. set request type = 5
3. paste the xml file content in the request body

3. Use the *Choose* button to select the signed document (result of step #2)
4. Use the *Send Certificate* button to upload the signed token to ADM's portal

> ⓘ The token will expire if it's not re-uploaded quick enough. In this case ADM portal will display this error: 'Maximum elapsed time between download and upload has been exceeded'. You'll then need to start the procedure again from step 1.

9 - Vault configuration

In order to finalize the connection between the SigningProxy and the Vault, more actions are needed in the *Id Issuer Configuration* of the Vault .

- Add the *SigningProxy Url*
- Select the Vault client certificate in the combobox for *Proxy Authentication Certificate*

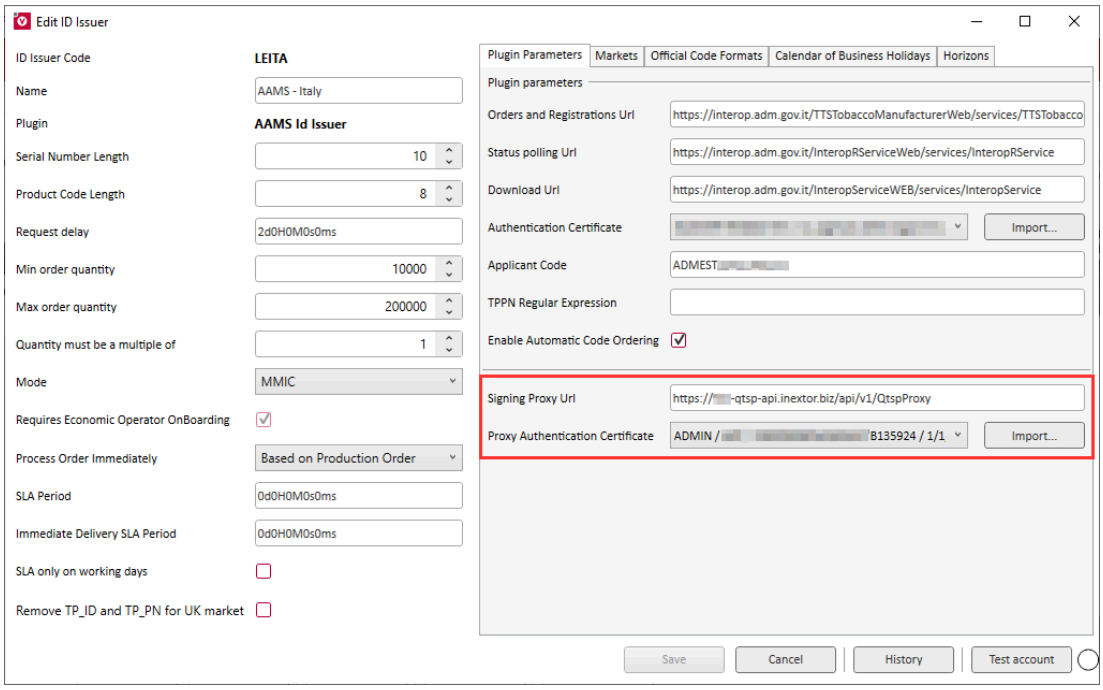


fig.7 AAMS Id Issuer configuration on the Vault Admin.

Vault Integration Development: Decompose codes with Ait Service (Vault)

Ait Service

The Ait Service is accessible on both the Gate and the Vault. Tt is composed of the following different components:

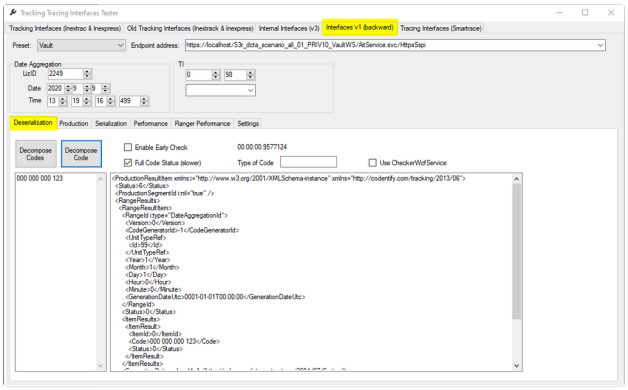
1. Deserialization Service: This part is used to recover information about codes. (code decomposition)
2. Serialization Service: This part is used to generate the codes corresponding to a production (code rebuilding)
3. Production Service: This part is used to recover information about a production.

On this page, we will focus on the Deserialization service.

Creating a Request

Using the Interface service tester

The Interface service tester can be used to query directly the Gate or Vault. It can be used to decompose one or several codes. The different parameters are explained below. The checkbox *Use CheckerWcfService* should be checked only if the user wants to decompose codes using the WcfCheckerService instead of the AitService.



Using Postman

If you prefer to use Postman, here are some specificities. The header of the request must contain:

Key	Value
Content-Type	application/soap+xml
SOAPAction	DecomposeCodes

And below an example of the body:

AitService api request

```
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope" xmlns:a="http://www.w3.org/2005/08/addressing">
<s:Header>
<a:Action s:mustUnderstand="1">DecomposeCode</a:Action>
```

```
<a:MessageID>urn:uuid:3c38b569-3963-40fa-b871-cd8c60ac7ed3</a:MessageID>
<a:ReplyTo>
<a:Address>http://www.w3.org/2005/08/addressing/anonymous</a:Address>
</a:ReplyTo>
<a:To s:mustUnderstand="1">https://staging-vault-api.inextor.biz/AITService.svc/HttpsCertificate</a:To>
</s:Header>
<s:Body>
<DecomposeCode xmlns="http://codentify.com/tracking/2013/06">
<supi>0028CMWLN49T</supi>
<decomposeQueryOptions xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
<EarlyCheck>true</EarlyCheck>
<GetFullCodeStatus>true</GetFullCodeStatus>
<TypeOfCode></TypeOfCode>
<CountHistoryRecords>false</CountHistoryRecords>
<FlagCode>false</FlagCode>
<CityId>0</CityId>
<MaxSupiDisplayed>0</MaxSupiDisplayed>
<Requestor>InterfacesClientTester</Requestor>
</decomposeQueryOptions>
</DecomposeCode>
</s:Body>
</s:Envelope>
```

Parameters

Several parameters can be used in the query. Note that in the InterfacesServiceTester, not all the parameters can be set.

Parameter	Type	Description
EarlyCheck	boolean	Gather information about the code even if the Checker does not have production information yet.
GetFullCodeStatus	boolean	Used to retrieve the code status (Only on Checker)
CountHistoryRecords	boolean	Set to true to get some information about the code check history
FlagCode	boolean	Use to mark the code as Used
CityId	integer	specify the geonameId of the location where the product was found
MaxSupiDisplayed	integer	specify the number of historical record returned (ignored if CountHistoryRecords = false)
Requestor	string	Name of the person who is making the request
TypeOfCode	string	Specify the type of code the requestor wants to decompose
UseCheckerWcfService	boolean	Use only if you want to use the WcfCheckerService instead of the AitService

Response

The response typically looks like this:

Ait service response

```
<ProductionResultItem xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://codentify.com/tracking/2013/06">
  <Status>131072</Status>
  <ProductionSegmentId>
    <SegmentStartDateUtc>2020-05-11T12:32:56.56Z</SegmentStartDateUtc>
    <CodeGeneratorId>2135</CodeGeneratorId>
    <FactoryInfo>
      <MID>01</MID>
      <MCID>OX</MCID>
      <CheckerFactoryIds>
        <CheckerFactoryId>
          <CheckerCode>PMI</CheckerCode>
          <McgCode>MCG_OX</McgCode>
        </CheckerFactoryId>
      </CheckerFactoryIds>
    </FactoryInfo>
  </ProductionSegmentId>
  <RangeResults>
    <RangeResultItem>
      <RangeId i:type="AggregationId">
        <Version>1</Version>
        <CodeGeneratorId>0</CodeGeneratorId>
        <UnitTypeRef>
          <Id>13</Id>
        </UnitTypeRef>
        <Year>2002</Year>
        <Month>50</Month>
        <Day>0</Day>
        <Hour>0</Hour>
        <Minute>249</Minute>
        <AggregationId>10213500000000000249</AggregationId>
      </RangeId>
      <Status>1</Status>
      <ItemResults>
        <ItemResult>
          <ItemId>488</ItemId>
          <Code>TYF0 BUV MHP 1KHQ</Code>
          <Status>16</Status>
          <ItemProperties>
            <KeyValueOfstringstring xmlns="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
              <Key>Prefix</Key>
              <Value>13511073762115584</Value>
            </KeyValueOfstringstring>
            <KeyValueOfstringstring xmlns="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
              <Key>NgpDuns</Key>
              <Value>0003000400020000001E8</Value>
            </KeyValueOfstringstring>
            <KeyValueOfstringstring xmlns="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
              <Key>NgpPlatform</Key>
              <Value>3</Value>
            </KeyValueOfstringstring>
            <KeyValueOfstringstring xmlns="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
              <Key>NgpProduct</Key>
              <Value>4</Value>
            </KeyValueOfstringstring>
            <KeyValueOfstringstring xmlns="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
              <Key>NgpSite</Key>
              <Value>2</Value>
```

```
            </KeyValueOfstringstring>
            <KeyValueOfstringstring xmlns="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
              <Key>NgpDevice</Key>
              <Value>488</Value>
            </KeyValueOfstringstring>
            <KeyValueOfstringstring xmlns="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
              <Key>ProductionDateTime</Key>
              <Value />
            </KeyValueOfstringstring>
          </ItemProperties>
        </ItemResult>
      </ItemResults>
      <GenerationDate xmlns:d4p1="http://schemas.datacontract.org/2004/07/System">
        <d4p1:DateTime>2020-05-11T12:32:56.56Z</d4p1:DateTime>
        <d4p1:OffsetMinutes>120</d4p1:OffsetMinutes>
      </GenerationDate>
      <ProductionCode i:nil="true" />
    </RangeResultItem>
  </RangeResults>
</ProductionResultItem>
```

Status Analysis

If most of the response can be intuitive, the different statuses are spread across the response. Here is an explanation of how to interpret them.

ProductionResultItemStatus

This status is present at *//ProductionResultItem/Status* and represents the status of the production to which the code is associated.

It can have the following values:

Status	Value	Description
Validated	0	automatically approved by the system
Authorized	1	manually approved by an administrator
Rejected	2	manually rejected by an administrator
Destroyed	4	marked as destroyed
NoNoise	5	secret keys are not available on this Checker
Invalid	6	refused for automatic approval. Must be reviewed by administrator
Unknown	7	state is not known
GovMask	15	state is not known
InvalidSignature	16	data has been tampered since it was signed by author CoG or MCG
Outdated	32	data synchronization happened too late to comply with the Checker's policy
UpdateTerminated	64	data was synchronized to the Checker after the production is marked as terminated

UnauthorizedStamp	128	production has manufactured products without sufficient e-stamps
UnauthorizedSKU	256	SKU was not authorized
UnauthorizedMCG	512	MCG was not authorized
UnauthorizedCOG	1024	COG was not authorized
UnauthorizedMCGOwner	2048	Manufacturer or importer was not authorized
UnauthorizedSKUOwner	4096	Legal entity was not authorized
UnauthorizedTaxLevel	8192	Tax level was not authorized
UnauthorizedMSK	16384	MSK was not authorized
NoMSK	32768	MSK is not available on this Checker
ExpiredMSK	65536	MSK is expired
ErrorMask	12713968	This mask is the combination of the system flags
ExpiringMSK	131072	MSK is about to expire
NegativeQty	262144	Erroneous negative quantity was reported
DecreasingQtyOverTime	524288	Erroneous decreasing quantity was reported
DeltaQtyExceeded	1048576	Erroneous quantity increment was reported
PrintedExceedsGenerated	2097152	Inconsistent printed and generated quantities were reported
InvalidReceipt	4194304	Invalid signature for receipt
SKUAuthorizationPending	8388608	Waiting for approval of SKU

RangeResultItemStatus

This status is present under *//RangeResultItem/Status* and corresponds to the status of the range in which the code belongs.

Possible values are:

Status	Value	Description
None	0	The range is empty and does not actually contain any product item
Normal	1	The range contains some items. This is the default case.

Extended	2	The range contains items manufactured after the EndDate of the production. (Can happen with the EarlyCheck option)

ItemResultStatus

This status is present under *//ItemResult/Status* and corresponds to the status of the code itself. **Note that for this information to be useful, the GetFullCodeStatus needs to be true**.

As a code can have a combination of different statuses, these one are flags which can be combined.

Status	Value	Description
FirstVerification	0	This is the first time that the code is checked
Used	1	The code is already used.
Counterfeited	2	The code is related with the product that has been counterfeited.
Waste	4	The code is wasted.
~~UnauthorizedCheck~~	~~8~~	~~The user is not authorized for this verification~~ Depricated
NotAvailable	16	Code authentication is not available. That means the system has no way to verify the Noise of the code, either because the production is marked as NoNoise or the MSK is owned by a different system.

General installation guide: Windows administrators

- Windows Administrators: Vault Proxy
- Windows Administrators: Certificate Management
- Windows Administrators: Special Configurations for Windows
- Windows Administrators: Master Data Services
- Windows Administrators: Gate
- Windows Administrators: Vault
- Windows Administrators: Liz

Windows Administrators: Vault Proxy

Introduction

A Vault Proxy is a server software currently used for the RRP market. The purpose of the Vault Proxy is to provide manufacturers the ability to verify printed codes on RPP devices locally (without an internet connection)

This method solves the issue when the Gate is unable to connect with the Vault due to network/connection issues.

This is achieved by storing all printed codes locally on a Proxy server to allow local verification.

The following sections contain all necessary information for the setup and configuration of a new Vault Proxy as well as instructions to configure MCG/Gate communicates with Vault via the Vault Proxy.

Vault Proxy: Prerequisites

This section details the technical prerequisites that must be in place prior to installation.

New Installation Steps for Vault Proxy Setup

This sections gives step-by-step detailed instructions for installing the Vault Proxy in standard configurations.

/wiki/spaces/INEXTOR/pages/395828160

This section tells you how to integrate the Vault Proxy to a Test Scenario.

/wiki/spaces/INEXTOR/pages/395828162

This section is a checklist that you need to review when you get any issue with the Vault Proxy setup.

Vault Proxy: Prerequisites

Preinstalled Software

The following items must be available on the server where the Vault Proxy will be deployed on:

- Microsoft .NET Framework 4.5
- IIS

Database

A database server with SQL 2012 at least must be available and accessible from the service. Refer to Database Setup - SQL Server Installation for help for installing SQL Server.

Network

- Internet Connection for Vault

Certificates and Licenses

Required Certificates:

Certificate Name	Type	Issuer
Proxy Root Certificate (public key)	"*.cer"-file	Trusted CA or Self-signed
Vault Proxy Certificate (private key)	"*.pfx"-file	Proxy Root Certificate
Inexto Client Certificate (public key)	"*.cer"-file	Vault Proxy Certificate
Vault Root Certificate (public key)	"*.cer"-file	(from Vault)
Vault Technical Provider Certificate (public key)	"*.cer"-file	Vault Root Certificate (from Vault)
Vault Certificate (public key)	"*.cer"-file	Vault Technical Provider Certificate (from Vault)
Vault Client Certificate (private key)	"*.pfx"-file	Vault Certificate (from Vault)

Note: Only authorized personnel are allowed to create the certificates.

Required Firewall Settings

Make the following ports exceptions for the firewall for incoming traffic:

Server	Name	Port	Type
Proxy Server	Internet	443	TCP
Data Server	SQLServer	1433	TCP

Installation Media

The following items are provided in the installation media. The items in **bold** are necessary for every installation. The items in *italic* correspond to installation options that may not be necessary in every case.

Item	Type	Description
InextoClientWS	Folder	Vault Proxy web service folder to be copied on application server.
VaultProxyDatabase.dacpac	DAC Package	DAC Package to be deployed in order to create the Vault Proxy database.
Proxy Root Certificate	*.cer file	Proxy Root Certificate (trusted authority - public key)
Vault Proxy Certificate	*.pfx file	Vault Proxy Certificate (with private key) is used to issue Inexto Client Certificate for each factory

Item	Type	Description
Inexto Client Certificate	*.cer file	The certificate is used to authenticate the request from Gate and look up the mapped Vault Client Certifciate (e.g: Snedai Client certificate)
Vault Root Certificate	*.cer file	Vault Root Certificate (trusted authority - public key)
Vault Technical Provider Certificate	*.cer file	Vault Technical Provider Certificate (public key)
Vault Certificate	*.cer file	Vault Certificate (public key) used to issue Vault Client Certificate
Vault Client Certificate	*.pfx file	The certificate is issued by Vault and included to each request sent to Vault (e.g Snedai Client certificate)
RootCertificateManager.exe	Executable	Only required if Root and TP Certificates (or TP private key) are not provided
CertificateManager.exe	Executable	Tool to import the certificates into the Vault Proxy database
S3rCertGenerator.exe	Executable	Only required if not using a CA (e.g. Verisign) or another tool to issue certificate for Factory

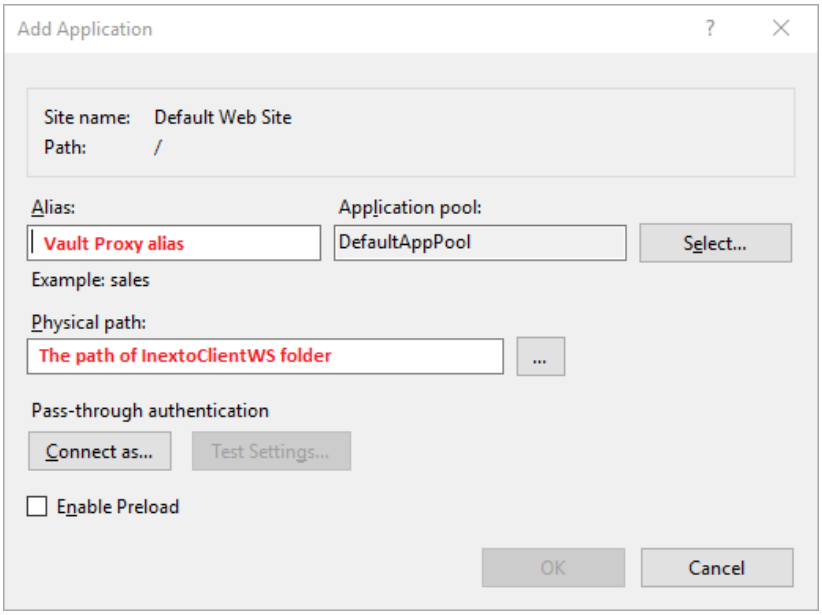
Vault Proxy: Web Application

Introduction

InextoClientWS is the main web application of Vault Proxy. It is responsible for receiving the requests sent from Factories and forward them to Vault with mapped certificate.

Deploy InextoClientWS to IIS

- Open "Control Panel", "Administrative Tools" and "Internet Information Services".
- Right click on the Web directory.
- Click "Add Application..."
- Input necessary information to deploy InextoClientWS to IIS



- The InextoClientWS is added to the selected website.

Vault Proxy: IIS Configuration

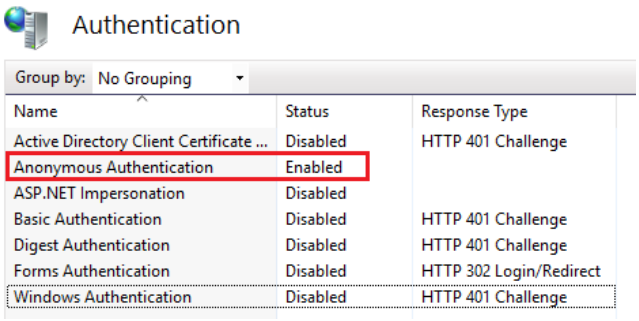
Enable SSL

Follow the link Enable SSL with Vault to enable SSL binding with **Proxy Root Certificate** on the Vault Proxy server.

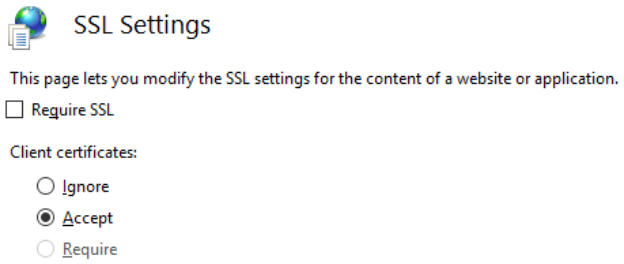
By this way, the proxy will be able to retrieve the certificate sent by Factory.

InextoClientWS configuration

- Open "Control Panel", "Administrative Tools" and "Internet Information Services".
- Expand the Web directory and select InextoClientWS node
- From the center menu, select Authentication node
- Enable Anonymous Authentication and disable other authentication modes



- Back to the center menu, select SSL Settings node
- Configure it as the following image



- Click Apply button on the left panel to save changes.
- Browse the InextoClientWS and make sure both FisWsV5.svc and MCGatewayWS.asmx are hosted properly.
- Both services should contain at least one certificate endpoint.

> Note: The address of certificate endpoint will be used at the further step when we configure Gate or MCG.

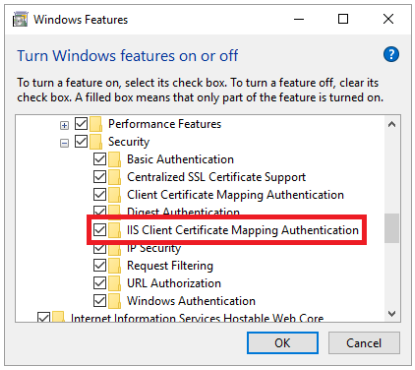
Vault Proxy: Inextor Integration

This page provides you a step by step instruction to integrate the Vault Proxy to Inextor solution.

Prerequisites

The following points are required to start the Vault Proxy integration:

1. Inextor solution was deployed properly. e.g: Vault, Gate, MCG...
2. The IISClientCertificateMapping feature was enabled on the server where we deployed the GateManagerWS to.



Setup the connection between Vault and Gate/MCG

If Gate/MCG is already configured to connect Vault properly, please ensure the Vault connection was configured with SSPI authentication and skip this step.

1. Create a Vault connection request from Gate/MCG Administration
2. In Vault website, create a factory connection with SSPI authentication by importing the Vault connection request.
3. In Gate/MCG Administration, import the certificate file issued by Vault
4. Correct the IP address of Vault connection and make sure the Vault connection works properly before moving to the next step.

Configure Gate/MCG to connect to Vault Proxy

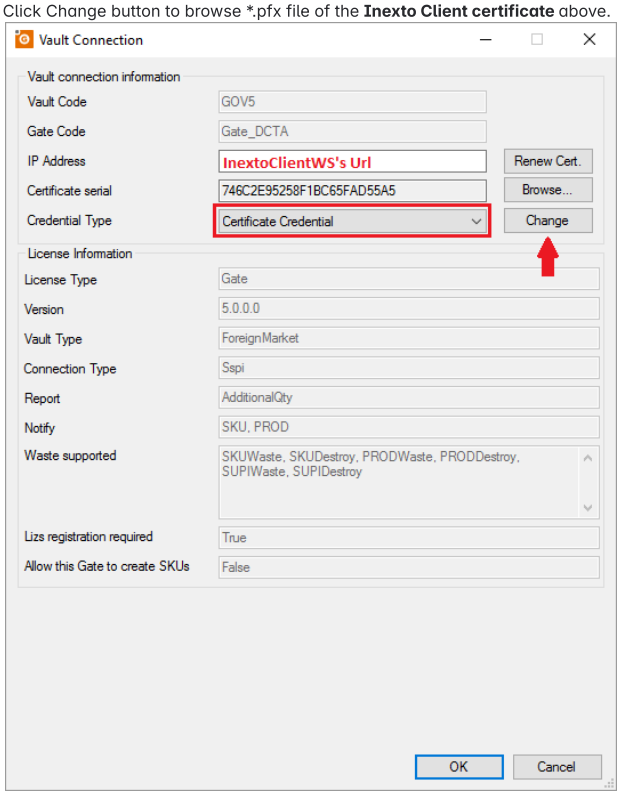
1. On Vault Proxy server, generate an **Inexto Client certificate** for Gate/MCG
   a. Use *S3rCertGenerator* (Installation Media) or any third party tool to select **Vault Proxy Certificate** and issue an **Inexto Client certificate** for the Gate/MCG.

   > NOTICE: The common name of **Inexto Client certificate** has to be the GateCode of the connecting Gate.

   b. Use **CertificateManager** (Installation Media) to import the new **Inexto Client certificate** (.cer) to Vault Proxy database.
   c. Send *.pfx file of the new **Inexto Client certificate** to Gate/MCG operator.

2. Update Vault connection so that Gate/MCG connects to the Vault Proxy
   a. Make sure the **Inexto Client certificate** above is issued by a trusted CA on the machine where the Gate/MCG is running.
   b. Open Gate/MCG Administration and navigate to Vaults tab.
   c. Edit the Vault connection of the Vault which you want to connects via the Vault Proxy.
   d. Update field IP Address to the Url of InextoClientWS.
   e. Change the Credential Type to Certificate Credential.

f. Click Change button to browse *.pfx file of the **Inexto Client certificate** above.



g. Click OK to save the changes.

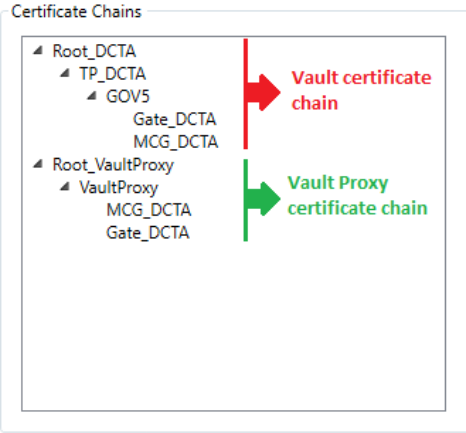
Configure VaultProxy to connect to Vault (GateManagerWS)

1. On GateManagerWS server:
   a. Use *S3rCertGenerator* (Installation Media) or any third party tool to issue a **Vault Client certificate** (e.g: Snedai Client certificate) for the Gate/MCG.

   > NOTICE: The common name of **Vault Client certificate** has to be the GateCode of the Gate.

   b. Follow /wiki/spaces/CDFYMASTER/pages/399849374 to create an IISClientCertificateMapping for the **Vault Client certificate** above and the Factory's login user.
   c. Send *.pfx file of **Vault Client certificate** and it's certificate chain to the operator of Vault Proxy server

2. On Vault Proxy server:
   a. Make sure the **Vault Client certificate** above is issued by a trusted CA on the Vault Proxy server.
   b. Use **CertificateManager** (Installation Media) to import the **Vault Client certificate** (*.pfx file) and its certificate chain to Vault Proxy database.
   c. Check the certificate chains in Vault Proxy database. It should contains two certificate chains as the following.



   d. Open InextoClientWS web.config file
      i. Update GateManagerWsUrlFragment to the Url of GateManagerWS.

      > Note: The format of this setting is similar to IP Address field of Vault Connection in Gate/MCG.

      ii. Correct DBConnectionString setting to access the Vault Proxy database.

Windows Administrators: Certificate Management

The INEXTOR Reference Guide for Certificate Management is a comprehensive document that provides guidance on managing certificates for Id Issuers.

The document covers a range of topics, including importing and installing certificates, using the Certificate Manager, and best practices for managing certificates.

The guide is intended for users of all levels of experience and provides step-by-step instructions and helpful tips to ensure that users can effectively manage their certificates

> Managing certificates (that have private key) should be done with great care. As a minimum, consider the following:
>
> - Make sure to remove all certificates from your PC once no longer needed (that is, once the certificates have been created, they should no longer be stored on your PC).
> - Store the certificates securely and have them password-protected.
> - When transporting the certificates you should use secure communication (not e-mailing) and have the certificates password-protected.

Below are the links to the documentation lists for certificate management:

Certificate Import for Id Issuers

This page explains how to import certificates into the Vault database using the Vault UI.

Certificate Manager

This page explains how to import certificates into the Vault database using the Certificate Manager tool.

Installing Certificates

This page explains how to install certificates on a local computer in Windows Certification Store.

The following documentation is for legacy users **(before 2024)**

/wiki/spaces/INEXTOR/pages/395812029

This page explains how to create root, Technical Provider (TP), Vault and Liz certificates using the Root Certificate Manager and TP Certificate Manager tools.

/wiki/spaces/INEXTOR/pages/395812035

This page explains how to use the Certificate Generator tool to sign IIS certificate requests.

Certificate Tool

Introduction

Version 10.1.0.5020 introduces a comprehensive update to certificate management.

Certificates generated by earlier versions of INEXTOR (specifically for Liz licenses, Gate connections, or actor authentication) are designated as version 5 and were signed using the SHA1 algorithm. Since SHA1 is now deemed outdated, INEXTOR has transitioned to using SHA2-256 for signing its certificates, which are now classified as version 10. It is important to note that older certificate versions can still be generated for compatibility with legacy Gates and Lizs.

Users with existing Vault installations are encouraged to upgrade their certificates to version 10. Meanwhile, new installations should utilize the version 10 certificates from the outset.

The newly introduced certificate tool is capable of performing both actions. As the migration process is a one-time operation, detailed instructions can be found at the end of this page.

Download link: Certificate Tool

New Vault Installation

Before it can be started, a Vault needs:

- A root certificate, which is self-signed,
- A technical provider (TP) certificate, signed by the root certificate,
- A Vault certificate, signed by the technical provider certificate

INEXTOR uses other types of certificates:.

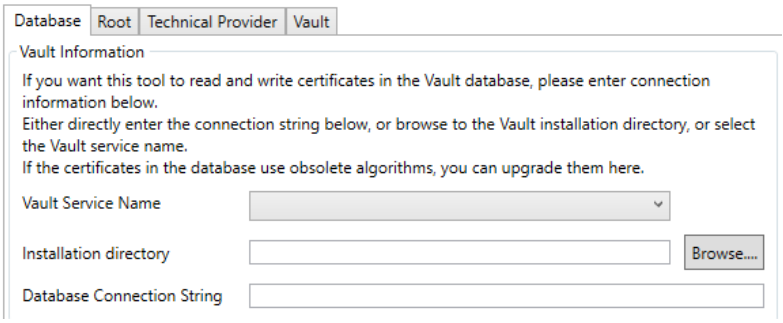
Certificate Type	Signed by	Generated
Gate	Vault certificate	On the Vault Web site, when managing a Gate connection
Liz	TP certificate	On the Vault Web site, when creating Liz licenses
MSK	Vault certificate	By the Vault, when setting up the Liz
Actor	Vault certificate	On the Vault Web site, when managing entities.

They don't require a tool, as the Vault can generate them during its normal operations (see the 'Generated' column in the table above).

These certificates are used to authenticate a program that asks to access the Vault. When the Vault itself wants to authenticate to another program (e.g. an ID issuer), another type of certificate is used. This other type is generated by the organization managing the program and is not handled by this tool. The Vault administration program should be used to import such a certificate.

Connecting to Vault database

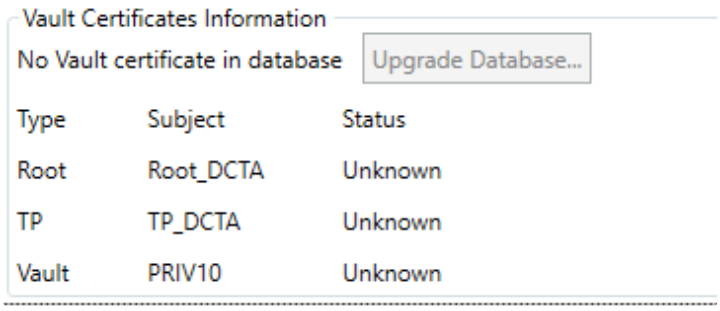
After launching the certificate tool, you need to connect it to the Vault database. This is done in the first tab of the tool:



The tool needs the full database connection string. There are 3 ways to enter it:

- Paste directly in the 'database connection string' text box. If you have the VaultService.exe.config file at hand, you can copy the *DBConnectionString* setting from there.
- If the directory where the Vault Service will run is accessible from the tool, you can enter the directory path in the 'installation directory' text box. The tool will automatically read the VaultService.exe.config file and get the connection string from there.
- If the tool is running on the same machine as the Vault server, you can select the desired Vault in the 'Vault service name' drop-down list. The tool will thus find the Vault installation directory and the VaultService.exe.config file.

After a successful connection, the tool displays, in the same tab, the certificates it find there and their status:



On a new Vault installation, there is no certificate at all.

Reading the database can take some time, so there is no database access while the user enters data in the 'database connection string' text box. If you type it in manually, you need to set the focus to another part of the window to trigger a database read. If you choose a directory or a service name, the database read is immediate.

General Certificate Information

The bottom part of the window shows information that is common to all certificates. Each certificate requires a subject, an organizational unit, an organization name, a city, a state and a country. The subject will be given for each certificate, as explained below. The organizational unit is the certificate type (i.e. Vault, TP or root). The other 4 elements can be set in the tool.

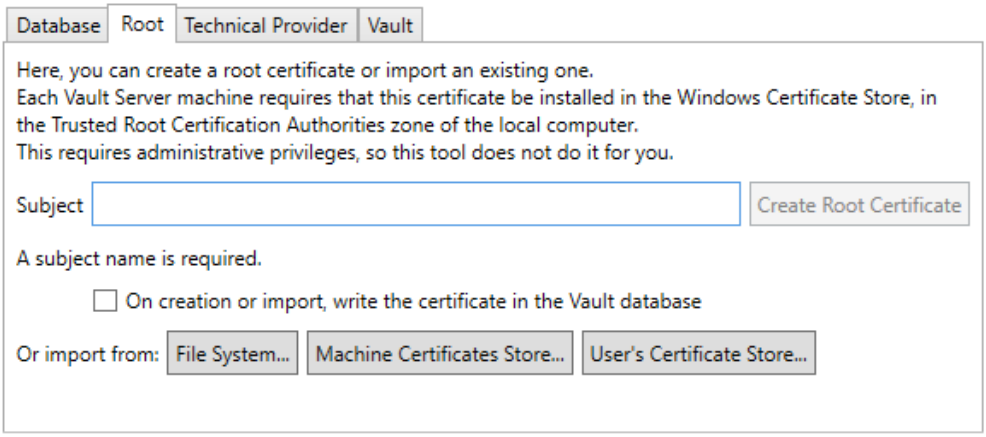


If the connected database already contains certificates, their contents are used to populate this information.

Creating certificates generates some files, so the tool needs an output directory in the 'Save files in' text box. Most creation buttons are disabled until there is a valid directory in this text box.

Create a Root Certificate

The second tab of the tool, called Root, allows creating root certificates.



The only missing information to create a root certificate is the certificate subject, which should be entered in the 'Subject' text box. Note that spaces are not allowed.

Activate the 'on creation or import, ...' checkbox and click the 'Create Root Certificate' button.

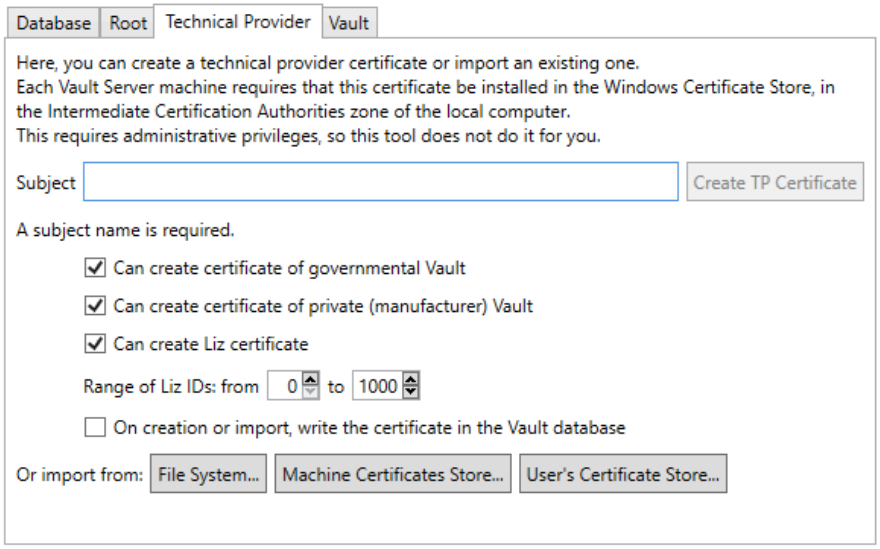
The following actions are performed:

- A self-signed certificate is created,
- A zip file with the certificate and a PowerShell script to install it is created,
- The certificate was imported in the database (if the checkbox was activated).

Always keep the zip files in a secure location, as you may need the certificates later on.

Create a Technical Provider Certificate

The third tab of the tool, imaginatively called 'Technical Provider', allows creating TP certificates.



The process is identical to creating a root certificate, except that a TP certificate contains licensing information:

- whether the new certificate will be able to create a governmental Vault certificate,
- whether the new certificate will be able to create a manufacturer Vault certificate,
- whether the new certificate will be able to create Liz licenses.
  - If yes, the range of Liz identifiers needs to be specified.

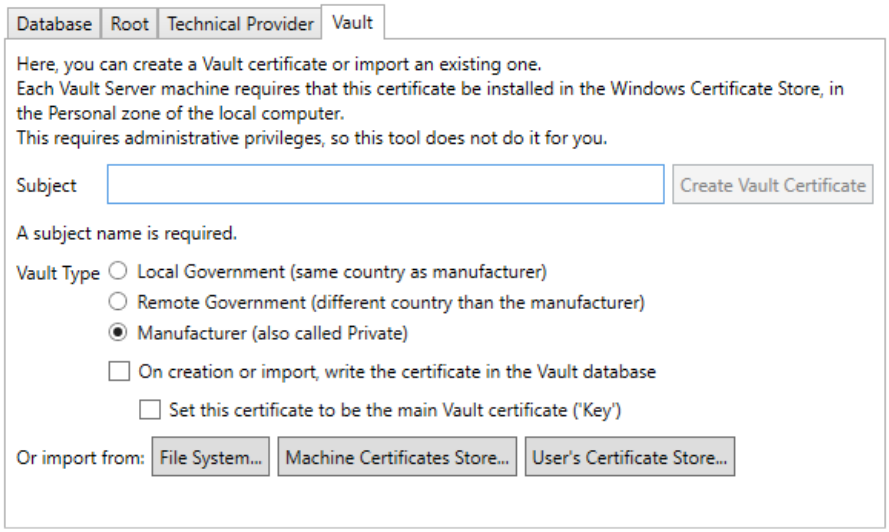
The corresponding checkboxes and text boxes allow setting these properties.

On creation, the same actions are performed as when creating a root certificate.

Creating the TP certificate requires the private key of the root certificate. If you are following the procedure for setting up a new Vault, the previous step (creating the root certificate) has provided the tool with this private key. If you are jumping directly to this tab, the creation button will be disabled until you import a root certificate (see the 'Adding Certificates to a running Vault' section).

Create a Vault Certificate

The fourth tab of the tool, called Vault, allows creating a Vault certificate.



The process is identical to creating a root or TP certificate, except that a Vault certifcate contains the Vault type:

- Local government
- Remote (foreign) government
- Manufacturer (i.e. private Vault)

The corresponding radio buttons allow setting the type.

When importing the newly-created certificate, a new option enables to set the certificate as the main one for the Vault. This is required when installing a new Vault.

Similarly to creating a TP certificate (see note above), creating a Vault certificate requires the private key of its parent, the TP certificate. If you are following the procedure for setting up a new Vault, the previous step (creating the TP certificate) has provided the tool with this private key. If you are jumping directly to this tab, the creation button will be disabled until you import a TP certificate (see the 'Adding Certificates to a running Vault' section just below).

The Vault instance is ready to start.

In order to accept Gate or actor certificates that will connect to the Vault, Windows needs to know about the full certificate chain (root - TP - Vault). Each creation above has generated a zip file with a PowerShell script. Each script must be run on each machine where the Vault server runs, in order to install the certificates in the Windows certificate store. The script references the correct section in the certificate store (Trusted Root Certification Authorities for the root certificate, Intermediate Certification authorities for the TP and Vault certificates).

For Gate installations, these scripts must also be run on the Gate machine if it uses certificate authentication to authenticate to the Vault.

Adding Certificates to a running Vault

After a Vault is up and running, creating new certificates may be needed. The typical example is to create a new TP certificate, in order to generate new Liz licenses (on the Vault Web site).

The process is similar to the first installation. The only difference is the need to provide a parent certificate (except in the case of the root, of course).

Look back at the screenshots for root, TP and Vault certificates tab. There is a range of buttons for importing:

- a PFX file,
- a certificate from the 'Computer' part of the Windows certificate store,
- a certificate from the 'Current User' part of the Windows certificate store.

When importing from the Windows certificate store, the tool automatically chooses the correct certificate section (e.g. 'trusted root certification authorities' for a root certificate).

After connecting the tool to the Vault database and importing the parent certificate in the tool, select the certificate characteristics, its subject name and click on the creation button. As in the new installation procedure, a zip file with the certificate and a PowerShell script is created in the output directory.

Creating Certificates without installing them in the Vault Database

It is also possible to create certificates and import them later (or not at all) in the Vault database. Simply do not activate the 'On creation or import...' checkbox. It is also unnecessary to connect the tool to the database (note that, in this case, the checkbox is disabled).

The rest of the procedure is identical, including the requirement to load a parent certificate.

Importing Certificates in the Vault Database

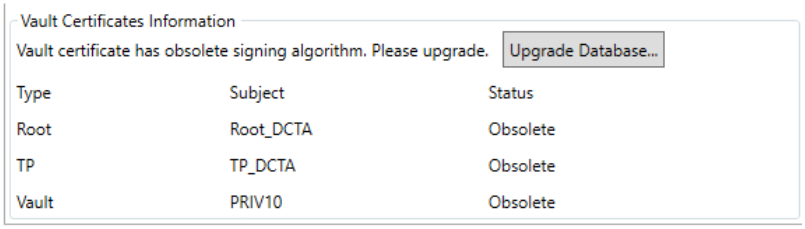
Existing certificates can also be imported in the Vault database.

The process is again identical: connect the tool to the database, activate the 'On creation or import...' checkbox and import a certificate with the import buttons.

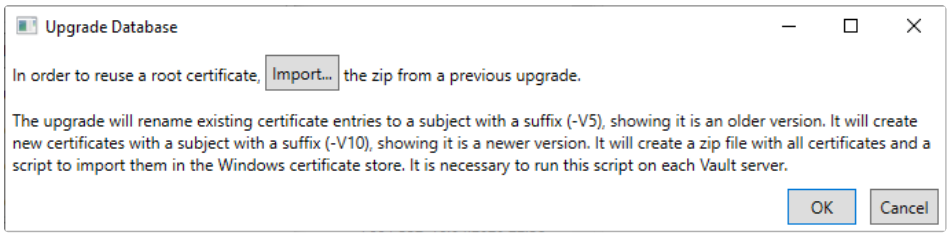
Migrating an older Vault

Any Vault with version under 10.1.0.5xxx should be upgraded to the V10 certificates.

After connecting the tool to the Vault database, the certificates information part of the first tab shows that the certificates are obsolete and the 'Upgrade Database' button gets activated (assuming an output directory has been set).



After clicking on this button, a confirmation window opens:



Simply click on the 'OK' button, and new certificates will be created in the database.

The exact operations that the tool performs are:

- In the dbo.Certificates table of the Vault database:
  - The 'IssuedTo' column of the existing records for the Vault, TP and root certificates get a '-v5' suffix in the subject part (it is not possible to change the certificates themselves, as this would break existing clients),
  - The 'IssuedBy' columns of the existing records for certificates created by the Vault, TP or root certificates are modified to include the '-v5' suffix in the subject part,
  - The 'IsKey' column of the existing record for the Vault certificate is set to 'False',
  - New records are created for the new Vault, TP and root certificates, with a '-v10' suffix,
  - The 'IsKey' column of the new record for the Vault certificate is set to 'True', making it the main Vault certificate.
- A zip file is created with all certificates and a PowerShell script to install them.

⚠ Please look at the note above about needing to install the certificates on each Vault instance, as it applies in this scenario too.

Do not delete the old (version 5) certificates from the Windows certificate store as they are still needed for old clients.

If sharing the root certificate between several Vaults is needed (say Vault A and B), you can migrate Vault A and then import the created zip file when migrating Vault B, using the 'Import...' button on the migration window.

ℹ After this migration, there is a small difference between the old certificates stored in the Vault database and the 'IssuedTo' and 'IssuedBy' columns. As the software uses the 'IssuedTo' and 'IssuedBy' to find certificate chains, this difference does not matter.

ℹ After this migration, the new certificate of the Vault has a subject with a '-v10' suffix. The subject of this certificate is used to define the Vault name. INEXTOR removes this suffix when displaying the Vault name to the user.

ℹ It is not necessary to restart the Vault server after migrating the certificates chain.

Certificate Import for Id Issuers

Certificates that used to be installed in the Windows Store to authenticate with the Id Issuers must now be imported through the Vault UI.

The certificate that should remain selected on the UI is of type *Client* and must have a password, however it is still possible to import *Root* and *Intermediate* certificates as well.

The system will automatically add the *Root* and *Intermediate* certificates to the Windows Store once the chain is correctly imported.

How to import a certificate chain

1. Go the Id Issuer edit view;
2. In the Authentication Certificate row, click on Import;
3. Select the Root certificate (if the Id Issuer required one);
4. The system will alert the user that the certificate just imported does not have a private key: this is normal, the certificate is imported and will successively added to the Windows Store;
5. Select the Intermediate certificate (if the Id Issuer required one);
6. The system will alert the user that the certificate just imported does not have a private key: this is normal, the certificate is imported and will successively added to the Windows Store;
7. Select the Client certificate;
8. The system will ask the user to insert a password (some Client certificates do not have it, you can simply click ok), then the certificate will be imported to the database.

Legacy

Certificate Manager

This tool is used for importing and managing certificates into the Vault database (see Vault Setup - Database). Download link: Certificate Manager

It is important NOT to use version 3.2.1 of the tool with older versions (3.2.0 and earlier, or with Gate 3.2.2), since it is not compatible with the algorithms used in those versions.

The tool is also used to manage data protection. Certificates with private keys in the database can be protected. For being able to read and write the private keys in the database, keys for this purpose are stored in the Vault service configuration file. Furthermore those keys can be encrypted and can only be used on the specific machine and account where the Vault NT service is running.

The information is structured as follows. The sections marked as OPTIONAL are only relevant in case data protection is desired.

- Prerequisites
- Create keys for data protection (OPTIONAL)
- - Introduction
  - Create global key pair
  - Create specific key pair
  - Prerequisite to import certificates into Vault database
- Import certificates into Vault database
- Create keys for Vault services (OPTIONAL)

Prerequisites

Prior to launching the tool, configure the following parameters in the configuration file (CertificateManager.exe.config):

```
<!-- if pfx imported through the tool need a password, specify it here -->
<add key="CertificatePrivateKeyPassword" value="" />
```

The three Certificate files: Root (public key), TP (public key) and Vault Certificate (private key) should be reachable from the Server (on a local drive or a shared drive).

Create keys for data protection (Double RSA)

This section is only necessary if data protection is going to be used. Otherwise go to Section 3 to import the certificates into the database.

Introduction

Double RSA is used to protect the AES keys, which are in turn used to protect certificates containing private keys, which are stored in the Vault database. The private key for the certificate[s] (normally the Vault certificate; but the encryption algorithm will automatically encrypt any certificates with a private key in the database). Currently only the Vault certificate contains a private key) is then stored encrypted, together with the AES key (which is encrypted with the global public key and with the specific public key). Sections 2.2 and 2.3 describe the procedure for creating these keys. This means that the protected certificates stored in the Vault database can be decrypted by decrypting the AES key - either with the global private key or the specific private key. When deploying the Vault, only the specific key pair (public and private) and the global public key are deployed. The global

private key is only used in recovery situations where the specific key pair and the global public key have been lost. Furthermore, the global public, the specific public and the specific private keys, for encrypting/decrypting the AES key (which is in turn used for decrypting the protected certificates), are encrypted in the VaultService.exe.config configuration file, by means of Microsoft Data Protection. This is described in Section 4.



Certificate Data

This diagram shows how the keys and certificates are stored in the Certificates table in the Vault database.

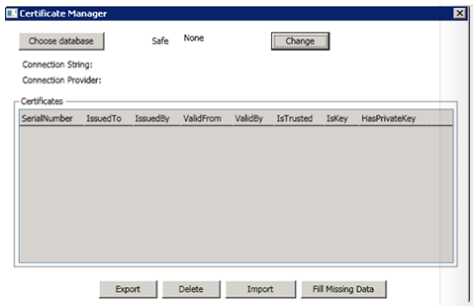
AES key	This key is generated by the Vault service whenever encrypting a certificate (eg: the first time a certificate is imported, if the RSA keys are changed or if the global private key is used to recover). It is then immediately encrypted by both the RSA global and the RSA specific public keys and stored in the certificate table, together with the certificate.
Global public key	Used to encrypt the AES key. Usually this key would be common for all the Vaults belonging to an organization. In this way the Global private key can be stored centrally and be used in a recovery where the specific public and private keys have been lost.
Global private key	Used to decrypt the AES key, when the specific private key has been lost.
Specific public key	Used to encrypt the AES key. Usually this key is unique for each Vault. If several Vault services support the same Vault installation in a load balancing solution, all the Vault Services would use the same specific public and specific key.
Specific private key	Used by the Vault service to decrypt the AES key.

Create Global key pair

Creating the global key pair should only be done once per organization! Once created, the global key pair should be kept in safe storage centrally so that any Vault RSA keys within the organization can be restored.

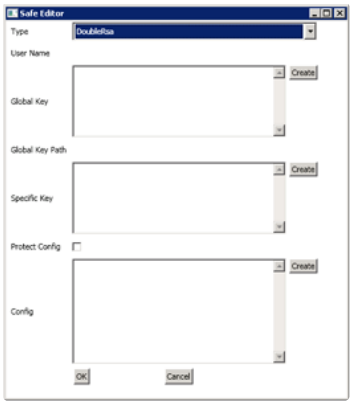
The global public key should be distributed to the person(s) installing the Vault. The global private key is only to be used in a recovery situation.

Launch the Certificate Manager.	



Certificate Manager

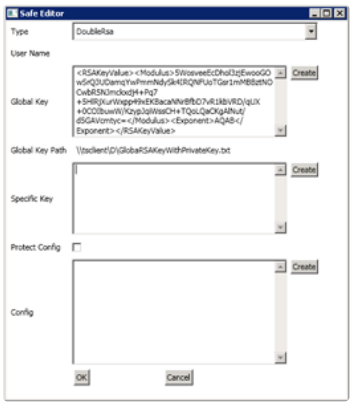
Press the button **Change**.



Safe Editor

Select the **DoubleRsa** method in the combo box. Press **Create** next to Global key. A dialog pops up allowing the key pair containing both public and private keys to be saved. Save the keys as: GlobaRSAKeyWithPrivateKey.txt

Save this key in a safe place centrally for recovery, in case a specific key pair is lost.



Create Global Key

Click in the textbox Global Key and press CTRL-A to select the text.

Copy the contents to a text file and name it "Public key for the Global key.txt" and keep it in a safe location.

The reason for safekeeping is that it is used for all Vaults in the organization.

The recovery using the GlobaRSAKeyWithPrivateKey depends on the fact that all the Vaults use the same global public key.

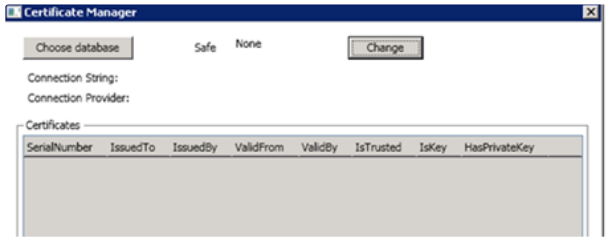
Distribute "Public key for the Global key.txt" to the person(s) installing the Vault using a secure method of file transfer.

Create Specific key pair

Creating the specific key pair should only be done once for a complete Vault, even if the Vault is deployed on several servers!

The specific key pair is used for encrypting/decrypting certificates that contain private keys in the Vault database.

Launch the Certificate manager.



Certificate Manager

Press the button **Change**.



Safe Editor

Select the **DoubleRsa** method in the combo box. Open the file "Public key for the Global key.txt" and paste the content into Global key text box.

Do not click the **Create** button, since this will create a new Global Key pair.



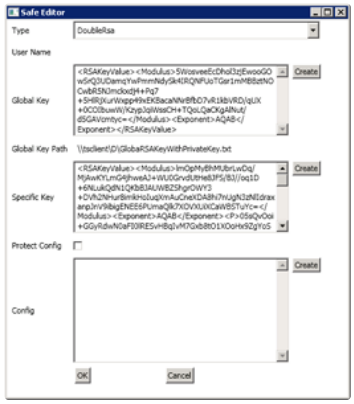
Enter Global Key

Press **Create** next to Specific key.

Click in the textbox **Specific Key** and press CTRL-A to select the text.

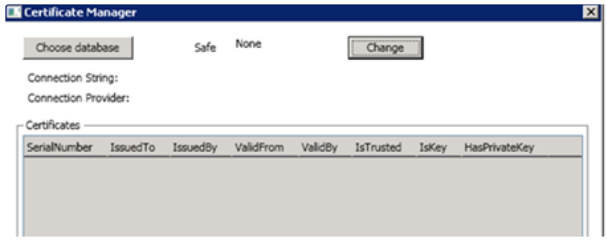
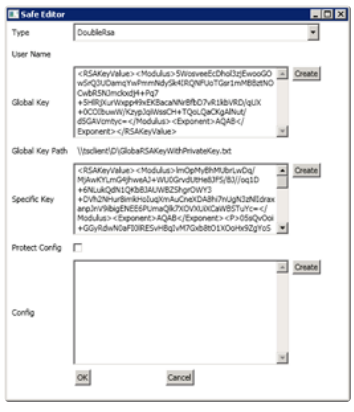
Copy the contents to a text file and name it "Specific key.txt" and keep it in a safe location. This key pair, (containing both private and public keys), is needed if the configuration file for the Vault is lost. With this key and the global public key, the configuration can be restored and the application can start.

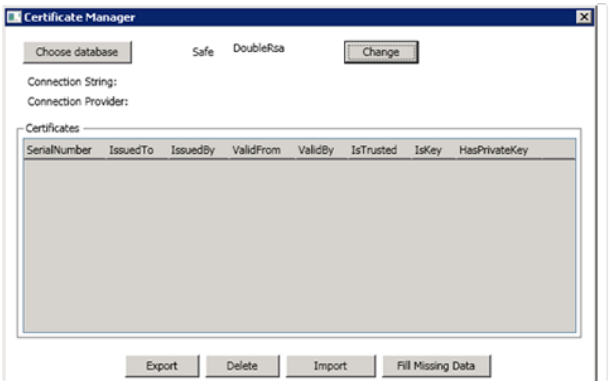
As a last resort, the global private key can be used to reinitialize the configuration.



Create Specific key

Prerequisite to import certificates into Vault database

Launch the Certificate Manager.	 Certificate Manager
Press the button **Change**.	 Safe Editor
Select the **DoubleRsa** method in the combo box. Open the file "Public key for the Global key.txt" and paste the content into Global key text box. Do not click the **Create** button, since this will create a new Global Key pair.	 Enter Global Key
Open the file "Specific key.txt" previously created and pastes the content into **Specific Key** text box.	 Enter Specific Key
Click the button **OK**. The description after Safe should state **DoubleRsa**. Click Choose database and proceed with importing certificates into the database as described in Section 3.	

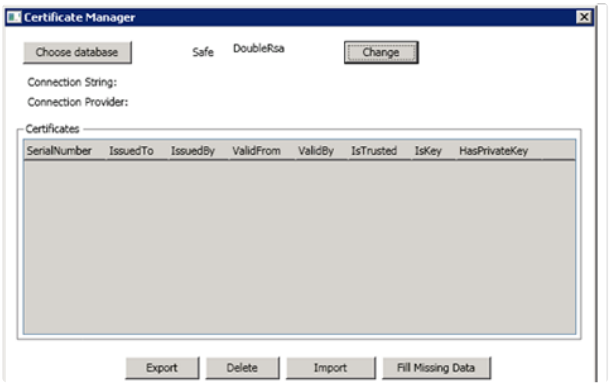


Choose Database

Import certificates into Vault database

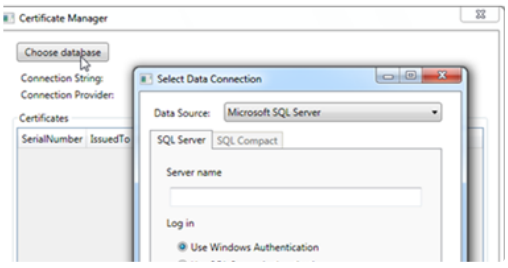
If you intend to use data protection, see Section 2.4 before proceeding.

Click on **Choose Database**.



Choose Database

Enter the Server name and Select the Vault Database. Click the button **OK**.



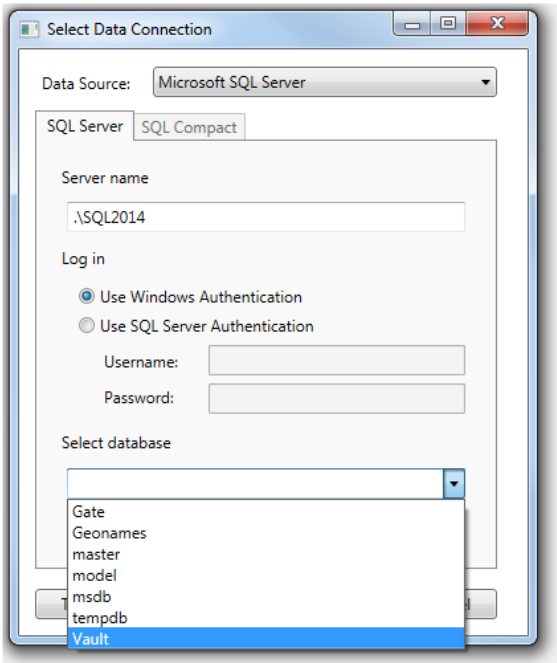
Select Data Connection

Now the Connection String and Connection Provider are shown and it is now possible to import (or export or delete) the Certificates.

Click on Import and locate the Root Certificate (.cer) and click open.

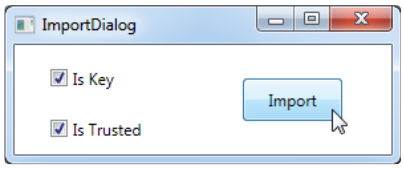
Note: You have to unzip the TP certificate archive generated using the RootCertificateManager and unzip the Vault certificate archive generated using the TPCertificateManager.

These archives contain the entire chain of certificates that need to be imported in the database.



Vault Database

The **Import Dialog** box will appear.



Import Dialog

Root certificate

For the Root certificate uncheck 'Is Key' and check 'Is Trusted'.

TP certificate

Import TP certificate with its private key (.pfx) from the TP certificate archive. For the TP certificate uncheck 'Is Key' and uncheck 'Is Trusted'.

Vault certificate

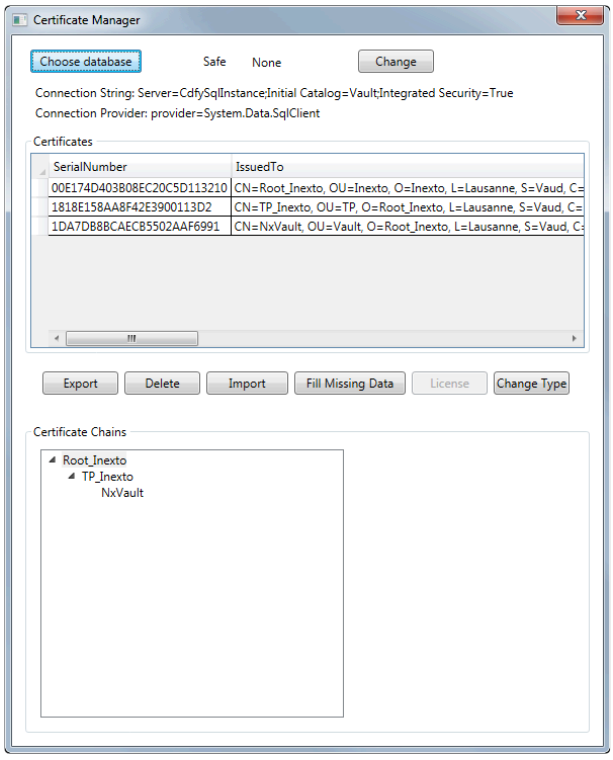
Import the Vault certificate with its private key (.pfx) from the Vault certificate archive. For the Vault Certificate check 'Is Key' and uncheck 'Is Trusted'.

Admin certificate

Import the ADMIN certificate from the Vault certificate archive. For the ADMIN certificate uncheck 'Is Key' and uncheck 'Is Trusted'.

The Certificates will be listed in the Grid, and the Certificate Chain will be shown in the lower-left corner.

Once all the certificates are imported, you can exit the tool.



Certificate List

If the configuration is to be protected with Microsoft Data Protection, proceed to Section 4.

Create keys for Vault services (Protect configuration)

This step is optional. By default we will use the value None for CryptoDatabaseSafeType:

<add key="CryptoDatabaseSafeType" value="None" />

The keys used by the Vault NT service to encrypt and decrypt certificates stored in the database should not be in clear text in the configuration file for the Vault NT Service; hence the keys must be encrypted.

The CertificateManager provides a method to do this for the RSA keys. If other elements are to be protected, please refer to Configuration Protection Tool.

The encryption algorithm (Microsoft Data Protection) used is unique for each server and also unique for the account under which the Vault NT Service is running.

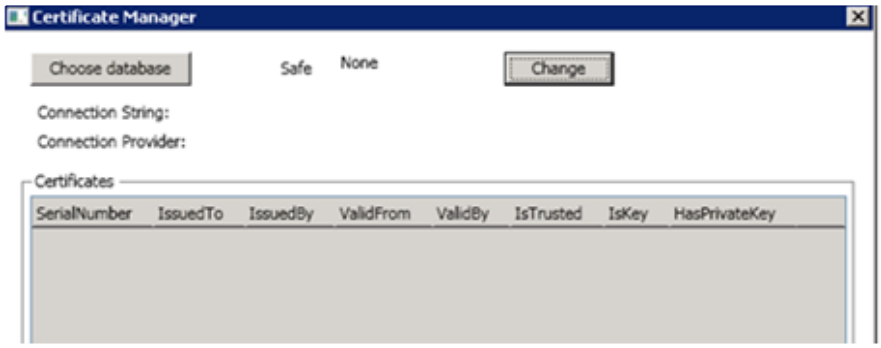
This has the consequence that you have to create protected configuration for every instance of Vault NT Service.

Furthermore, if the account used for running the Vault NT Service is changed, you have to create a new protected configuration.

Make sure that you are logged on as the user that will be running the Vault Service on the same computer which will be hosting the Vault Service.

If it is not possible to log on to the computer with that account, start the Certificate Manager by running the following command from the command prompt in the same directory as the CertificateManager tool (replace VaultserviceUserName with the account that will be running the Vault Service):

```
runas /user:VaultserviceUserName .\CertificateManager.exe
```



Certificate Manager

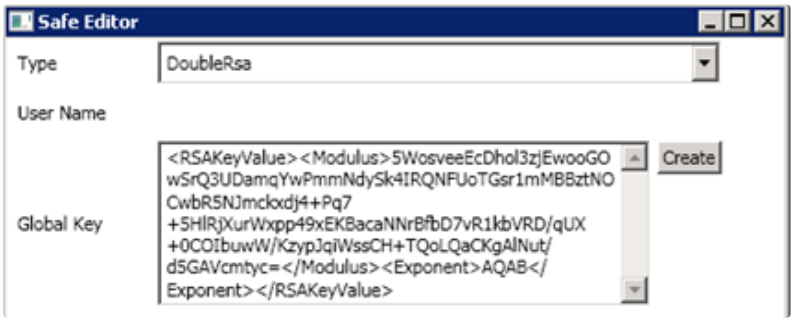
Press the button **Change**.



Safe Editor

Select the **DoubleRsa** method in the combo box. Open the file "Public key for the Global key.txt" and paste the content into Global key text box.

NOTE: DO NOT CLICK CREATE BUTTON, SINCE THIS WILL CREATE A NEW GLOBAL KEY PAIR!



Enter Global Key

Open the file "Specific key.txt" and paste the content into Specific key text box.

DO NOT CLICK CREATE BUTTON, SINCE THIS WILL CREATE A NEW GLOBAL KEY PAIR!

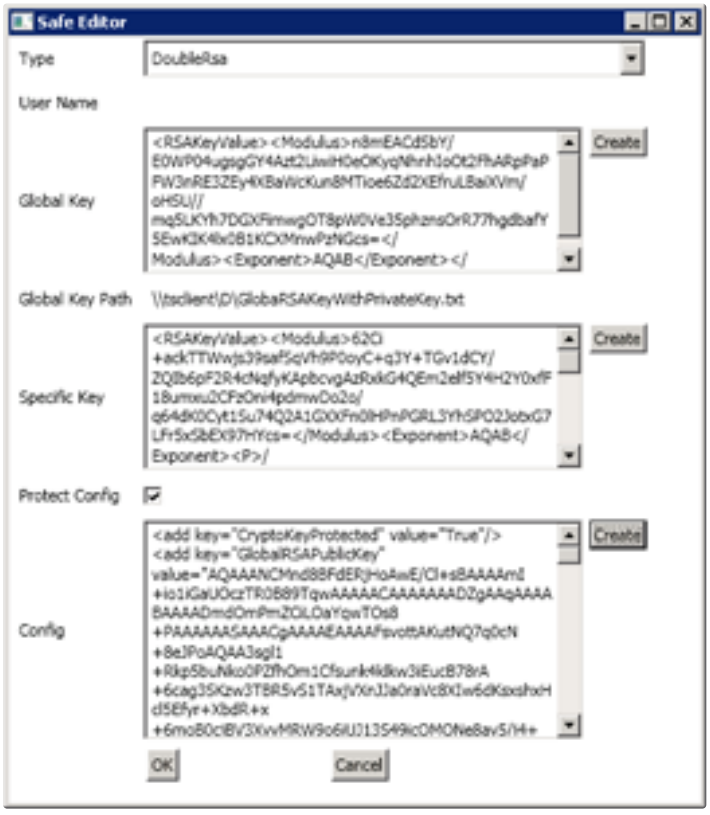


Enter Specific Key

Check the Protect Config checkbox in order to encrypt the keys for the config file and press the create button for the config section.



Protect Configuration



Safe Editor Global View

Mark all the text in the Config section by clicking the text box and pressing Ctrl-A. Right-click and choose copy. Then paste the contents into the VaultService.exe.config file under the section <appSettings>.

```
    ...
    <add key="CryptoKeyProtected" value="true" />
    <add key="GlobalRSAPublicKey" value="AQAAANCM ... nONs=" />
    <add key="CryptoSpecificRSAKey" value="AQAAA ... 4PkWbNR3ug=" />
    <add key="CryptoDatabaseSafeType" value="DoubleRsa"/>
</appSettings>
```

Note: The value “True” in the “CryptoKeyProtected” key must be changed to “true”.

Certificate Install (Windows Certification Store)

This page explains how to install certificates on a local computer in Windows Certification Store.

Microsoft Management Console

To import the Certificates in the "Certificates Management Console" you need to run the mmc application.

mmc.exe is the Microsoft Management Console application and is used to display various management plug-ins accessed from the Control Panel, such as the Certificate Management Console.

On some Windows Servers, a user cannot run the mmc.exe application, even when the user is a member of the Administrator Group.

When trying to run mmc.exe this error message could appear: "Current policies prevent instances of MMC invoked from scripts from opening in author mode." One solution to this problem is the following:

- Run regedit (C:Windows\regedit.exe)
- Explore HKEY_CURRENT_USER\Software\Policies\Microsoft\MMC
- There should be a DWORD value labeled RestrictAuthorMode with value of 1. Right-click on it and change the value to 0
- MMC should now work.

Root Certificate

Import the root certificate by following the steps below:

- Press "Start", "Run" and enter mmc.
- From file menu, select "Add/Remove Snap-in".
- Press "add" and select "certificates" and press "add" in the form.
- Select the Computer Account.
- Select the folder "Trusted Root Certificates Authorities", "Certificates"
- Right-click on "Certificates" and select "All Tasks" and "Import"
- Import the root certificate.
- Double-click the root certificate, and select the "Certification Path" tab. Check that the path is valid ("This certificate is OK") and that the path is 1 long (this is the root).

TP Certificate (Technical Provider)

Install the TP certificate by following the same steps as for Root Certificate, with the following exceptions:

- Select the folder "Intermediate Certificates Authorities", "Certificates"
- Double click on the TP certificate in the "Intermediate Certificates Authorities" folder, and select the tab "Certification Path". Check that the path is valid.

Vault Certificate

Install the Vault certificate (with public key) by following the same steps as for Root Certificate, with the following exceptions:

- Select the folder "Intermediate Certificates Authorities", "Certificates"
- Double click on the Vault certificate in the "Intermediate Certificates Authorities" folder, and select the tab "Certification Path". Check that the path is valid.

Windows Administrators: Special Configurations for Windows

Table of content

- Microsoft Failover Clustering
- Configuring the Schneider Style
- Enabling/Disabling Tobacco features
- How to Automate Resources (resx) Localization with the Translation Tool

Microsoft Failover Clustering

Microsoft Failover Clustering is an *active/passive* configuration. The GateService or VaultService must be installed on both nodes, but only one node runs the GateService or VaultService at a time.

For more information, you can refer to https://www.altaro.com/hyper-v/failover-cluster-manager/

What is Failover Cluster Manager?

This tool is used to create and maintain failover clustering. It deals with:

- roles, that is to say the GateService or VaultService.
- nodes: to implement some redundancy, you need at least 2 nodes running on 2 different physical machines.
- Shared storage, and networking for the cluster.

As with Hyper-V Manager, this is a Microsoft Management Console (mmc.exe) snap-in. It is very small and has no particular hardware or software requirements except for a dependency upon components of Windows Explorer. Because of that, it cannot run directly on Hyper-V Server or Windows Server Core installations. It can be used to remotely control such hosts, however.

How to Access Failover Cluster Manager

If you haven't yet installed the feature, it is a component of the *Remote Server Administration Tools*. Once activated, the icon for Failover Cluster Manager appears in the Administrative Tools group in Control Panel and on the Start menu/screen.

We recommed to setup two machines for the Inextor Service (Gate or Vault) and two machines for the database.

The cluster IP is the IP to use to configure the cluster (from the Failover Cluster Manager windows tool).

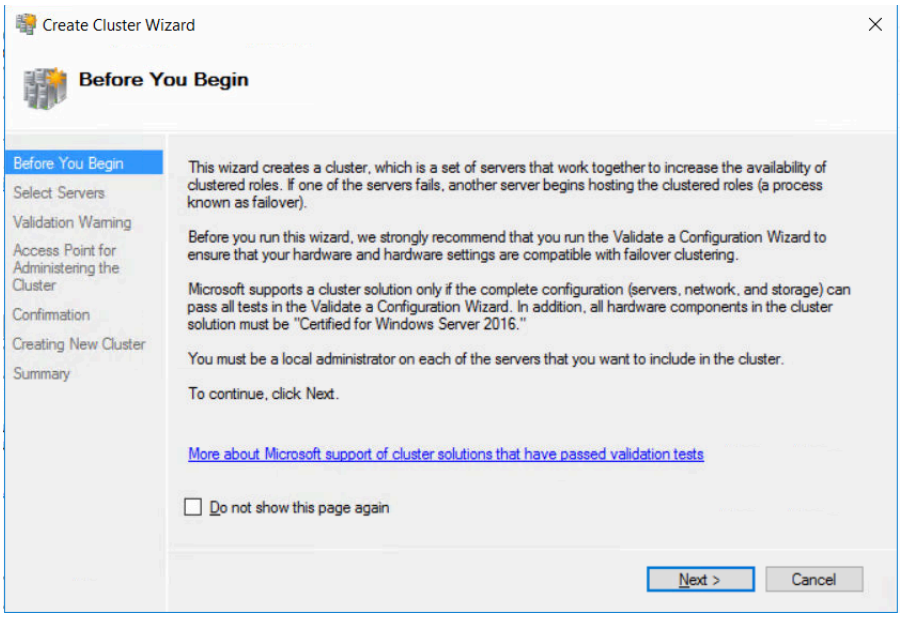
The role IP is the endpoint the client will connect to.

Application	Cluster IP for management	Application IP exposed to clients	Machines IP (nodes)
Gate or Vault	10.178.14.45	10.178.14.46	10.178.14.43 10.178.14.44
Database	10.178.14.49	10.178.14.50	10.178.14.47 10.178.14.48

So for example, on the Gate, the database connection string will reference the IP address of the database Application IP (10.178.14.50).

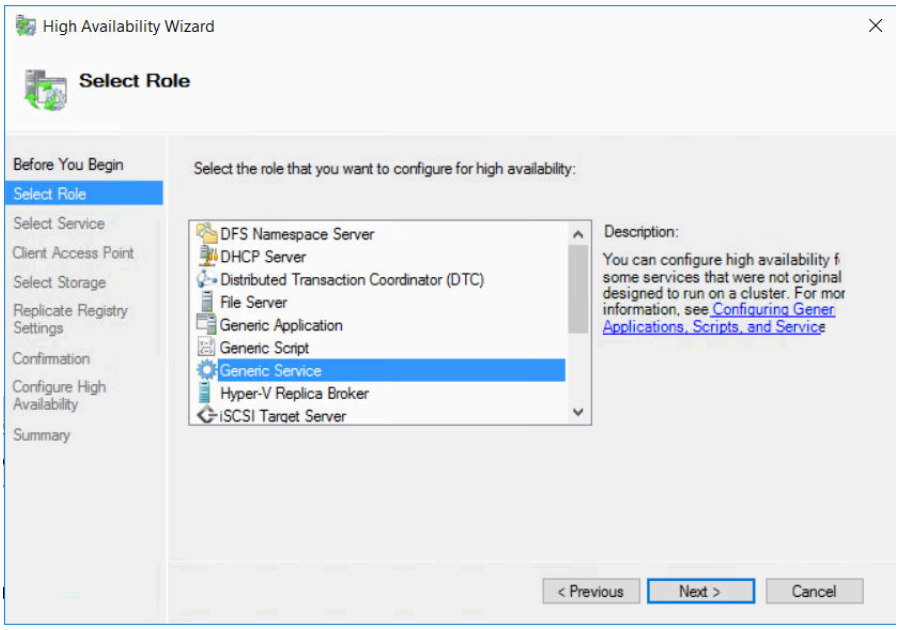
Create the Cluster

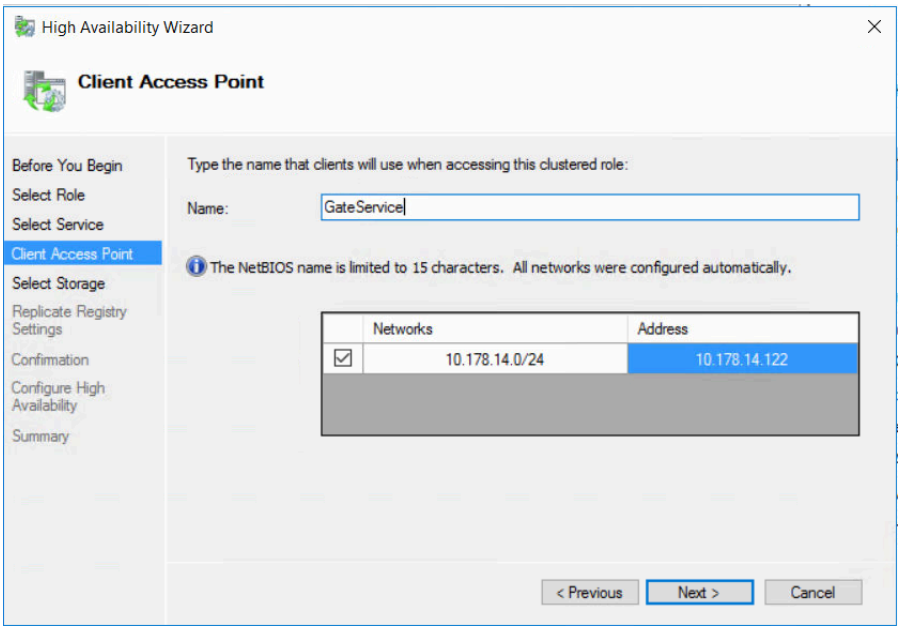
In this wizard, you select the servers and validate that they meet the prerequisites to work as a cluster.



Create the Role

1. Select "Generic Service"
2. Select the service "GateService" or "VaultService".
3. Configure the Client Access Point with a virtual IP address that will be the connection target for clients.

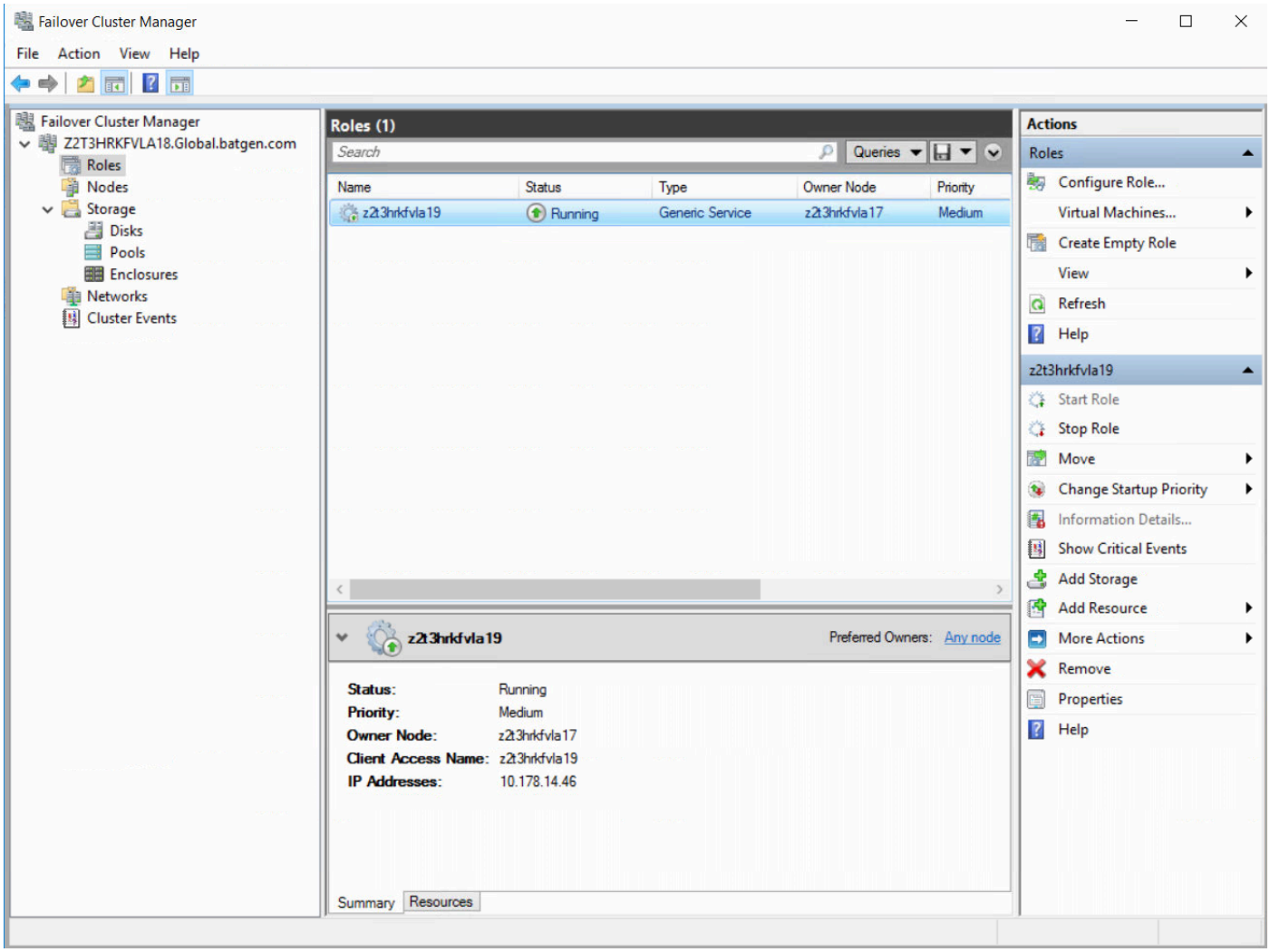




Start the Role

You do not need to start the application services from either node manually. The Cluster will do it automatically.

Click the "Start Role" button:


Failover Cluster Manager
File Action View Help
Failover Cluster Manager
Z2T3HRKFVLA18.Global.batgen.com
Roles
Nodes
Storage
Disks
Pools
Enclosures
Networks
Cluster Events
Roles (1)
Search
Queries
Name Status Type Owner Node Priority
z2t3hrkfvla19 Running Generic Service z2t3hrkfvla17 Medium
z2t3hrkfvla19
Preferred Owners: Any node
Status: Running
Priority: Medium
Owner Node: z2t3hrkfvla17
Client Access Name: z2t3hrkfvla19
IP Addresses: 10.178.14.46
Summary Resources
Actions
Roles
Configure Role...
Virtual Machines...
Create Empty Role
View
Refresh
Help
z2t3hrkfvla19
Start Role
Stop Role
Move
Change Startup Priority
Information Details...
Show Critical Events
Add Storage
Add Resource
More Actions
Remove
Properties
Help

Configuring the Schneider Style

Gate and Liz:

On the app.config of the GateAdmistration and LizOperation, add on the appsettings section the following key:

<add key="CustomSkinType" value="SCHNEIDER_ELECTRICS"/>

Vault Website:

On the web.config, modify the following keys to:

<add key="StyleName" value="styleSchneider"/>
<add key="ThemeName" value="InextorSchneider"/>
<add key="HeaderName" value="compactSchneider"/>

Note: The Vault Administration (Rich Client) is not on the scope of the configurations.

Enabling/Disabling Tobacco features

Disable the Operation Drivers

First install:

If it is the first install, remove the config units before starting the services, otherwise it will populate the OperationDriverComponents and the OperationDrivers Table with the tobacco operation drivers.

Gate:

On the app.config of the Gate Service, go to the SelectedConfigUnits section and then remove the following configurations:

```
<Unit>TpdOperationDriver</Unit>
<Unit>ZetesDriver</Unit>
<Unit>DlrOperationDriver</Unit>
<Unit>RussiaDriver</Unit>
<Unit>ForecastDriver</Unit>
```

Liz:

On the app.domain of the Liz Service (Path: LizService/LizService), go to the SelectedConfigUnits section and then remove the following configurations:

```
<Unit>LizPmiMesDriver</Unit>
<Unit>BatIosisDriver</Unit>
```

Environment already installed:

If an environment was already installed, stop the Gate and Liz services, delete the configunits as mentioned bellow, then remove from the Database the following:

Gate:

First remove all the tobacco related operation drivers components from the OperationDriverComponents table:

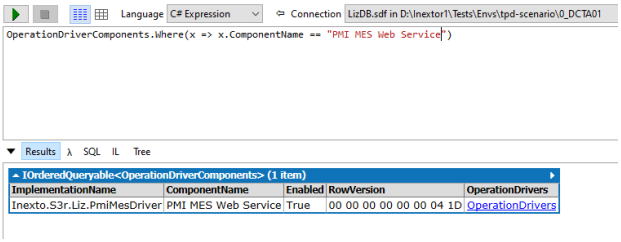
	ImplementationName	ComponentName	Enabled	RowVersion
1	Dcta.Dlr.Gate.DlrDriver	Activation Module	1	0x0000000000000D2C
2	Dcta.Eu.Gate.OperationDriver.ZetesDriver	Activation module	1	0x0000000000000D2E
3	Dcta.Eu.Gate.OperationDriver.ZetesDriver	Zetes Background Job	1	0x0000000000000D2F
4	Dcta.Eu.Gate.TpdDriver	Activation Module	1	0x0000000000000D31
5	Dcta.Russia.Gate.RussiaDriver	Activation Module	1	0x0000000000000D33
6	Dcta.Russia.Gate.RussiaDriver	Inexpress Synchronizer	1	0x0000000000000D34
7	Dcta.Russia.Gate.RussiaDriver	OMS Report Status Monitor	1	0x0000000000000D35
8	Dcta.Russia.Gate.RussiaDriver	Russian External Codes Synchronizer	1	0x0000000000000D36
9	Gate.OperationDriver.CodeOrderForecastDriver	Code Order Forecast Job	1	0x0000000000000D38

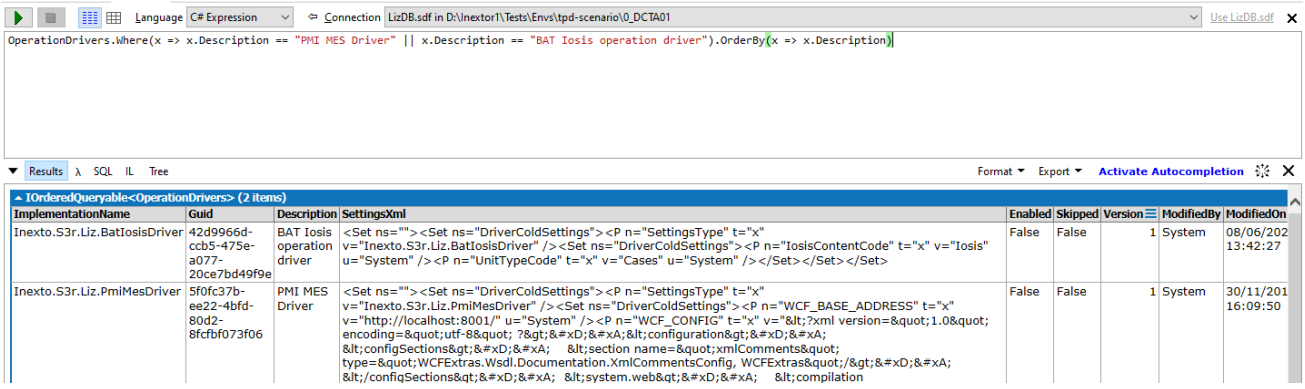
Then remove all the tobacco related operation drivers from the OperationDrivers table:

	ImplementationName	Guid	Description	SettingsXml	Enabled	Skipped	Version	ModifiedOn	ModifiedBy	RowVersion
1	Dcta.Dlr.Gate.DlrDriver	9888DFF1-D867-4EA4-974F-611F663D106E	DLR codes driver	<Set ns=""><Set ns="DriverColdSettings"><P n="Sett...	1	0	2	2019-07-30 07:08:57.640	ELCANET\kla	0x0000000000000D2B
2	Dcta.Eu.Gate.OperationDriver.ZetesDriver	D6EB6563-62EA-40AC-96D3-6B8E49AF419D	Zetes driver	<Set ns=""><Set ns="DriverColdSettings"><P n="Sett...	0	0	1	2020-04-14 22:31:50.440	System	0x0000000000000D2D
3	Dcta.Eu.Gate.TpdDriver	F3013043-0CC3-4E43-8201-E84B01A58AFA	TPD codes driver	<Set ns=""><Set ns="DriverColdSettings"><P n="Sett...	1	0	3	2018-06-11 08:59:27.000	INEXTO\ddelmotte	0x0000000000000D30
4	Dcta.Russia.Gate.RussiaDriver	40AA7DD5-00A6-4D92-990C-734789FADEC7	Russian codes driver	<Set ns=""><Set ns="DriverColdSettings"><P n="Sett...	0	0	1	2020-04-14 22:31:50.440	System	0x0000000000000D32
5	Gate.OperationDriver.CodeOrderForecastDriver	7AC0D669-4234-4C2F-8689-644E1225FB6B	Forecast driver	<Set ns=""><Set ns="DriverColdSettings"><P n="Sett...	1	0	7	2020-04-14 22:38:07.140	system	0x0000000000000D8C

Liz: (Note → Use LinqPad 5 or SSMS CE tool to access the Liz.sdf database)

Remove all OperationDriverComponents and OperationDrivers related to Tobacco.





After all the changes were made, restart the services.

Note: The Vault Administration (Rich Client) is not on the scope of the configurations, because it is strictly Tobacco Oriented.

Enable the tobacco features:

By default the configunits are already configured on the app.config of the Gate Service and the app.domain of the Liz Service, but if they have been removed, no need to reconfigure the Database, just apply again the removed configunits, the database will populate the OperationDriverComponents and OperationDrivers from the application just by starting the services.

Moved to Automated Language Translation

Windows Administrators: Master Data Services

First, follow the Master Data Services installation guide.

Then, install the MDS Importer and Exporter.

Table of content

- MDS Importer
- MDS Extractor
- SQL Server Master Data Services in Azure

MDS Importer

Introduction

The Master Data Services Importer Exporter is a service running on the MDS web server. Its job is to expose a REST web-api where manufacturers can enter their master data. The importer will then translate the data and push it to MDS. An exporter part will then push the validated master data to Inextrack.

Figure 1: Infrastructure. Note that the import and export part are drawn as separate boxes, but they are contained in the same service.

Infrastructure

The MDS Import Export is hosted on the same machine as the MDS web-service. Please follow the SQL Server Master Data Services in Azure documentation first.

ASP.NET Core 6.0

The MDS Import Export is using ASP.NET Core 6.0. The runtime must be installed on the virtual machine.

1. Go to https://dotnet.microsoft.com/en-us/download/dotnet/6.0 and download the latest Hosting Bundle. (Found under ASP.NET Core runtime → Windows)
2. Copy the installer to the virtual machine.
3. Run the installer.

DNS Alias

1. On the Azure portal, navigate to your virtual machine.
2. On the overview page, click on the DNS name link.
3. Press on Create alias record.
4. Choose a name for the API DNS. Set it as Type A.

IIS

Publish Service

For now, the MDS Import Export is manually deployed on the virtual machine. If the POC is accepted and a production version is used, the deployment should be automated.

1. In Visual Studio, select the MdsService project.
2. Right-click and choose to publish.
3. Choose to publish to a folder.
4. Copy the "publish" folder to the virtual machine's desktop.
5. Create a new folder C:\Program Files\MdsImportExport .
6. Copy the content of publish into the MdsImportExport folder.
7. Edit the file appsettings.json. Set option UseHttps to false. (It will use the local endpoint URL and no authentication.)

Setting	Default value	Description
RemoteEndpointUrl	https://inexto-mdm-qa.westeurope.cloudapp.azure.com/MDS/service/Service.svc/login	MDS endpoint when using HTTPS.
LocalEndpointUrl	http://localhost/MDS/service/Service.svc/mdsimporter	MDS endpoint when using HTTP. This is used when the MDS service is installed on the same machine as the MDS webservice.
UseHttps	true	Switch between HTTPS and HTTP connection.

Setting	Default value	Description
UserName	MdsImporterAppPool	User name to connect with https to MDS.
Password	mS5wPer433F%^&	Password to connect with https to MDS.
MaxBufferPoolSize	2147483647	SOAP communication setting.
MaxReceivedMessageSize	2147483647	SOAP communication setting.
MaxArrayLength	2147483647	SOAP communication setting.
MaxBytesPerRead	2147483647	SOAP communication setting.
MaxDepth	2147483647	SOAP communication setting.
MaxNameTableCharCount	2147483647	SOAP communication setting.
MaxStringContentLength	2147483647	SOAP communication setting.
ModelName	ISTD_Products	The model to use in MDS.
VersionName	VERSION_1	The version of the model to use in MDS.
InextrackUrl	https://inextrackapi.azure-api.net/gov/masterData/product	The Inextrack endpoint to push the product's master data.
ExportMinutesFrequency	5	The period at which the exporter push the products to Inextrack.
ExportMustBeApproved	false	When false, the exporter will push to Inextrack all the products. When true, the exporter will only push the approved products.
InextrackSubscriptionKey	80a357b380d144c0b275f55f5fc9a91f	The Inextrack Azure subscription key.
InextrackCertificateFilePath	certificates/INEXTRACK-PMI-MDS-20270728.pfx	The certificate to connect to the Inextrack endpoint.
InextrackCertificatePassword	MDSInexto01	The password of the Inextrack certificate.

Create website

1. Start IIS in admin mode.
2. Navigate to Sites. Right-click and choose Add Website.
3. Set the name MdsImportExport
4. For the physical path, choose C:\Program Files\MdsImportExport .
5. Use HTTP binding, but change the port to 8034.
6. Create the website.
7. Select your newly created website and press Bindings.
8. Add an HTTPS binding.
9. Set the host name to the DNS Alias created previously. The host name must be the domain name (name+zone). In Azure portal, you can click on the alias to get the domain name.
10. Check the Require Server Name Indication.
11. Select the same server certificate as used for MDS.
12. Go to Authentication. Enable Windows Authentication.

Access rights

Folder access

1. Go to C:\Program Files\MdsImportExport and right-click the folder.
2. Select Properties → Security
3. Press Edit → Add
4. Change Location to your server name.
5. Enter the object name as your application pool: *IIS AppPool\MdsImportExport* .
6. Press Check Names. It should find the account and underline the name.
7. Press OK, then give full control to the folder.
8. Press OK on the remaining windows.

Try to browse: http://localhost:8034/swagger . You should have access to the Swagger page.

Application user

The MDS Import Export needs to have access to MDS. We need to create a local user that will run the application pool of the service. This user will then be added in MDS with the appropriate rights.

1. On the virtual machine, start the Windows Run command.
2. Type lusrmgr.msc to get the users and groups management tool.
3. Add a local user. Right-click on Users, then choose New User.
4. Set Name and Full Name to "MdsImportExportUser". Set description to "User running the MdsImportExport app pool.".

5. Choose a password and keep it safe in Lastpass.
6. Select only the check-box "Password never expires".

1. Start IIS
2. Go to the Application Pools and choose the MdsImportExport's one.
3. Click on Advanced settings.
4. Under Identity, modify it to use the custom account we created previously.

1. Go to the MDS website.
2. Navigate to User and Group Permissions.
3. Add the user "MdsImportExportUser".
4. Edit the functions to have access to :
    a. Explorer
    b. Version Management
    c. Integration Management
5. Give all the rights to all the models.

MDS Extractor

Introduction

The Master Data Extractor is a service that extracts data from an SQL database and stores it inside Master Data Services (MDS).

The entities inside MDS are generic, so that they can be re-used for various customers or industries. The extractor uses customizable SQL queries to get the data from the customer's database, then format the data.

Architecture

Figure 1: Master Data Extractor moving data from the customer's database to the Inexto MDS. The Extractor is hosted on the same virtual machine as the MDS webservice.

MDS entities

MDS is configured with entities (tables) that represent generic master data. Each of the entities has a few columns. Any other data that are required are stored in the column Attributes in the JSON format.

Markets

Name	Description	Max size	Type	Mandatory
Code	Unique identifier of the market.	250	Text	Yes
Name	Name of the market.	250	Text	Yes
Attributes	JSON with all other data.	4000	Text	No

Table 1: Represent a market where product will be sold.

Products

Name	Description	Max size	Type	Mandatory
Code	Unique identifier of the product.	250	Text	Yes
Name	Name of the product. (Brand)	250	Text	Yes
Market	Market code where the product will be sold. Link to market.	-	Text	Yes
Attributes	JSON with all other data.	4000	Text	No

Table 2: Represent a product regardless of the packaging in which it comes.

Product Packagings

Name	Description	Max size	Type	Mandatory
Code	Unique identifier of the packaging. Usually <ProductCode>+"_"+<PackagingLevel>	250	Text	Yes
Name	Name of the packaging. For instance, the type of package.	250	Text	Yes
Product	Product code. Link to product.	-	Text	Yes
Gtin	The GTIN of the packaging.	20	Text	No
Quantity in parent	Quantity of this packaging inside its parent.	-	Number	No
Attributes	JSON with all other data.	4000	Text	No

Table 3: Represent the various packaging used by a product during its life cycle.

Product Manufacturing Locations

Name	Description	Max size	Type	Mandatory
Code	Unique identifier. For instance <ManufacturingCenterId>+"_"+<ProductCode>	250	Text	Yes
Name	Name of the product. (Brand)	250	Text	Yes
Product	Product code. Link to product.	-	Text	Yes
Manufacturing center ID	ID of the factory where the product will be manufactured. It must match with the Gate code.	20	Text	Yes
Attributes	JSON with all other data.	4000	Text	No

Table 4: Represent a combination of a product and the place where it will be manufactured. A product can be manufactured from more than 1 location, in which case, we would have one entry per location.

Caching

In order to avoid re-injecting all the master data in MDS each time the extractor runs, a caching mechanism is used. Each time some master data is successfully created or updated in MDS, it will be saved in a JSON file. The next time the extractor queries data from the customer's database, it will compare it with the file cache. The extractor will process only the master data that is different from the cache.

There is one file cache per tables in MDS. The files are stored in the folder "Cache" inside the service's folder.

We opted to cache the data in files instead of in memory, because the extractor runs once every hour (default). Most of the time, the service does nothing, so it was not optimal to block a few gigabytes of RAM for nothing. The extra cost of reading/writing the files once every hour is negligible.
The other benefit of the file cache, is that if the service is stopped, the cache is persisted. This avoids having to reprocess all the master data at each new start of the extractor.

Installation



File 1: The Master Data Extractor. Can be used for testing.

Configuration

App settings

A number of configuration must be made in the **appsettings.json** before starting the service.

Setting	Default	Description
BackgroundJobPeriod	1 hour	How often the service extract master data.
MdsRequestBulkSize	100	Max number of elements sent to MDS at once. This avoids timeout and limits the number of elements to retry in case of communication issue. Create and update requests are different, so you could have a create request with 23 elements and an update request with 77 elements.
DatabaseConnectionString	-	The SQL Server connection string to the customer's database.
QueryProduct	Database/QuerySmdlProducts.sql	Path where the SQL query to get the products is stored.

Setting	Default	Description
QueryMarket	Database/QuerySmdlMarkets.sql	Path where the SQL query to get the markets is stored.
QueryProductPackaging	Database/QuerySmdlProductPackagings.sql	Path where the SQL query to get the packagings is stored.
QueryProductManufacturingLocation	Database/QuerySmdlProductManufacturingLocations.sql	Path where the SQL query to get the manufacturing location is stored.
ProductCachePath	Cache/ProductCache.json	Path where the product's cache will be created.
MarketCachePath	Cache/MarketCache.json	Path where the market's cache will be created.
ProductPackagingCachePath	Cache/ProductPackagingCache.json	Path where the packaging's cache will be created.
ProductManufacturingLocationCachePath	Cache/ProductManufacturingLocationCache.json	Path where the manufacturing location's cache will be created.
MdsCommunicationSettings		Below are the settings to connect to MDS.
RemoteEndpointUrl	https://pmi-mds-qa.testbed.inextor.biz/service/Service.svc/login (QA)	URL of the MDS webservice in HTTPS. (Used for development and testing.)
LocalEndpointUrl	http://localhost:8035/service/Service.svc/mdsexporter (QA)	URL of the MDS webservice in HTTP. (Used in production, when the extractor is installed locally on the MDS machine.)
UseHttps	false	Switch between HTTPS and HTTP. Switch between the above 2 URLs.
UserName	ExtractorMdsUser (QA)	User name to login into MDS,
Password	spTC@2P8Sz3Z&w (QA)	Password to login into MDS.
MaxBufferPoolSize	2147483647	WCF setting.
MaxReceivedMessageSize	2147483647	WCF setting.
MaxArrayLength	2147483647	WCF setting.
MaxBytesPerRead	2147483647	WCF setting.
MaxDepth	2147483647	WCF setting.
MaxNameTableCharCount	2147483647	WCF setting.
MaxStringContentLength	2147483647	WCF setting.
ModelName	-	Name of the MDS model.
VersionName	VERSION_1	Version of the MDS model. Check on the MDS portal to know which version is currently used.

Database query

The queries used to extract the master data are found in the folder Database of the service. They can be modified to add or remove data from MDS. A number of rules must be followed to ensure the good operation of the extractor.

- Query must return a column "**Code**". This value must be unique.
- Query must return a column "**Name**".
- Query must return the other columns from Table 1-4 except Attributes. Those columns can return null if it's not mandatory.
- Query for Table 2-4 must return the unique code of the **market** or **product**. This code will be used by MDS to link the entities together.
- Any number of additional columns can be added to the query. They must return a VARCHAR or NVARCHAR. Those extra columns will be added in the JSON of **Attributes**.
- Be careful that the queries return different codes for each record. Do a "SELECT DISTINCT Code" with and without the distinct and compare the returned quantity. They should be equal. If two records share the same code, the second one will overwrite the first one in MDS.

SQL Server Master Data Services in Azure

Introduction

This documentation describes how the Microsoft Master Data Services is hosted in Azure.

The main documentation followed is :https://learn.microsoft.com/en-us/sql/master-data-services/master-data-services-host-database-on-managed-instance?view=sql-server-ver16

Glossary

Abbreviation	Name	Description
MDS	Master Data Services	A solution from Microsoft to manage master data. It's a module of SQL Server.
SQL MI	SQL Managed Instance	SQL Server hosted in Azure.
VM	Virtual Machine	Virtual machine hosted in Azure.
VN	Virtual Network	Virtual network hosted in Azure.

Infrastructure

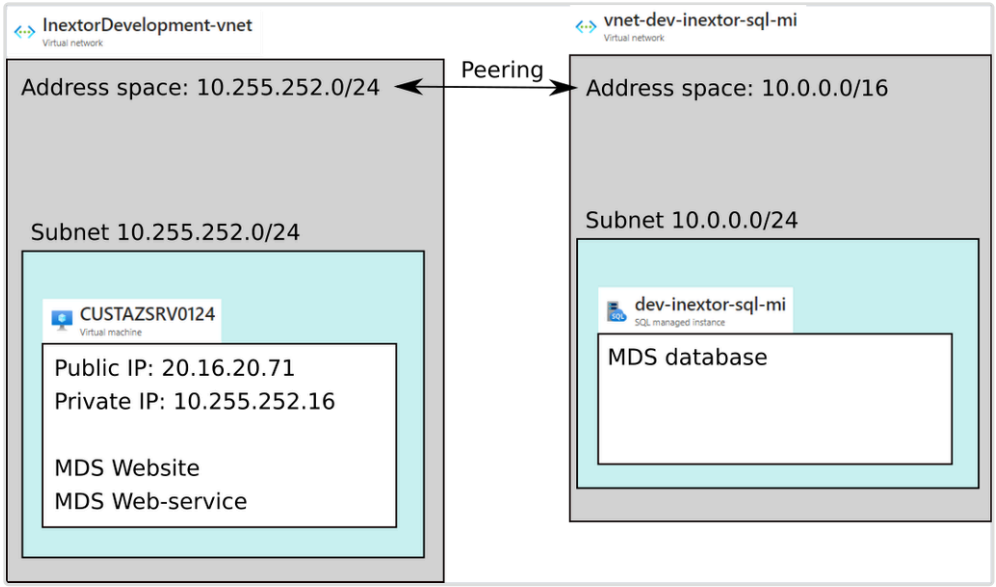


Figure 1: Infrastructure of the Jordan Master Data POC. The MDS database is in an Azure SQL Server Managed Instance. The MDS website and web-service are installed on an Azure virtual machine.

Connect SQL Managed Instance to MDS web server

The server hosting the web application must have access to the SQL MI. There are various ways to do this connection. Microsoft documentation: https://learn.microsoft.com/en-us/azure/azure-sql/managed-instance/connect-application-instance?view=azuresql

For the POC, the SQL MI and VM are on different virtual network. I opted to peer the 2 virtual networks together. See tutorial : https://learn.microsoft.com/en-us/azure/virtual-network/tutorial-connect-virtual-networks-portal

For the future deployments, we could try to have both the SQL MI and VM in the same virtual network.

Installation

Virtual Machine

1. Ask for admin right for the Azure Subscription containing the VM and MI.
2. Remote desktop to the VM. (You can find the connection info in the Azure portal under Connect.)
3. Start the Server Manager
4. Select Add Roles and Features.
5. Add the roles and features list requested in the Microsoft documentation. The doc has one mistake. "Dynamic Content Compression" is not a feature, but a role found under Web Server → Performance.

Database

1. Copy the SQL Server Setup executable on the server for the web application.
2. Run the setup.exe and choose Installation → "New SQL Server standalone installation or add features to an existing installation".
3. Select the license you will use. (For QA, we used the Developer license.)
4. In the feature selection, choose only Master Data Services.
5. Proceed with installation.

Admin user

First, we need to create an admin user that the MDS installer will use to deploy the database.

1. In the Azure portal, go to your SQL Server Managed Instance.
2. Go to Connection Strings.
3. Copy from a connection string the server part. It should be something like: tcp:*****.database.windows.net
4. With SSMS, login to the MI. Use "Azure Active Directory - Universal with MFA". Your user must have admin access to the MI (use Inexto Access Management).
5. Go to the Security folder and right-click on it. Select New→ Login
6. Create a new "SQL Server authentication" login.
7. Choose name and password. Keep them somewhere safe.
8. Add in the Server Roles:
   a. bulkadmin
   b. dbcreator
   c. processadmin
   d. public
   e. securityadmin
   f. serveradmin

g. setupadmin
h. sysadmin

9. Leave the rest as default and save.

Deploy database

1. Start the Master Data Services Configuration Manager in admin mode. (In the Windows start bar, it's located in the SQL Server folder.)
2. Go to Database Configuration.
3. Press Create Database.
4. You will first need to connect to the SQL MI.
   a. The server instance can be found in the Azure portal ; same as in the previous section. It should look like: tcp:*****.database.windows.net
   b. Use SQL Server Account with the login you created in the previous section.
5. Give a name to your database.
6. Use the default collation.
7. Use your personal account for Administrator account. It will be the first account in MDS.
8. Finish the deployment of the database.

Email notification

This step configures SQL Server to use the SMTP server.

1. You will need the Inexto SendGrid SMTP server's info. You can find them in LastPass. Search for "IdIssuerKit - SMTP".
2. In the Master Data Services Configuration Manager, choose Database Configuration.
3. Select the MDS database.
   a. The server can be found in the Azure portal. It should look like tcp:*****.database.windows.net.
   b. Use the SQL Server Account with admin user you created.
4. Go to Database Mail Profile and press Create Profile.
   a. Profile name: MdsEmailProfile
   b. Account name: MdsEmailAccount
   c. Email address: noreply@inexto.com
   d. Display name:
   e. Reply email address:
   f. SMTP server: smtp.sendgrid.net
   g. Port number: 587
   h. This server requires a secure connection (SSL): YES
   i. Basic authentication: See LastPass for user and password
5. Create the profile.
6. Master Data Services URL for notifications: Set the DNS name of your server. This is required for the links inside the emails to work properly. (If you haven't decided yet, you can come back later to this step.)
7. Press Apply.

The SMTP port needs to be authorized for the SQL Managed Instance.

1. Go to Azure portal.
2. Find the Virtual Network where the SQL MI is.
3. Navigate to the subnet and note the Network Security Group.
4. Navigate to the Network Security Group.

5. In Outbound security rules, add a rule:
    a. Source: Range from the sub-network
    b. Destination: Any
    c. Source port: *
    d. Destination port: 587
    e. Protocol: Any
    f. Action: Allow

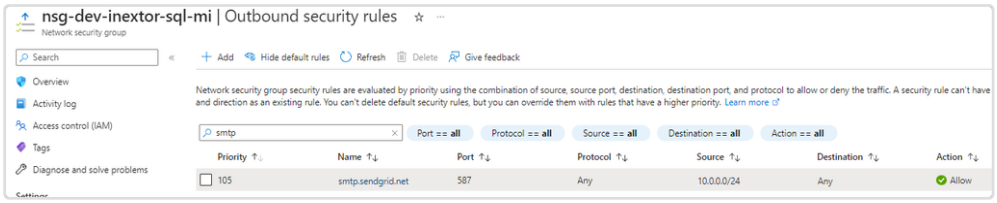


Figure 2: Outbound rule for the SMTP server.

Automated Business Rules

The Business Rules that are created in MDS can be executed from the website, but can't be automated from there. We will create an SQL job that will periodically run the business rules.

When an email notification is configured in the business rule, a user can be notified by email when an entry has failed a rule.

In order to run the business rule, we execute the stored procedure [mdm].[udpValidateModel]. See doc: https://learn.microsoft.com/en-us/sql/master-data-services/validation-stored-procedure-master-data-services?view=sql-server-ver16

1. With SQL Server Management Studio, connect to the Managed Instance.
2. Go to SQL Server Agent→ Jobs
3. Right-click on Jobs and select New Jobs.
4. Enter the name MDS_<DbName>_RunBusinessRules. Write a description.
5. Set the owner to the user you created in the step "Admin user".
6. Category = [Uncategorized (Local)]
7. In Steps, press New
    a. Step name = RunUdpValidateModel
    b. Type = Transact-SQL script
    c. Run as = leave empty
    d. Select the MDS database and past the script that execute the stored procedure. (See sample below.)
    e. In Advanced, set Run as user to the user created in the setp "Admin user".
    f. Press OK
8. In Schedule, configure the schedule to your need. (We advise once per hour.) Press OK.
9. No need to configure the Notifications and Targets. Press OK.

ExecuteStoredProcUdpValidateModel

```
DECLARE @RC int
DECLARE @User_ID int
DECLARE @Model_ID int
DECLARE @Version_ID int

SET @User_ID = (SELECT ID
                FROM mdm.tblUser u
```

```
                WHERE u.UserName = 'CUSTAZSRV0123\Choose a user created in MDS')
SET @Model_ID = (SELECT Top 1 Model_ID
                FROM mdm.viw_SYSTEM_SCHEMA_VERSION
                WHERE Model_Name = 'Choose your model')
SET @Version_ID = (SELECT MAX(ID)
                FROM mdm.viw_SYSTEM_SCHEMA_VERSION
                WHERE Model_ID = @Model_ID)

EXECUTE @RC = [mdm].[udpValidateModel]
   @User_ID
  ,@Model_ID
  ,@Version_ID
  ,1
```

Code 1: Execute the stored procedure that runs the business rules. You might need to come back later to configure the user and model.

We need to give access to the SQL user to run the job and the stored procedure.

1. Go to Security→ Logins. Edit the user created during "Admin user".
2. In User Mapping, map the MDS database with db_datareader, db_datawriter.
3. Map the "msdb" database with db_datareader, db_datawriter, SQLAgentOperatorRole, SQLAgentReaderRole, SQLAgentUserRole.
4. Press OK.
5. Go to Database→ MDS database→ Security→ Users. Edit the user created during "Admin user".
6. In Securables, press Search. Add the Stored Procedure [mdm].[udpValidateModel].
7. Press OK.

Web Application

Deploy web application

1. Start the Master Data Services Configuration Manager.
2. Choose Web Configuration.
3. Select the Default Web Site and press create.
4. In the application pool, use your Windows credentials. (It will only be used during setup.)
5. Finish the web application deployment.

Active directory issue

When browsing the website for the first time, and you log with your user account, you might encounter an error saying the **adalsql.dll** is not found. This DLL or its successor, adal.dll, is found in C:\Windows\System32.

To add the DLL, install SSMS on the Azure Server.

Configure

Connection string

We need to change the SQL connection string of the web application.

1. Go to C:\Program Files\Microsoft SQL Server\160\Master Data Services\WebApplication
2. Copy the file web.config on the desktop.
3. Edit the copy and locate the <connectionStrings>. By default, it's using Active Directory Integrated. We will switch it to SQL Server authentication.

4. Replace "**Authentication="Active Directory Integrated"**" by "**User ID={User ID};Password={Password};**". (Use the same user as the one you created in SQL Server.)
5. Save the file. Close it. Finally, copy it and paste it back to its original folder.
6. Try to browse: http://localhost/MDS

DNS

Set a DNS name for your server. This can be done in the configuration of your VM.

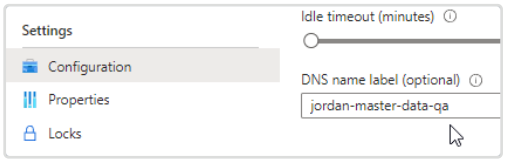


Figure 3: Virtual Machine configuration on the Azure Portal.

Application pool

We need to give appropriate access to the application pool running MDS.

Start IIS. Verify that the MDS application pool is set to ApplicationPoolIdentity.

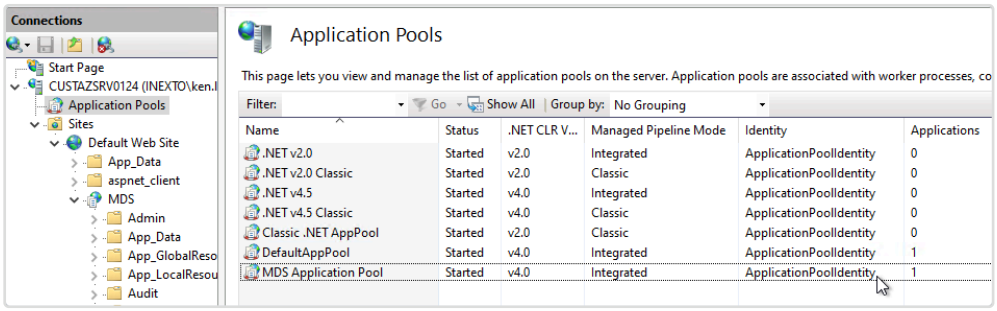


Figure 4: MDS has it's own application pool.

1. Right-click on folder *C:\Program Files\Microsoft SQL Server\160\Master Data Services\WebApplication*
2. Select Properties → Security
3. Press Edit → Add
4. Change Location to your server name.
5. Enter the object name as your application pool: *IIS AppPool\MDS Application Pool* . (Change the second part if your pool has a different name.)
6. Press Check Names. It should find the account and underline the name.
7. Press OK, then give full control to the folder.
8. Press OK on the remaining windows.

Do the same steps for the below folder and files. Some should inherit from the parent folder, but sometimes it doesn't work.

- C:\Program Files\Microsoft SQL Server\160\Master Data Services\MDSTempDir
- C:\Program Files\Microsoft SQL Server\160\Master Data Services\WebApplication\MDSTempDir
- C:\Program Files\Microsoft SQL Server\160\Master Data Services\WebApplication\web.config

1. Start regedit.exe
2. Go to HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Microsoft SQL Server\160
3. Right-click and give full access to the MDS application pool, like you did for the folders.

HTTPS

We need to enable HTTPS. There are 3 components that needs to be compatible with https:

- Website
- API
- Excel connection

Server certificate

The first step is to create a server certificate. One important point, is that the root certificate must contain an Alternative Name with the DNS you choose previously.

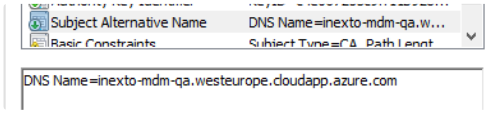


Figure 5: Example of root certificate with an Alternative Name containing the DNS.

If the environment will be used by a customer, you will need a certificate from a certified authority. Ask the Inexto hosting team to provide and configure it.

For the POC environment, a self-signed certificate is created. Do not use the function in IIS to generate it. IIS is using the VM's name in the cert and there is no option to add the DNS.

You can use OpenSSL to generate the root and server certificates. Below, are the commands for the POC.

```
# OpenSSL 1.1.1
# Self signed root and server certificates.

# Create the root key and cert. Use your DNS in the subject and subjectAltName.
req -x509 -newkey rsa:2048 -sha256 -days 3650 -nodes -keyout MdsServerSelfSignedRoot.key -out MdsServerSelfSignedRoot.crt -subj "/CN=inexto-mdm-qa.westeurope.cloudapp.azure.com" -addext "subjectAltName=DNS:inexto-mdm-qa.westeurope.cloudapp.azure.com"

# Create a certificate request. You will be asked info about your company.
req -key MdsServerSelfSignedRoot.key -new -out MdsServerSelfSigned.csr

# Create the server certificate CRT.
# The EXT file should be a text file containing the 3 below lines. Don't forget to change the DNS.
# authorityKeyIdentifier=keyid,issuer
# basicConstraints=CA:FALSE
# subjectAltName = DNS:inexto-mdm-qa.westeurope.cloudapp.azure.com
x509 -req -CA MdsServerSelfSignedRoot.crt -CAkey MdsServerSelfSignedRoot.key -in MdsServerSelfSigned.csr -out MdsServerSelfSigned.crt -days 3650 -CAcreateserial -extfile MdsServerExtension.ext

# Combine the server certificate and the key to create a PFX version.
pkcs12 -export -in MdsServerSelfSigned.crt -inkey MdsServerSelfSignedRoot.key -out MdsServerSelfSigned.pfx

# On the server, install in Local Machine, the root CRT in the Trusted Root Certification Authorities folder.
# In IIS, import the PFX for the HTTPS binding. It should go in the Local Machine-> Personal folder.
# The server certificate should have a valid certificate chain.
```

Code 2: OpenSSl 1.1.1 commands to create a self-signed server certificate and it's self-signed root CA.

IIS

To enable HTTPS, this guide explains some of the changes required, but not all of them: https://learn.microsoft.com/en-us/sql/master-data-services/install-windows/secure-a-master-data-manager-web-application?view=sql-server-ver16

1. Start IIS in admin mode.
2. Click on the website of MDS, then, press on Bindings (right of the window).
3. Add a https binding and use the server certificate created in the previous section.
4. Click on the MDS application, then, double-click Authentication.
5. Enable only Windows Authentication.
6. Select Windows Authentication, then press Advanced Settings.
7. Set the Extended Protection to Off.

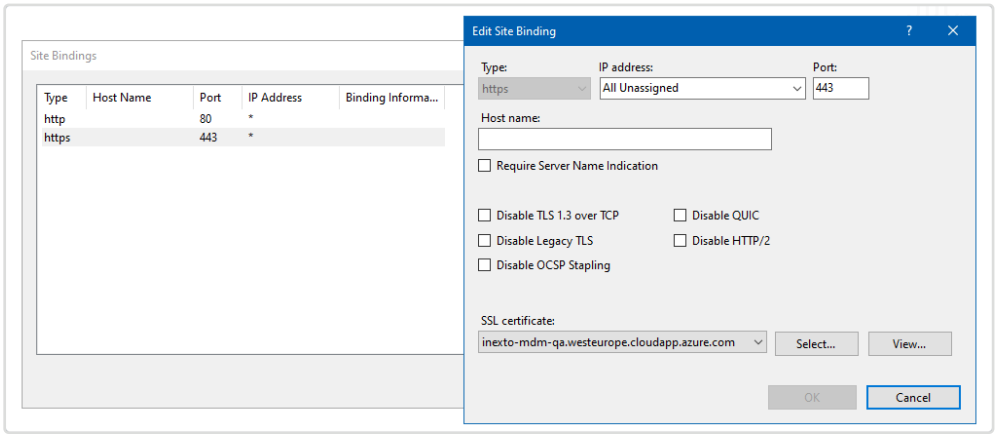


Figure 6: HTTPS binding.

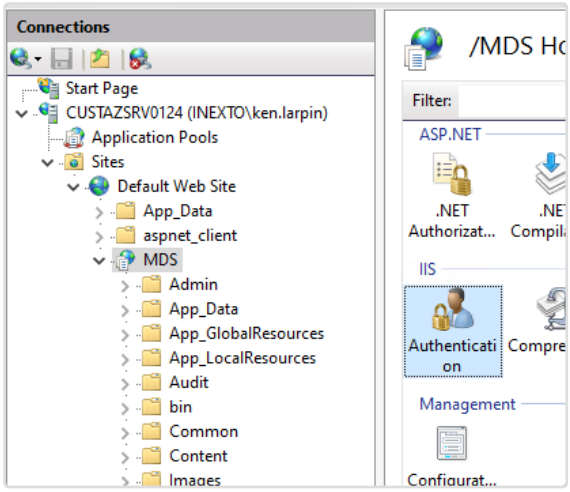


Figure 7.a

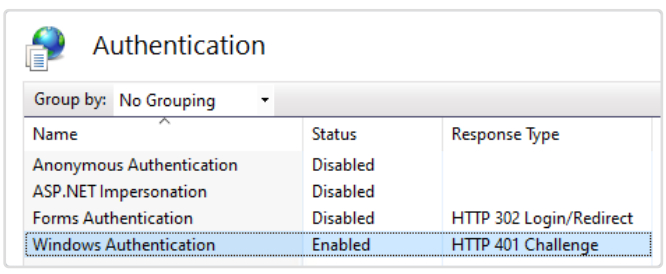


Figure 7.b

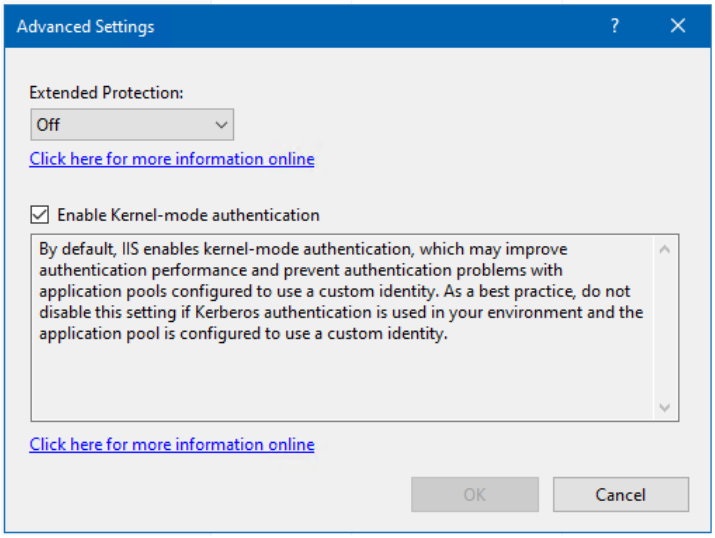


Figure 7.c

Figure 7: Enable Windows Authentication, but without the Extended Protection. This will allow to log to the MDS website with your Windows account.

Web.config

1. Now we need to configure the MDS web.config file.
2. Go to C:\Program Files\Microsoft SQL Server\160\Master Data Services\WebApplication
3. Copy the file web.config on the desktop.
4. Edit the copy and change the bindings and services with the code below.
5. Save the file. Close it. Finally, copy it and paste it back to its original folder.

```
<bindings>
  <wsHttpBinding>
    <binding name="mdsWsHttpBinding" maxBufferPoolSize="2147483647" maxReceivedMessageSize="2147483647">
      <readerQuotas maxArrayLength="2147483647" maxBytesPerRead="2147483647" maxDepth="2147483647" maxNameTableCharCount="2147483647" maxStringContentLength="2147483647" />
      <security mode="Transport">
        <message clientCredentialType="Windows" />
      </security>
    </binding>
  </wsHttpBinding>
  <basicHttpBinding>
    <binding name="mdsBasicHttpBinding" maxBufferPoolSize="2147483647" maxReceivedMessageSize="2147483647">
      <readerQuotas maxArrayLength="2147483647" maxBytesPerRead="2147483647" maxDepth="2147483647" maxNameTableCharCount="2147483647" maxStringContentLength="2147483647" />
      <security mode="Transport">
        <transport clientCredentialType="Windows" />
      </security>
    </binding>
    <binding name="importerBasicHttpBinding" maxBufferPoolSize="2147483647" maxReceivedMessageSize="2147483647">
      <readerQuotas maxArrayLength="2147483647" maxBytesPerRead="2147483647" maxDepth="2147483647" maxNameTableCharCount="2147483647" maxStringContentLength="2147483647" />
      <security mode="TransportCredentialOnly">
        <transport clientCredentialType="Windows" />
      </security>
    </binding>
  </basicHttpBinding>
```

```
</bindings>
<services>
  <service behaviorConfiguration="mdsWsHttpBehavior" name="Microsoft.MasterDataServices.Services.Service">
    <endpoint binding="wsHttpBinding" bindingConfiguration="mdsWsHttpBinding"
bindingNamespace="http://schemas.microsoft.com/sqlserver/masterdataservices/2009/09"
contract="Microsoft.MasterDataServices.Services.ServiceContracts.IService" />
    <endpoint address="bhb" binding="basicHttpBinding" bindingConfiguration="mdsBasicHttpBinding"
bindingNamespace="http://schemas.microsoft.com/sqlserver/masterdataservices/2009/09"
contract="Microsoft.MasterDataServices.Services.ServiceContracts.IService" />
    <endpoint address="mdsimporter" binding="basicHttpBinding" bindingConfiguration="importerBasicHttpBinding"
bindingNamespace="http://schemas.microsoft.com/sqlserver/masterdataservices/2009/09"
contract="Microsoft.MasterDataServices.Services.ServiceContracts.IService" />
  </service>
  <service behaviorConfiguration="mdsWsHttpBehavior" name="MDS1">
    <endpoint binding="wsHttpBinding" bindingConfiguration="mdsWsHttpBinding"
bindingNamespace="http://schemas.microsoft.com/sqlserver/masterdataservices/2009/09"
contract="Microsoft.MasterDataServices.Services.ServiceContracts.IService" />
    <endpoint address="bhb" binding="basicHttpBinding" bindingConfiguration="mdsBasicHttpBinding"
bindingNamespace="http://schemas.microsoft.com/sqlserver/masterdataservices/2009/09"
contract="Microsoft.MasterDataServices.Services.ServiceContracts.IService" />
    <endpoint address="mdsimporter" binding="basicHttpBinding" bindingConfiguration="importerBasicHttpBinding"
bindingNamespace="http://schemas.microsoft.com/sqlserver/masterdataservices/2009/09"
contract="Microsoft.MasterDataServices.Services.ServiceContracts.IService" />
  </service>
</services>
```

Code 3: WCF configuration. mdsWsHttpBinding is the HTTPS binding for the website. mdsBasicHttpBinding is the HTTPS binding for the Excel plugin. importerBasicHttpBinding is the HTTP binding for the MDS Importer (locally on the same VM as MDS).

Try to browse from your own PC.

- *Website: https://{DNS name}/MDS*
- *API: https://{DNS name}/MDS/service/service.svc*

Configure the model

In MDS, the model is equivalent to an SQL database. If you need a brand new master data model, you can create a model and add entities according to your needs. If you are re-using a model, you can follow the next chapters about export and import.

Export model

1. On the MDS website, go to System Administration.
2. In the ribbon, select Deployment.
3. Choose Create.
4. Follow the wizard to create a package file.

This package contains the structure of the model, but not the data. (For backup purpose, use SQL Server backup functionality.)

Import model

1. On the MDS website, go to System Administration.
2. In the ribbon, select Deployment.

3. Choose Deploy.
4. Choose the package file created in the previous chapter.

User setup

Setup process that the users will need to do.

First, the user needs to have an account in the Inexto Active Directory.

Second, the user will need to be added in a group with MDS rights or be added as an individual MDS user. An administrator can perform this from the MDS website's "User and Group Permissions".

Website

To access the website, browse: *https://{DNS name}/MDS*

You will be prompted to enter your credentials. It's your email address and Windows password.

Excel add in

To use the Excel Add In, you must first install it. In the MDS website, you will find a link to download the MSI.

Once installed, a Master Data tab will be present in Excel's ribbon. You can manually connect to MDS from Excel or download a connection file from MDS website.

Since Excel can't ask for your credentials, you need to enter them in the Windows Credential Manager.

1. In the Windows search bar, enter "Credential Manager".
2. Select Windows Credentials
3. Select "Add a Windows credential"
4. Enter the DNS for the internet or network address and your Windows login.

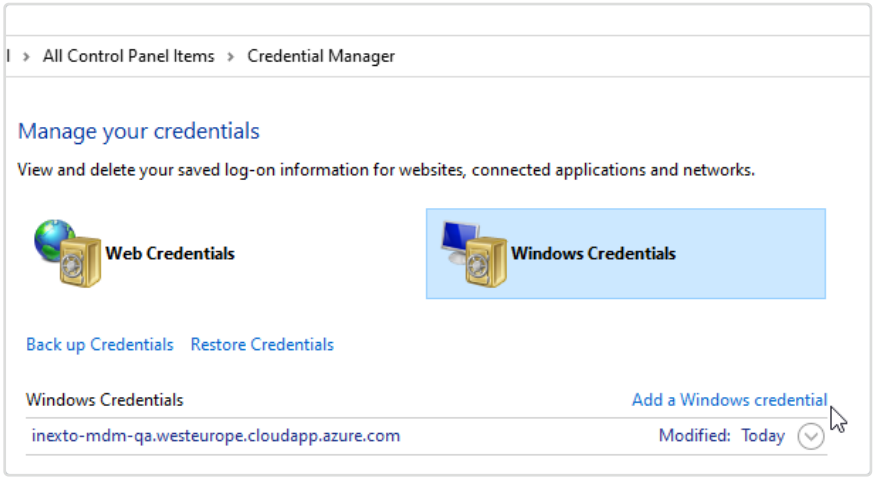


Figure 8: Add credentials that will be used by Excel Add In.

Do not remove the Add In from Excel's option. It's near impossible to re-install it.

Windows Administrators: Gate

Gate Installation

Installation of the Gate Administration

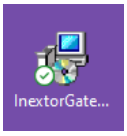
Installation of the Gate Administration without the Gate Service is possible. However, the Gate Admin still requires a Gate service in order to run.

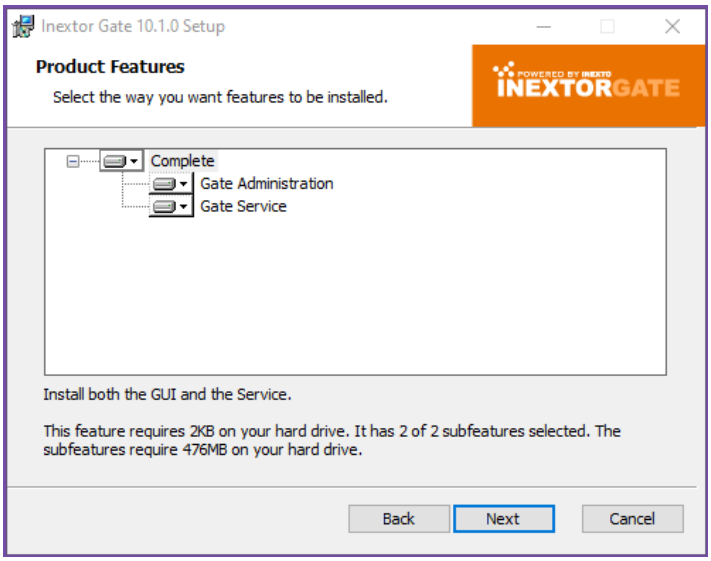
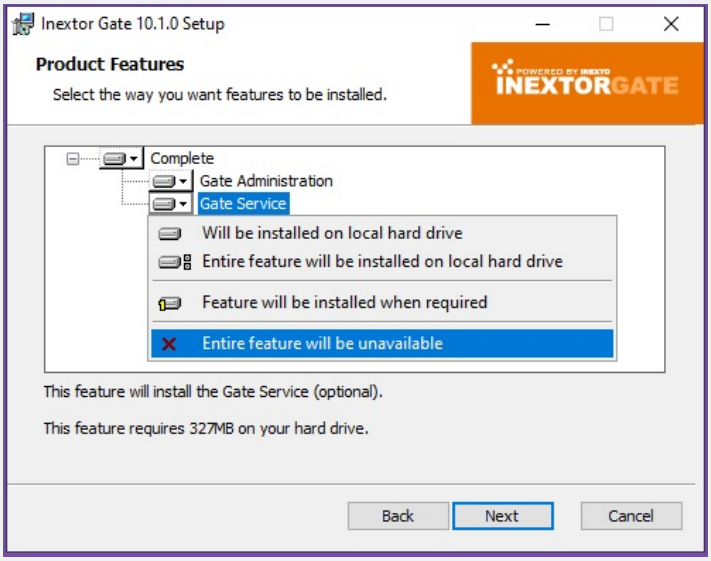
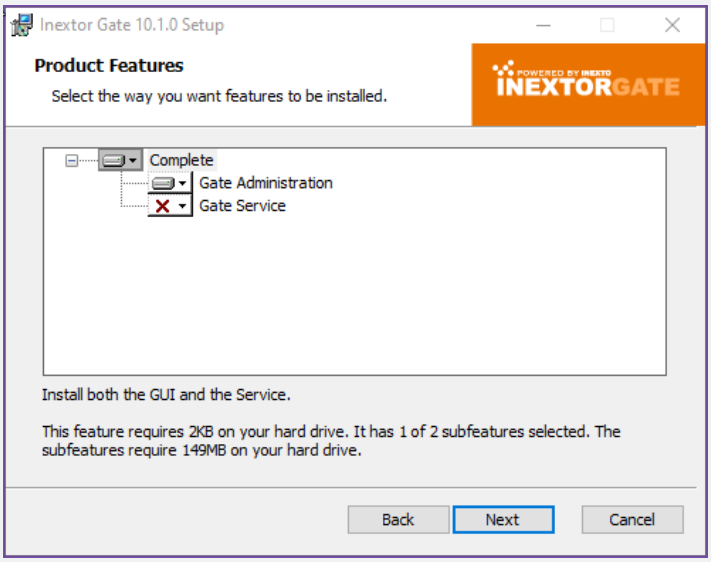
In case of an upgrade of an existing Inextor installation, you can either perform a side-by-side installation to maintain both versions or do a complete upgrade (preferred). If you choose to perform a side-by-side installation, refer to GateService.exe.config for the required configuration settings.

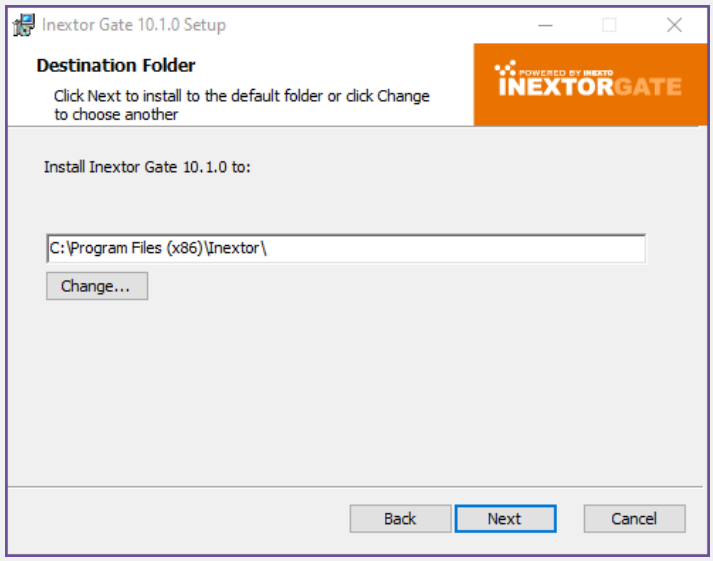
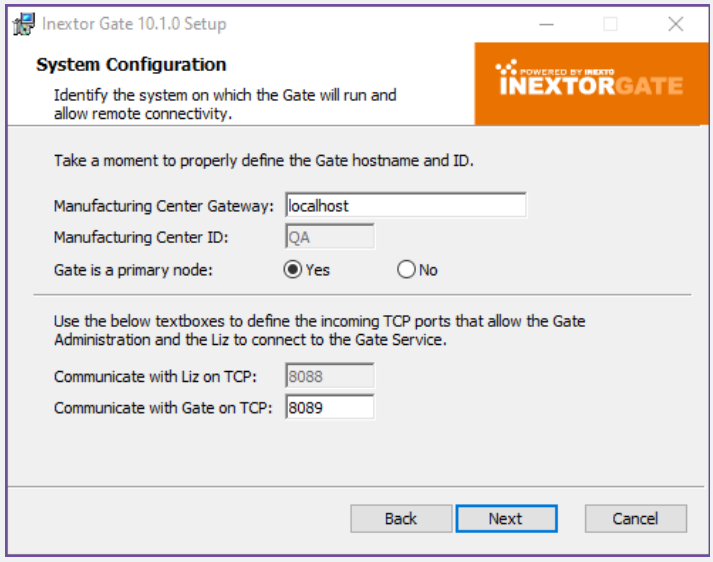
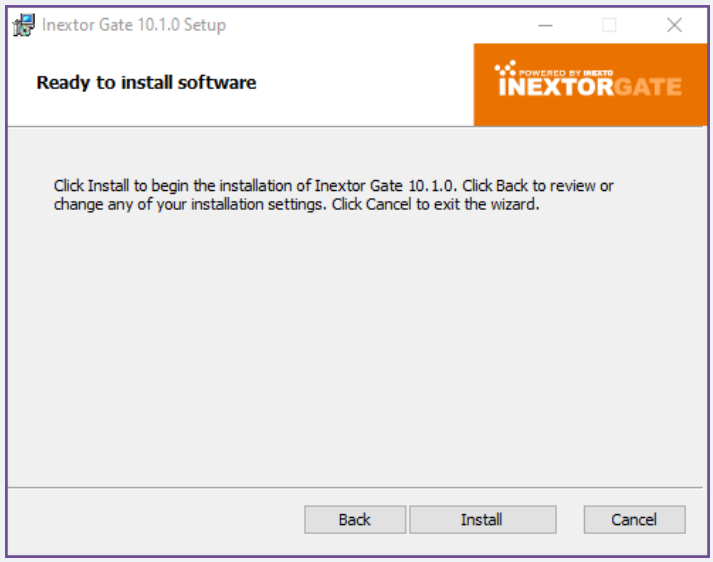
To install the **Gate Admin** and **Gate Service** follow steps 1-3, 5-14 (skip 4)

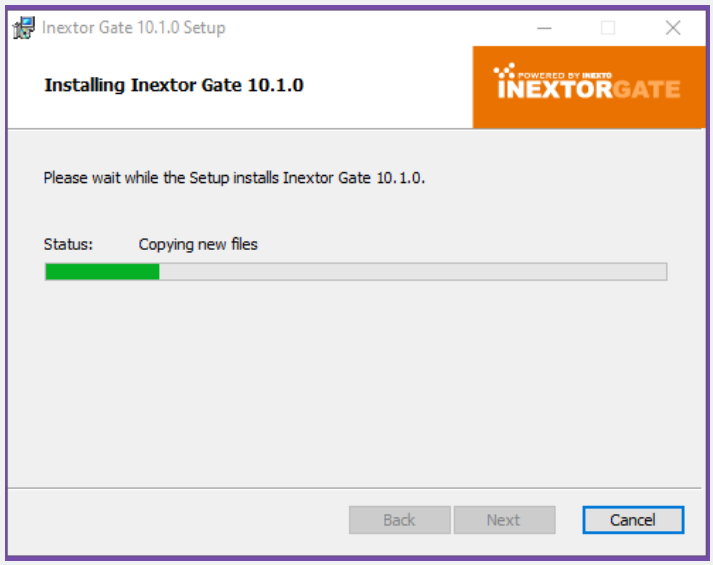
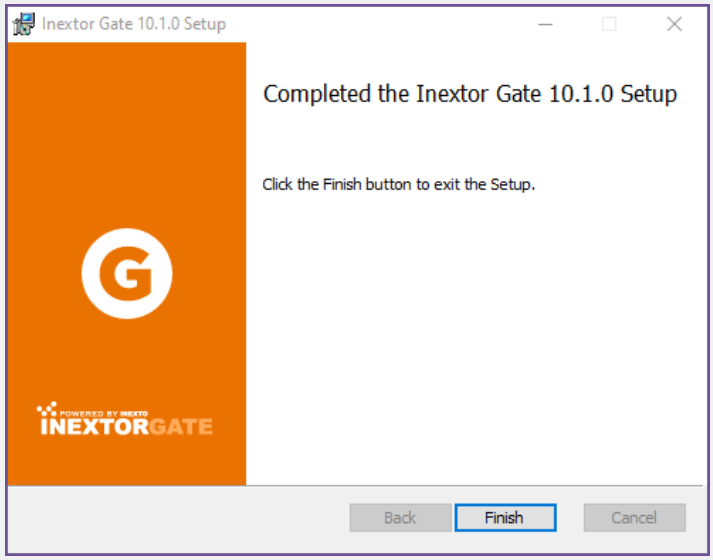
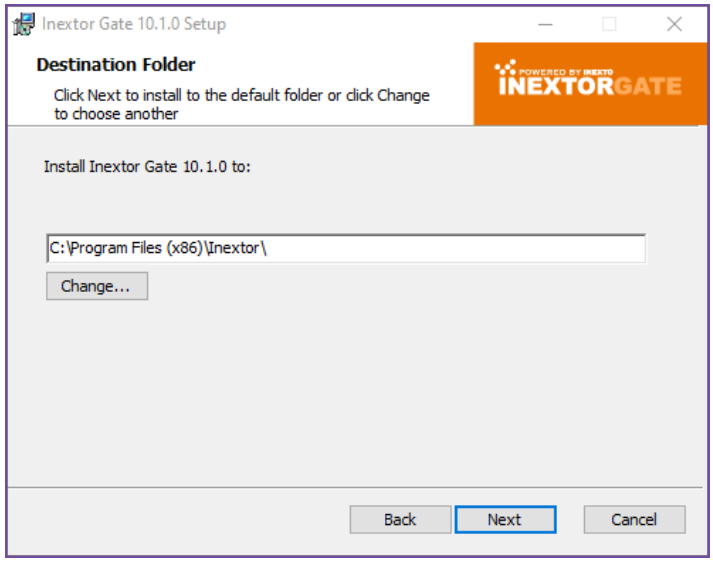
> Starting from version 2907, the log configuration files of the Gate service are no longer overwritten during an upgrade.

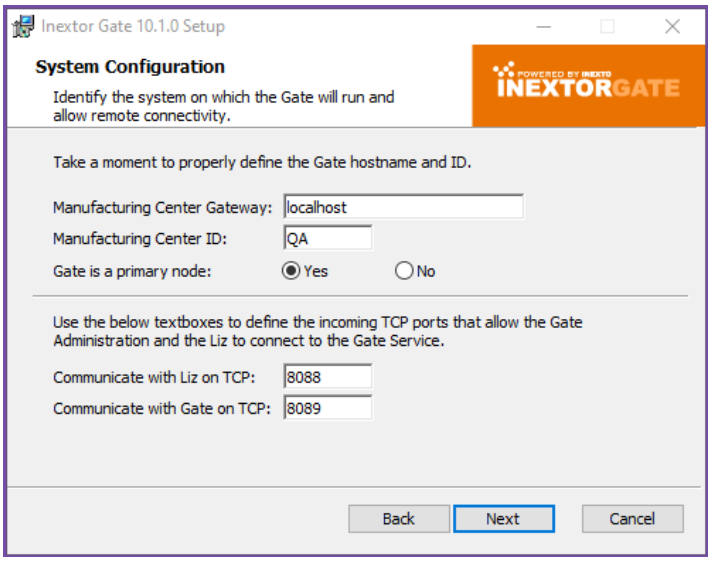
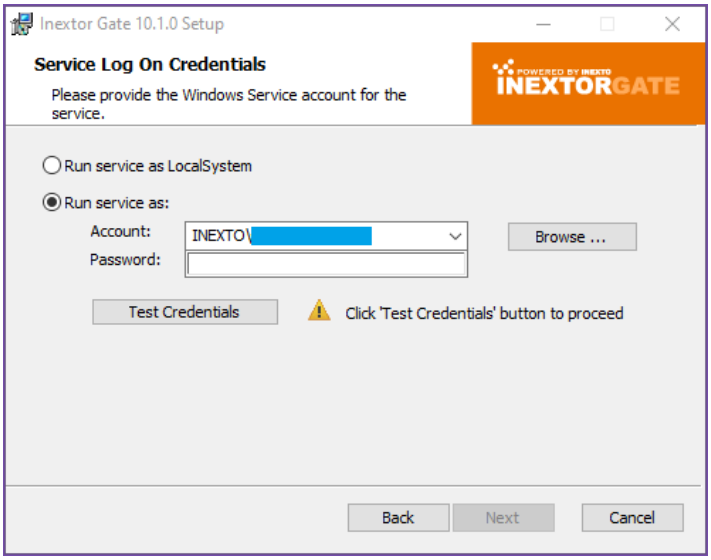
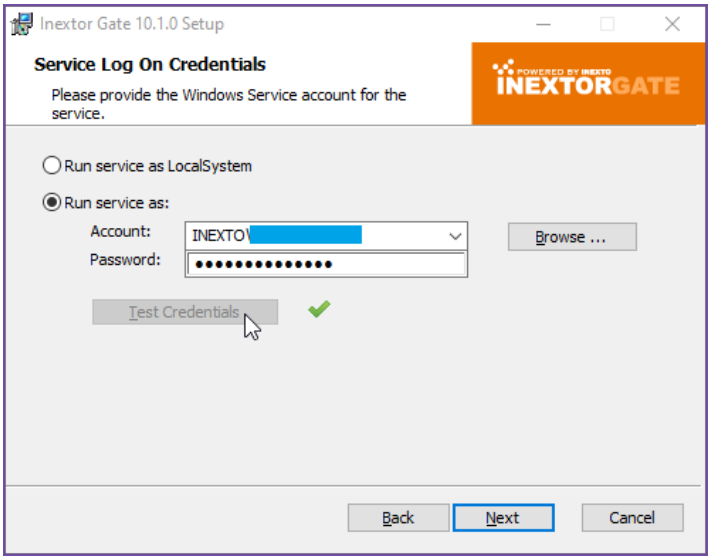
To install only the **Gate Admin** follow steps 1-4.

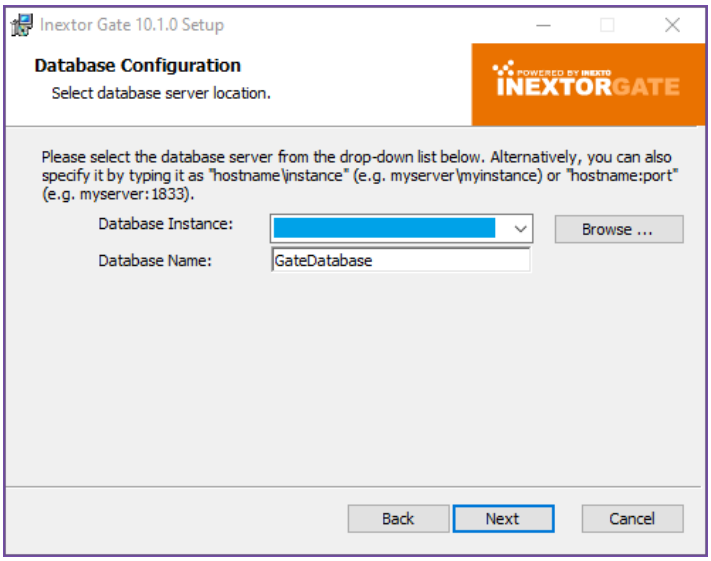
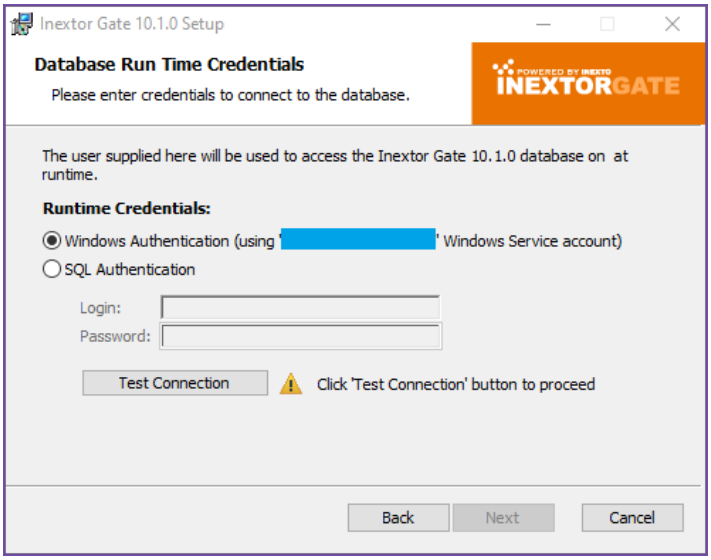
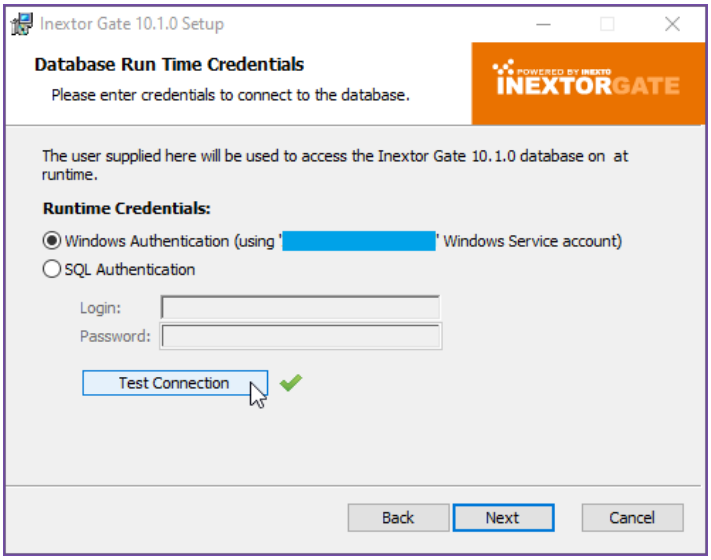
Step	Description	Screenshots
1	The installation file is named **InextorGate-10.1.0.XXXX.msi** where **XXXX** is the build number of the MSI package. If this file is not available for you, please ask the Support Desk. • Click **Next** to being the installation process	
2	In this step, you can decide to proceed or cancel the installation process. • Click **Next** to continue with the installation In case of an existing Inextor installation the system will propose to do: • either a side by side installation or a complete upgrade if your installation has a different version • or repair or remove the installation if the version is the same **Gate Service upgrade** Prior to doing the upgrade, you will need to stop Gate service (on each server in case of multiple Gate servers running).	

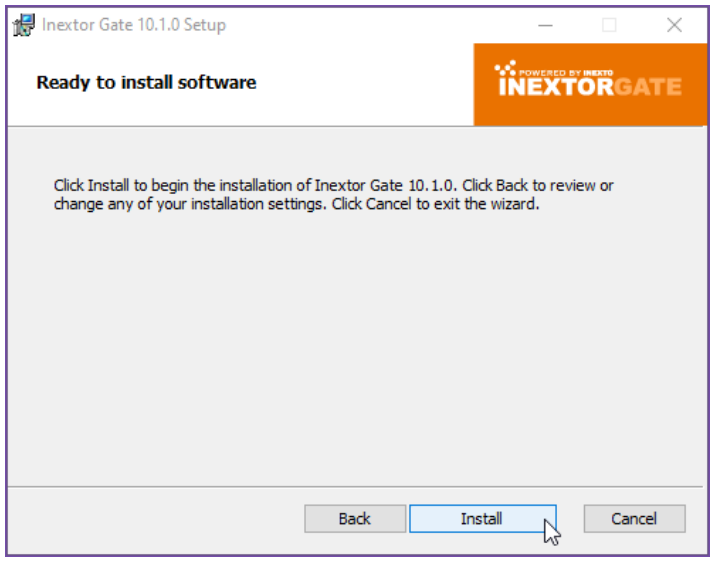
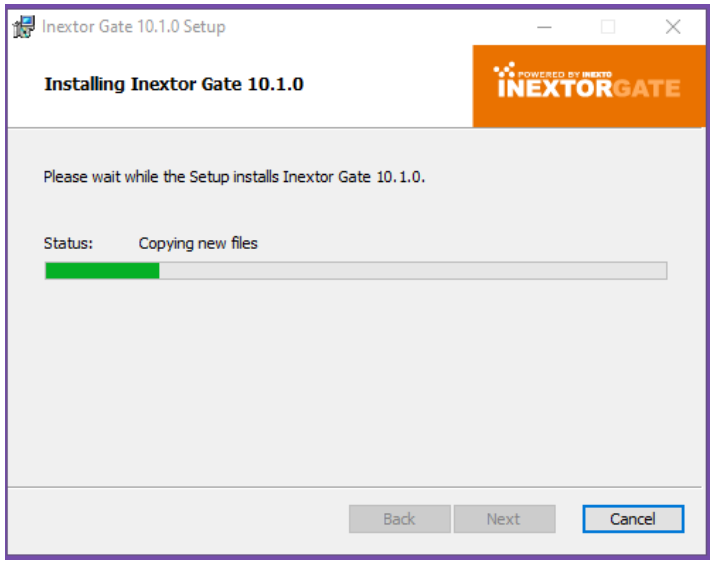
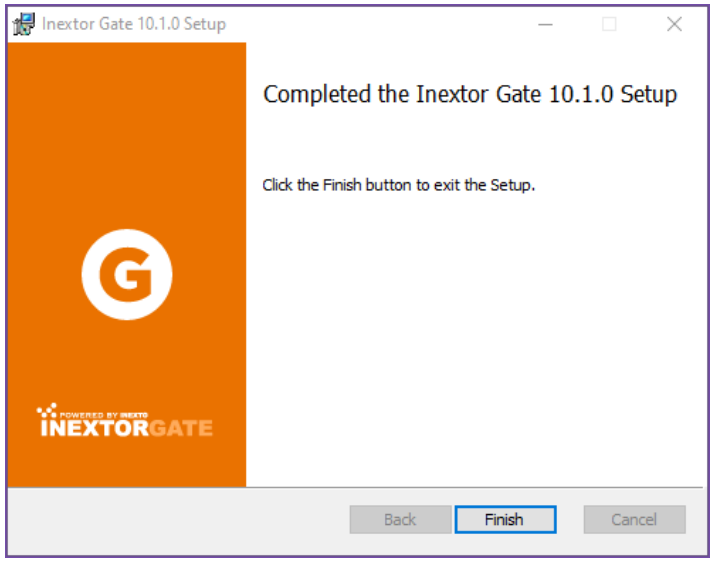
3	In this step, you can decide to install the Gate Administration, the Gate Service or both • Click **Next** to install both	
4A	If only the Gate Administration is required, disable the Gate Service • Click **Next** to install only the Gate Administration	
4B	The Gate Service has been excluded from the setup • Click **Next** to proceed	

4C	Select the folder to install the Gate Administration • Click **Next** to continue the installation	
4D	Change the host and port details or use the default (recommended) values for communication with the Gate Service and the Liz Service. Refer to the Gate network: Required firewall settings page for the ports description and recommended values used. If you are installing a primary and secondary Gate Servers, select which one here. If you do not plan to have a secondary server, just keep the 'Yes' radio button checked. Refer to the KB9 - How to Enable Load Balancing for 2 Gates to Provide High Availability page to choose the correct setting for the primary node • Click **Next** to continue **GPO on AD groupd** Having the GPO applied on the AD Groups will automatically open these ports for you.	
4E	Choose to go back or proceed with the installation • Click **Install** to start the installation process	

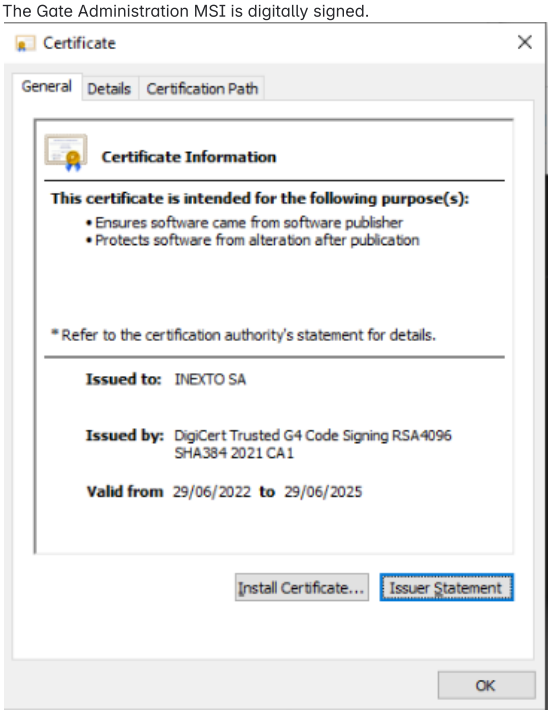
4F	Wait until the installation process is completed.	
4G	The Gate Administration has been installed successfully. • Click **Finish** to exit.	
5	Select the folder to install the Gate Application and Gate Service • Click **Next** to continue the installation	

6	Change the hostname, manufacturing Center ID, port, and Gate type or use the default (recommended) settings. Refer to the Gate network: Required firewall settings page for the ports description and recommended values used. If you are installing a primary and secondary Gate Servers, select which one here. If you do not plan to have a secondary server, just keep the 'Yes' radio button checked. Refer to the KB9 - How to Enable Load Balancing for 2 Gates to Provide High Availability page to choose the correct setting for the primary node • Click **Next** to use the default settings. **GPO on AD groupd** Having the GPO applied on the AD Groups will automatically open these ports for you.	
7	Enter the user account and password used to run the Gate Service • Fill in the Account and Password fields and click **Test Credentials** **Local system accounts configuration** Local System account only works for Gates that are connected using certificate to the Vault (only governmental Vaults or for TPM).	
8	Make sure the credential test is successful. • Click **Next** to continue.	

9	Enter the SQL Database Instance name and the Database Name to continue. Note: It's required to create the database before executing this step. • Click **Next** to continue.	
10	Select Windows Authentication (recommended) and check if the user can access the database. • Click on **Test Connection** to check the connection with the database	
11	Make sure the connection test is successful. • Click **Next** to proceed.	

12	The software is now ready to be installed • Click **Install** to begin the installation process.	
13	The installation process is underway, please wait for it to complete.	
14	The Gate software has been installed successfully. • Click **Finish** to exit.	
15	To assess the Gate installation, verify that the **Inextor Gate Service** is running with the account you specified during the setup. If the service failed to start, make sure that the account is set properly and that this account can actually access the SQL database.	

The Gate Administration MSI is digitally signed.



Install the Gate Client and Service in silent mode

The msi can be installed via command, without needing to use the installer UI.

When upgrading the Gate, whether you are using the msi in UI mode or silent command-line mode, any change you made to the Gate configuration files (GateService.exe.config and GateAdministration.exe.config in the installation folder) will not be preserved. If you have made changes to this file after installation, you need to:

- copy it to a different location before the upgrade
- perform the upgrade
- check for differences between your old configuration file and the new configuration file that was installed during the upgrade
- Apply any changes to the configuration file in the installation folder as needed, or copy your old file over the new one if no setting will be lost.

To do a silent installation/upgrade, use one of the commands below. The commands include sample values, which need to be changed to reflect your environment, including the name of the msi file and the installation location.

When upgrading the Liz using the silent msi installation command, you need to specify all the settings you would enter in the UI (described above) in the command, unless you know you are using the default value.

After the installation is complete, a log file will be generated (GateInstallLog.txt) which allows troubleshooting in case the installation failed.

Parameters	Description
DATABASE_SERVER1	Database server
DATABASE_NAME	Database name or catalog
SERVICE_LOGON_TYPE1	Service logon
SERVICE_USERNAME	Service username
SERVICE_PASSWORD	Service password of the user
ISPRIMARYNODE	If primary node or not
TCPLIZ	TCP port Liz
TCPGATE	TCP port Gate
INSTALLDIR	Directory of the installation
INSTALL_TYPE	Type of installation: ▪ Complete ▪ Client ▪ Service

Install client and service to run as LocalSystem:

Minimum command line (if all settings have default values):

LocalSystem Installation without Settings

```
msiexec.exe /qn /i InextorGate-[Tobacco]-10.1.0.0.msi DATABASE_SERVER1=ServerAlias DATABASE_NAME=GateDabaseName SER
```

All settings specified (can omit any that have default value):

LocalSystem Installation with Settings

```
msiexec.exe /qn /i InextorGate-[Tobacco]-10.1.0.0.msi DATABASE_SERVER1=ServerAlias DATABASE_NAME=GateDabaseName SER
```

Install client and service to run under a different account

Minimum command line (if all settings are as default):

Installation without Settings

```
msiexec.exe /qn /i InextorGate-[Tobacco]-10.1.0.0.msi DATABASE_SERVER1=ServerAlias DATABASE_NAME=GateDabaseName SER
```

All settings specified (can omit any that have default value):

Installation with Settings

```
msiexec.exe /qn /i InextorGate-[Tobacco]-10.1.0.0.msi DATABASE_SERVER1=ServerAlias DATABASE_NAME=GateDabaseName SER
```

Configuration Files

This section gives an overview of the configuration files for the Gate. They are described in greater detail in the following pages:

- Configuration - GateService.exe.config
- Configuration - GateAdministration.exe.config

Configuration files for Gate application

Configuration file	Description
GateService.exe.config	Configuration for the Gate application
ShiftConfiguration.xml	Configuration for the ShiftCode pluggin
ShiftCalendar.txt	Calendar used by the ShiftCode pluggin

Configuration files for Gate Administration GUI application

Configuration file	Description
GateAdministration.exe.Config	Configuration for the Gate application

Locations

Configuration file	Description
GateService.exe.config	C:\[DeploymentFolder]\Inextor\Gate 10.X.X\Service
ShiftConfiguration.xml andShiftCalendar.txt	C:\[DeploymentFolder]\Inextor\Gate 10.X.X\Service\ShiftCode
GateAdministration.exe.Config	C:\[DeploymentFolder]\ Inextor\Gate 10.X.X\Administration

Gate Administration

Configuration - GateAdministration.exe.config

The format for the file is defined by a schema for configuration files for based on .NET Framework (http://msdn.microsoft.com/en-us/library/1fk1t1t0.aspx).

Parameters in section “appSettings” are described below. The logging is configured by default in file Logging.Serilog.json. Normally the other section could be left with the default settings.

```
<configuration>
    ...
    <appSettings>
        <!-- Vault Gate Administration GUI Settings -->
    </appSettings>
    ...
</configuration>
```

The table below shows the keys available to be configured in the configuration file. The keys with a default value below should be left unchanged unless there is a specific need for another setting. The keys below with a note should always be considered and determined for your Gate Admin GUI specific needs.

Key	Default value	Description
HideAssemblyResolutionFailure	(XmlSerializers)| (resources)	Skips these exceptions, without trying to localize them. Resources are missing for these system dlls and that would cause infinite recursion in the EL4Net exception hook.
ProgramName	GateAdministrationExe	Name of the Program.
ProgramUnitName	GateAdministrationExe	Name of the Program unit built by this deployment.
ProgramVersion	10.1.0	Version of the Program for this deployment.
SolutionCompany	Inexto	Name of the Company providing this Solution.
SolutionName	Inextor	Name of the Solution.
TargetHost	localhost (also see note)	Defines the Name of the host to which the default connection refers.
TargetPort	8089 (also see note)	Defines the port used for the default connection. Must match the server's ListenPort token.

Note: The value of those keys are defined when installing the Gate Administration GUI.

The table below shows additional keys available to be configured in the configuration file.

Key	Default value	Description
ForceAzureADAuthentication	Empty	If not empty, must point to a Vault Web site, where the user will be asked to authenticate agains Azure AD. This supersedes the connection configuration performed in the user interface. The setting has priority over 'ForceWindowsAuthentication.' Note: only available from version 4600+
ForceWindowsAuthentication	Empty	If the value is "True", Windows authentication is used to authenticate with the Gate Server. This supersedes the connection configuration performed in the user interface. Note: only available from version 4600+

Microsoft Failover Clustering Gate

See page Microsoft Failover Clustering.

Configure a port on the Gate to be used by an Operation Driver

Some operation drivers might fail when trying to start them, with an error message stating that the Gate is not allowed to open a TCP port.

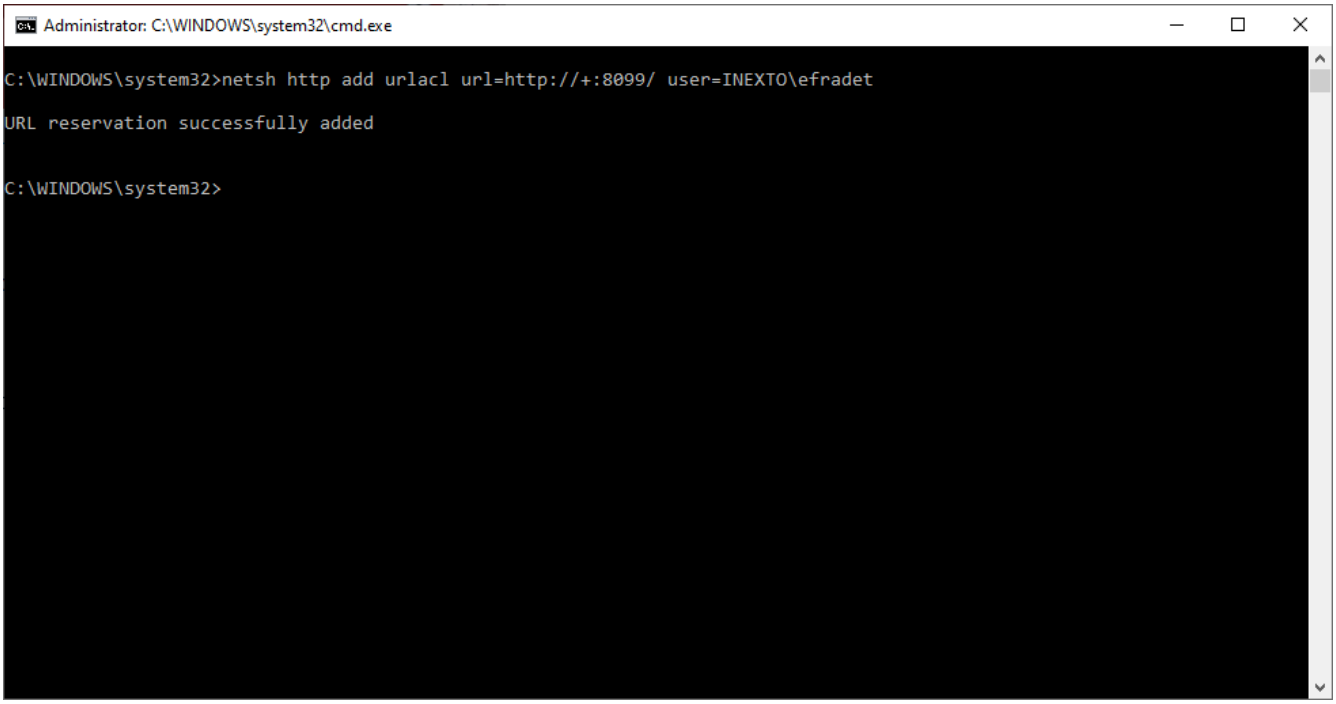
In such a case, you must use the built-in netsh utility to reserve a specified URL for non-administrator users and accounts.

To do it:

1. Open a command-line console as an administrator.
2. Execute the following command (replace **DOMAIN\username** by the Service Account which is running the Service and 8001 by the correct port you are trying to use):

NETSH Command

```
netsh http add urlacl url=http://+:8001/ user=DOMAIN\username
```

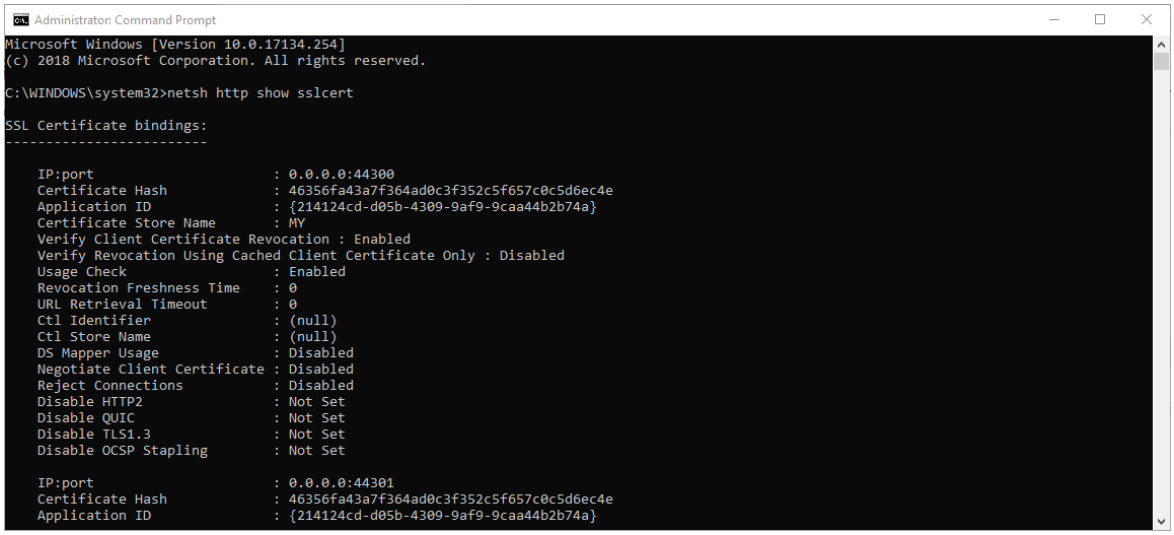


How to Register a Server Certificate for WCF Operation Drivers

You should open a Command Prompt in Admin mode and run the following commands :

```
netsh http show sslcert
```

This will give you a list of already present certificates :



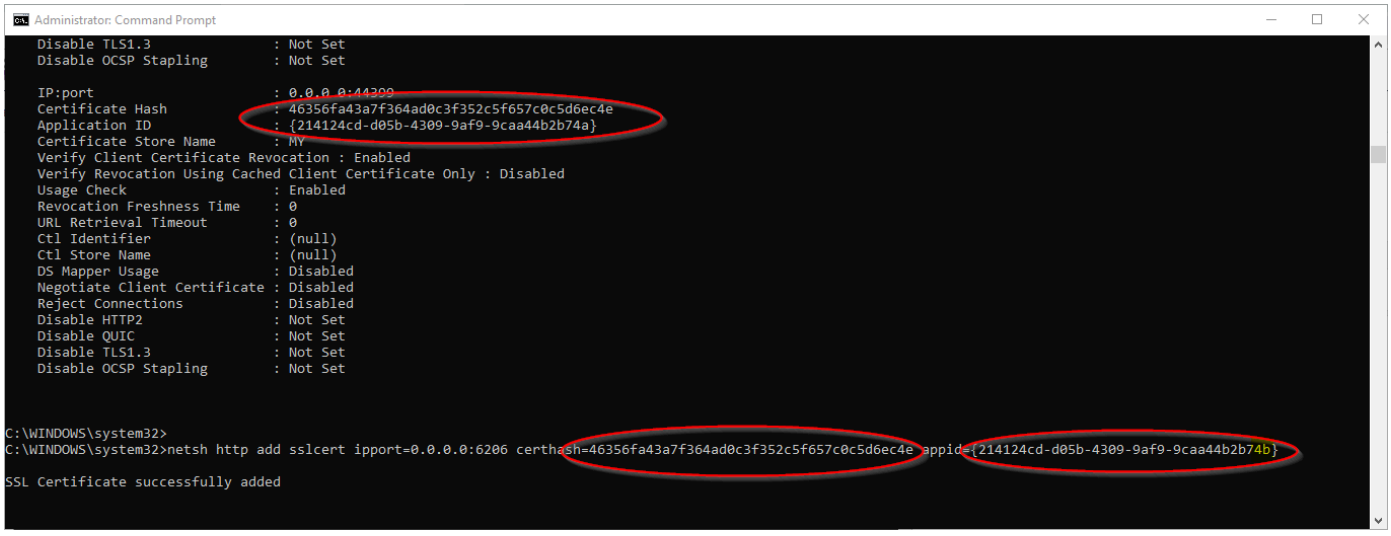
Then, you have to check if the port that you target is part of these certificates, and if not, add it using that command:

```
netsh http add sslcert ipport=0.0.0.0:6206 certhash=[HASH] appid=[GUID]
```

Where:

- 6206 is the port that you want to reach
- [HASH] can be used from an existing entry returned previously, here I took the hash from the last entry, ie 46356fa43a7f364ad0c3f352c5f657c0c5d6ec4e
- [GUID] is the guid of the application that is allowed; you can put any new Guid.

Here is the result :



Installing Report Files on the Gate Server

The Gate loads report files (.rdl) located under the path defined by the application setting *SsrsReportsPath*.

By default, this path points to the folder SSRS of the Gate Service installation path (example: C:\Program Files (x86)\Inextor\Gate 10.1.0\Service\SSRS).

The path can be changed to another location by modifying the setting *SsrsReportsPath* accordingly. The Gate service needs to be restarted for the change to take effect.

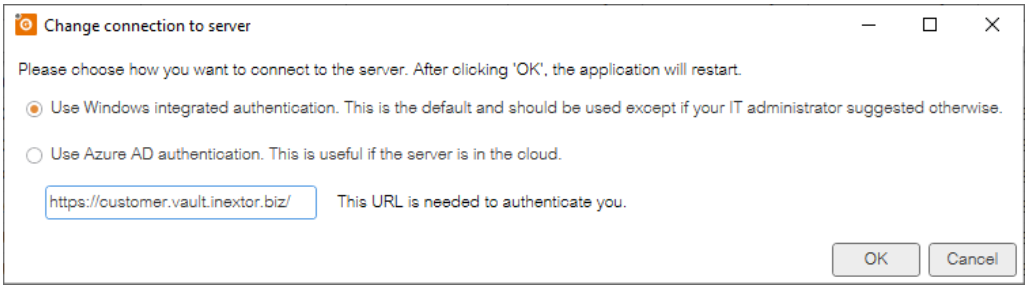
User Authentication Gate

By default, users launching the Gate administration program connect to the Gate server using the Windows integrated authentication mechanism (i.e. their Windows credentials are transparently re-used).

The list of Active Directory groups they are a member of is used to manage their permissions on the Gate.

If the Gate server is running on a machine that is not part of a Windows domain, this mechanism did not work. Starting from version 10.1.0.3350 of the Gate, this scenario is now supported.

The Gate administration program has a button called 'Change connection' in its information pane. Clicking it presents us with a configuration window:



Activating the first radio button is the default configuration, suitable for use in a Windows domain.

Activating the second radio button enables the use of Azure AD to authenticate the user. It is necessary to specify the URL of the Vault Web site, in order to perform this authentication. Typically, single sign-on will be enabled and the user is not prompted for credentials.

When configuring groups for Gate security (see page Gate Security UI: Gate Security), the group name needs to be entered without any domain name.

Gate Service

Configuration Files - GateService.exe.config

This page describes the settings in the GateService.exe.config file. The information is organized as follows:

- Configuration Files - GateService.exe.config
  - Configuration Parameters and Settings
    - General
    - Proxy Configuration
    - Lizs settings
    - MCID
    - Shift Code
  - Side-by-side Gate Installation

Configuration Parameters and Settings

General

The format for the file is defined by a schema for configuration files for application based on .NET Framework (http://msdn.microsoft.com/en-us/library/1fk1t1t0.aspx).

Parameters in section “appSettings” are described below. The logging is configured by default in file Logging.Serilog.json.

The table below shows the keys which are only available to be configured in the configuration file. The keys with a default value below should be left unchanged unless there is a specific need for another setting. The keys below with a note should always be considered and determined for your Gate specific needs.

> ⚠ All the other keys should be configured in the Applications Settings. It is still possible to update them in the configuration file of the Gate server, but such a direct change will be lost during upgrades.

Key	Default	Description
CryptoDatabaseSafeType	None	Specify the mechanism used to protect data store in database. • None: data is not encrypted DataProtect: encrypt symmetric key using Data Protection, and then use symmetric key to encrypt DB • SingleRsa: encrypt symmetric key using Rsa algorithm with one 'specific' key, and then use symmetric key to encrypt DB

		• DoubleRsa: encrypt symmetric key using Rsa algorithm with one 'specific' key and again for one 'global' key, then use symmetric key to encrypt DB • RsaDataProtect: encrypt symmetric key using Data Protection and again for one 'global' key (using rsa algorithm), then use symmetric key to encrypt DB.
DataSetValidation	./ExtendedFields/GateDataSetValidation.xml	Path to the file containing the validation rules for all input data types.
DBConnectionString		Connection string for accessing the Gate database.
GateAdmHost	localhost	Host name of GateAdministrationGUI service.
GatewayHost	localhost	Host name of Gate Service.
IsPrimaryNode (Available since 10.0.5)	true	Whether this server is the primary Gate. When the value is set to false (Secondary instance), the instance will be inactive, checking periodically if the primary is running correctly. If it detects that the primary node is not writing to the database, then it will wake up to do the automatic fail over. If the value is set to True and there is already a server running, the service won't start because we cannot have two primary nodes.
ListenPort	8089 (also see Gate Service#note)	Listen port of GateAdministrationGUI service.
ListenPortForLiz	8088 (also see Gate Service#note)	Listen port of Liz that Gate connect to.
MCID	Refer to MCID and to Gate Service#note	Manufacturing Center Identifier.
ProgramName	GateService	Name of the Program.
ProgramUnitName	GateService	Name of the Program unit built by this deployment.
ProgramVersion	10.0.0	Version of the Program for this deployment.
SolutionCompany	Inexto	Name of the Company providing this Solution.

SolutionName	Inextor	Name of the Solution.

Proxy Configuration

If a Proxy Server is used for accessing the Vault on Internet, you should define that in the Gate Administration program. In the Configuration Application Settings tab, filtering by "Proxy", you will find the same parameters from System.Net, see below.

Key	Default Value	Description
ProxyEnabled	FALSE	Specifies whether a web proxy is used.
ProxyAddress		Specifies the proxy URI to use.
ProxyScriptLocation		Specifies the location of the configuration script. Do not use the ProxyBypassOnLocal attribute with this attribute.
ProxyUseSystemDefault	FALSE	Specifies whether to use Internet Explorer proxy settings. If set to True, subsequent attributes will override Internet Explorer proxy settings.
ProxyBypassOnLocal	FALSE	Specifies whether the proxy is bypassed for local resources.
ProxyBypassArrayList		An array of regular expression strings which contains the URIs of the servers. (separated by comma ',')
ProxyUseDefaultCredentials	FALSE	Specifies whether the default credentials for this host are used to access the web proxy.
ProxyUserNameCredentials		User name of the custom credentials.
ProxyPasswordCredentials		Password of the custom credentials.
ProxyDomainCredentials		Domain of the custom credentials.

Normally, these parameters can be left with the default settings.

If the current proxy server configuration is done in the system.net section, as shown in the sample below:

```
<system.net>
    <defaultProxy useDefaultCredentials="true">
        <proxy usesystemdefault="False" proxyaddress="http://proxy.xx:8000"bypassonlocal="True"/>
        <bypasslist>
            <add address="[a-z]+\.app\.xx" />
```

```
        </bypasslist>
    </defaultProxy>
</system.net>
```

Then the Application Settings should be:

Key	Value
ProxyEnabled	True
ProxyAddress	http://proxy.xx:8000
ProxyUseSystemDefault	False
ProxyBypassOnLocal	True
ProxyBypassArrayList	[a-z]+\.app\.xx
ProxyUseDefaultCredentials	False

Proxy address and bypass list above are examples only.

Then remove the system.net section.

Lizs settings

Settings prefixed with "Lizes" will apply to all Liz services running in the factory, connected to the Gate.
The settings are synchronized at each **production start**.

MCID

MCID (Manufacturing Center ID) is a multiple character identifier of the Gate. The code is used on the Vault by the manufacturer to identity his Gates. Preferably the manufacturer should use a unique code (MCID) for every manufacturing site but it is not a prerequisite. However when creating the Gate on the Vault a Gate Code is defined, see Vault Reference - Creating Factories for creating Gates on the Vault. The Gate Code has to be unique within that specific Vault. That is:

- It is not possible to use the same Gate Code for two Gates on the Vault.
- It is possible, though not advisable to use the same MCID for two different manufacturing sites.

The value is defined in the GateIdentity table (the value in the config file is ignored)

Shift Code

This plugin implements a mechanism in order to extract from a calendar the shift information. The patterns are stored on the ShiftConfiguration.xml file and the base calendar on the file ShiftCalendar.txt.

It can be configured on the GateService.exe.config:

```
<!-- Shift Code Information -->
<add key="ShiftCodeClassName" value="Cvs.ShiftCode.ShiftCode, Cvs.ShiftCode"/><!--Defines the ShiftCode class to use the pattern must also include the AssemblyName-->
<add key="ShiftConfigurationFile" value="./ShiftCode/ShiftConfiguration.xml"/><!--Relative path to the ShiftConfiguration fil-->
<add key="ShiftCalendarSource" value="File"/><!--Source for the Calendar file generated by the Gate (None|Database|File).-->
```

```
<add key="ShiftCalendarFile" value="./ShiftCode/ShiftCalendar.txt"/><!--Define the file to search as source for Gate shift calendar. It is possible to include the MachineID using {0} in the pattern.-->
```

Side-by-side Gate Installation

For major upgrades, we make a side-by-side deployment of the Gate service, so that old Lizs and new Lizs connect to two different Gate services and Gate databases. This deployment has a direct impact on the Ranger that calls the AitService of the Gate.

To enable it, activate the following key in the newer GateService.exe.config file:

```
<add key="SideBySideGateEnabled" value="true"/>
```

and replace localhost:8099 by the address of the old Gate.

```
 <!-- This section is used only if the Side by side Gate is enabled (SideBySideGateEnabled=true) (normally during the migration process)-->
  <system.serviceModel>
    <bindings>
      <basicHttpBinding>

        <binding name="BasicHttpBinding_IDeserializationTTService"
         maxBufferSize="2147483647"
         maxBufferPoolSize="2147483647"
         maxReceivedMessageSize="2147483647">
          <readerQuotas maxDepth="2147483647"
                        maxStringContentLength="2147483647"
                        maxArrayLength="2147483647"
                        maxBytesPerRead="2147483647"
                        maxNameTableCharCount="2147483647" />
        </binding>
        <binding name="BasicHttpBinding_IProductionTTService"
         maxBufferSize="2147483647"
         maxBufferPoolSize="2147483647"
         maxReceivedMessageSize="2147483647">
          <readerQuotas maxDepth="2147483647"
                        maxStringContentLength="2147483647"
                        maxArrayLength="2147483647"
                        maxBytesPerRead="2147483647"
                        maxNameTableCharCount="2147483647" />
        </binding>
        <binding  name="BasicHttpBinding_ISerializationTTService"
         maxBufferSize="2147483647"
         maxBufferPoolSize="2147483647"
         maxReceivedMessageSize="2147483647">
          <readerQuotas maxDepth="2147483647"
                        maxStringContentLength="2147483647"
                        maxArrayLength="2147483647"
                        maxBytesPerRead="2147483647"
                        maxNameTableCharCount="2147483647" />
        </binding>
      </basicHttpBinding>
    </bindings>
    <client>
      <endpoint address="http://localhost:8099/Inextor.S3r.Gate.Service/SerializationTTService"
```

```
          binding="basicHttpBinding" bindingConfiguration="BasicHttpBinding_ISerializationTTService"
          contract="AIT.Inextor.SerializationTrackingTracing.ISerializationTTService"
          name="BasicHttpBinding_ISerializationTTService" />
      <endpoint address="http://localhost:8099/Inextor.S3r.Gate.Service/DeserializationTTService"
          binding="basicHttpBinding" bindingConfiguration="BasicHttpBinding_IDeserializationTTService"
          contract="AIT.Inextor.SerializationTrackingTracing.IDeserializationTTService"
          name="BasicHttpBinding_IDeserializationTTService" />
      <endpoint address="http://localhost:8099/Inextor.S3r.Gate.Service/ProductionTTService"
          binding="basicHttpBinding" bindingConfiguration="BasicHttpBinding_IProductionTTService"
          contract="AIT.Inextor.SerializationTrackingTracing.IProductionTTService"
          name="BasicHttpBinding_IProductionTTService" />
    </client>
  </system.serviceModel>
```

Windows Administrators: Vault

Windows Service

Table of Contents

- Introduction
- Stop and uninstall current service
  - Stop service
  - Uninstall script
  - Uninstall service
- Deployment folder
- VaultService.exe.config
  - Not using SQL Server mirroring with witness
  - Using SQL Server mirroring with witness
  - Vault Service listen port
  - Log folder
  - Audit log folder
- Vault Certificates
- Data protection
- Installing the service
- Performance and startup issues
  - Setting the key generatePublisherEvidence
  - Register the Enterprise Libraries in the GAC
  - Setting the timeout for start of services

Introduction

The table below describes which sections to consider when performing a new installation, upgrade or disaster recovery.

Sections	Description
3, 4, 5, 6, 7	New Installation
2, 3, 4, 7	Upgrade
3, 5, 6, 7	Disaster recovery

In case of performing an upgrade, always keep the current running system (folder C:\ VaultVxxxx) and install in a new folder. In this way you can revert back in case of any unexpected issue.

Stop and uninstall current service

Stop service

To stop the service you will have to follow these steps:

- Open the services.msc GUI from the Run option on the Start menu or go to Administrative Tools and choose Services.
- Locate the service "Inextor Vault X.X.XXXX" where the "X" indicates the version of the Vault.
- Right click the service and choose Stop.

Uninstall script

Copy the uninstall.bat file from the installation media to the folder C:\VaultVXXXX\Vaultservice Edit the file and change the path to the exe-file **VaultService.exe**.

Example of the content in uninstall.bat

```
"C:\Windows\Microsoft.NET\Framework\v4.0.30319\InstallUtil.exe" /u ".\Vaultservice\VaultService.exe"
```

Uninstall service

To uninstall the service you will have to follow these steps:

- Open the Inextor Installation folder that you used to install the Vault Service, usually located in "C:\[DeploymentFolder]\VaultVXXXx".
- Execute the "uninstall.bat" file.
- Once the program is finished you will have to check that the service is now uninstalled. To check this you will have to repeat steps 1 and 2 from the Stop service section.

Deployment folder

Create a folder, name the folder InextorX, where X is the version.

Copy the folder Vaultservice from the installation media (or from backup ONLY in case of disaster recovery) into the folder InextorX.

VaultService.exe.config

The configuration files should already have been prepared for you, however you might have to check the settings described in this section. For a full reference on the VaultService.exe.config file and other configuration files, refer to VaultService.exe.config and Business Configuration Parameters. For editing the VaultService.exe.config file you have to follows these steps:

- Locate the configuration file usually located on "C:\[DeploymentFolder]\Vaultservice\VaultService.exe.config".
- Right click the file and click on Edit option (e.g. use Notepad).

Not using SQL Server mirroring with witness

For the database connection you have to define the connection string if not already done for you:

```
"Data Source=<DatabaseServerName>(\<InstanceName>);
Initial Catalog=<DatabaseName>;
Integrated Security=True;
Application Name=S3rVault"
```

Using SQL Server mirroring with witness

For the database connection you have to add the partner to the connection string if not already done for you:

```
"Data Source=<DatabaseServerName>(\<InstanceName>);
Failover Partner=<MirrorDatabaseServerName>(\<InstanceName>);
Initial Catalog=<DatabaseName>;
Integrated Security=True;
Application Name= S3rVault"
```

Vault Service listen port

```
Listen Port is set to 8086 by default. If another port is desired, change it here:
```

```
<addkey="ListenPort" value="8086" />
```

Log folder

Logging is configured in an external file which location is defined by the following token:

```
<add key="SerilogConfig" value="Logging.Serilog.json" />
```

Audit log folder

Auditing is configured in the same external file as logging.

If Audit logging is enabled, three log files are produced (see Audit Configuration).

For more information on enabling audit logging and modifying audit configuration settings, refer to Audit Configuration.

> **Note**: Unless necessary, we recommend deactivating audit due to the large size of the generated logs.

Vault Certificates

The Vault certificate and depending certificates (Root/TP) must be installed in Windows Certification Store under "Local Computer" account, see Installing Certificates in Windows Certification Store.

Data protection

The keys used by the Vault NT service to encrypt and decrypt certificates stored in the database cannot be in clear text in the configuration file for the Vault NT Service, hence the keys must be encrypted. To create keys for the Vault service, see Create Keys for Data Protection. There might be additional elements that need protection in the configuration file of the Vault NT service, e.g. authentication information; see Configuration Protection Tool.

> Note: In case of disaster recovery where the Vault NT Service Server is replaced, the data protection actions described in this section must be redone.

Installing the service

In folder [Drive]:\InextorX\Vaultservice, execute the file Install.bat.

Example of the content in install.bat

```
"C:\WINDOWS\Microsoft.NET\Framework\v4.0.30319\InstallUtil.exe" /i ".\VaultService.exe"
```

You will be asked for username and password during the installation. These are the credentials for the user account under which the Vault Service should run.

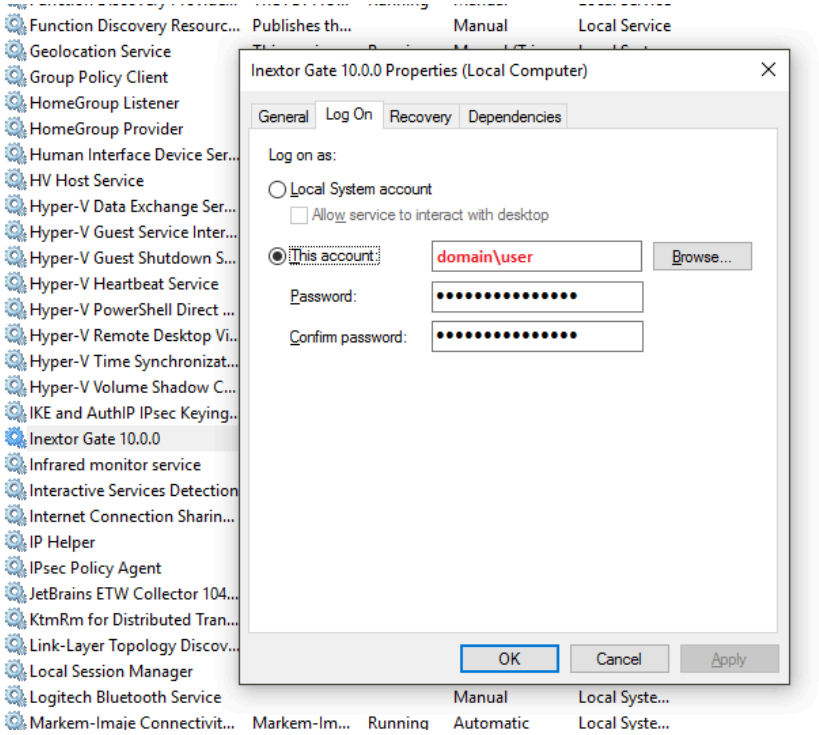
Requirements for the account that the service should run under

For using Windows Integrated security, the Vault service should run under service account (not local system account).

The account should be local administrator on the PC and optionally be a member in the domain. However the account does not have to be a domain administrator.

> Note: If the account is not local administrator on the PC, then you need to verify that the user account has adequate rights.
>
> - Bypass traverse checking
> - Increase a process working set

From the "control panel", start "administrative tools" and "services". Double click on the service and select the "Log On" tab. Verify the log on properties for the service.



Service Log On

Start the service. In case the service does not start, or starts but automatically closes, see troubleshooting section in the document Administration Performance Monitoring

Preferably, the service is configured to be started automatically and not manually (see note below).

To do this, perform the following steps:

- Open the services.msc GUI from the Run option on the Start menu or go to Administrative Tools and choose Services.
- Locate the service "Inextor Vault X.X.XXXX" where the "X" indicates the version of the Vault.
- Right click the service and choose Properties.
- On the General tab locate the Startup type drop down box.
- Select Automatic.
- Click Apply and then OK button.

> Note: If you have a secondary server and it is located on a disaster site, you might not want that service to be active (hence it should be set to manual), if you do not want any traffic to be directed to the disaster site during normal operation.

After the service has been started, open the Vault service log file and check that no errors exist in it (see document Administration Performance Monitoring for more information). To locate the Vault service log file you will have to perform the following steps:

- Locate the VaultService.exe.config file. "C:\[DeploymnentPath]\Vaultservice\VaultService.exe.config"
- Right click it and choose edit option.
- Check the path of the Serilog configuration file
- Check in Serilog configuration file where the logs are stored

Go to the path indicated on the configuration file and open it.

Performance and startup issues

If a server is not connected to the Internet, there could be problems with slow startup and potentially a timeout when starting the service. In the following sections there are some suggestions on how to improve the startup of the service:

Setting the key generatePublisherEvidence

This key prevents the OS from checking signed assemblies on the Internet, It is located under Configuration/runtime

```
<runtime>
    <generatePublisherEvidence enabled="false" />
</runtime>
```

Register the Enterprise Libraries in the GAC

This will further improve the startup time. Use gacutil to register the enterprise library dlls located in the Vaultservice folder. Gacutil /i [filename]

Setting the timeout for start of services

To further guarantee the startup of the program, the timeout for the service manager can be altered: Change the ServicesPipeTimeout to 120000 milliseconds. The procedure is as follows:

- Open registry database
- Navigate to HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control
- Locate the ServicesPipeTimeout value on the right pane. If it is not there, add a new DWORD value and name it ServicesPipeTimeout
- In the value field enter 120000 (decimal base)
- Close the Registry Editor down and Restart the Server machine.

Vault Web applications installation

Table of Contents

- Introduction
- Stopping IIS
- Software
- Configuration files
  - Vault Web site - Web.config
  - Web Services - Web.config
  - Vault Web Site - VaultConnection.configunit
  - Web Services - VaultConnection.configunit
- Changing Virtual Directory
-
  - Web Service Application (WS - Virtual Directory)
  - Web Service Application (Web - Virtual Directory)
  - Web Service Application (WebWin - Virtual Directory)
- Starting IIS

Introduction

Note: The Vault Website and Web Services can be installed on the same server as the Vault NT service or on separated servers.

The table below describes which sections to consider when performing a new installation, upgrade or disaster recovery.

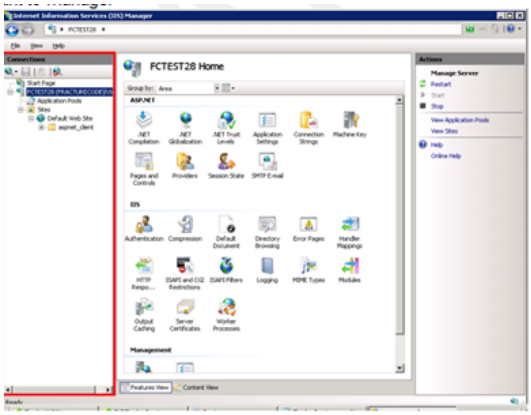
Sections	Description
3, 4	New Installation
2, 3, 4, 5, 6	Upgrade
3	Disaster recovery

In case of performing an upgrade, always keep the current running system (folder C:\InextorX for versions earlier than 10.0) and install in a new folder. In this way you can revert back in case of any unexpected issue.

Stopping IIS

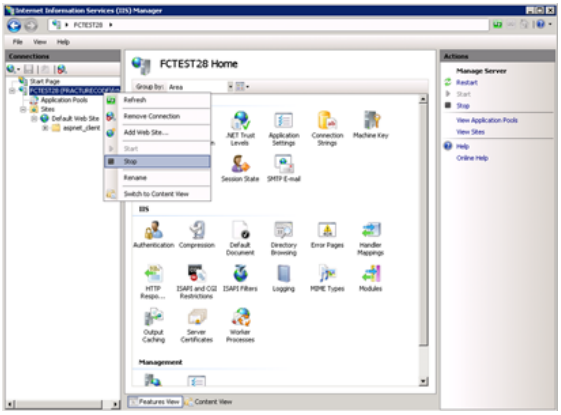
To stop the IIS, it is necessary to have administrative rights over the Web Site that is going to be stopped. Perform the following steps to stop the IIS:

- Open Administrative Tools and click Internet Information Services (IIS) Manager.
- On the left side of the screen there is a Menu showing the Connections, search the server that you want to manage.



Connections Menu

- Right-click the server and choose the Stop option.



Stop Server Option

Software

If you have not done so already in a previous installation step, create a folder, preferably at C:\, and name the folder InextorX where X is the major version.

Copy the folder **GateManagerWS**, **Inexto.S3r.Vault.Website2** and **VaultWS** from the installation media (or from backup ONLY in case of disaster recovery) into the folder InextorX.

Additionally you can deploy several web services depending on your need:

VaultWS

To deploy the Web Services that are used by other systems like iTrack and Ranger to decompose codes for instance, copy the folder VaultWS from the installation media (or from backup ONLY in case of disaster recovery) into the folder InextorX.

GateWS

To deploy the Web Service that the Factory uses to communicate with the Vault, copy the folder GateatewayWS from the installation media (or from backup ONLY in case of disaster recovery) into the folder InextorX.

Configuration files

The configuration files should already have been prepared for you, however you might have to check the settings described in this section. For a full reference for the configuration file and other configuration files, see Configuration Files - Vault Website and Configuration Files - Web Services.

Vault Web site – Web.config

In the configuration file C:\InextorX\VaultWebsite\Web.config :

- Enable the credentials providers that are to be used:

```
<!-- add credential providers -->
<Unit>
    CertificateCredentialProvider
</Unit>
<Unit>
    SSPICredentialProvider
</Unit>
<Unit>
    LoginPasswordCredentialProvider
</Unit>
```

Adding various credential providers enables the corresponding login methods to the Vault website, as follows:

- CertificateCredentialProvider: add it in order to enable logging in to the Vault website with client certificates
- SSPICredentialProvider: add it in order to enable logging in to the Vault website with Windows authentication
- LoginPasswordCredentialProvider: add it in order to enable logging in to the Vault website with a login and password defined by an administrator on the Vault website.

Web Services – Web.config

File: C:\InextorX\yyyyWS\Web.config Configure the file similarly to the website configuration, as described in Section Vault Web site - Web.config. For the GateatewayWS, the Credential Providers are specific to the Gate. The following ones should be used instead of their counterpart in the other services:

- For Certificate authentication: <Unit>GateCertificateCredentialProvider</Unit>
- For Windows Integrated authentication: <Unit>GateSSPICredentialProvider</Unit>

Vault Web Site – VaultConnection.configunit

File: C:\InextorX\VaultWebsite\bin\VaultConnection.configunit. If using load balancing, check that loadbalancedSessionAffinity is activated:

```
<!-- Activate the LoadBalancing if required -->
<UseOrOverride>
    LoadBalancedSessionAffinity,ConfigUnitVersion=1.0
</UseOrOverride>
If using load balancing, check that single server is disabled:
<!-- In case of single server -->
<!--
    <ComponentDefinition Name="eln.RemotingPublicationManager">
        <StaticComponent>
            <Implementation> <RemotingPublicationManagerConfig>
                <RemotingConnectionDefinition Name="DefaultRemotingConnection" OnMiss="Add" OnExist="Fail">
                    <PolicyName>TcpSecure</PolicyName>
                        <TcpConnectionParams ConnectionId = "0">
                            <HostName>localhost</HostName>
                            <Port>8086</Port>
                        </TcpConnectionParams>
                    </RemotingConnectionDefinition>
                </RemotingPublicationManagerConfig>
            </Implementation>
        </StaticComponent>
    </ComponentDefinition>
-->
```

If using load balancing, check that multiple server is enabled. Enter the name of the servers running the Vault NT Service and change port number if needed. You can add additional servers in the configuration if used.

```
<!-- In case of loadbalancing -->
<ComponentDefinition Name="eln.RemotingPublicationManager">
    <StaticComponent>
        <Implementation>
            <RemotingPublicationManagerConfig>
                <RemotingConnectionDefinition Name="DefaultRemotingConnection">
                    <TcpConnectionParams ConnectionId = "0" OnExist="Fail" OnMiss="Add">
                        <HostName>tstmx02.testing.local</HostName>
                        <Port>8086</Port>
                    </TcpConnectionParams>
                    <TcpConnectionParams ConnectionId = "1" OnExist="Fail" OnMiss="Add">
                        <HostName>tstmx04.testing.local</HostName>
                        <Port>8086</Port>
                    </TcpConnectionParams>
                </RemotingConnectionDefinition>
            </RemotingPublicationManagerConfig>
        </Implementation>
    </StaticComponent>
</ComponentDefinition>
```

Enable the provider(s) that the Web site uses and enter the priorities:

```

<!-- Define AutoLogin Connection Priorities -->
<ComponentDefinition Name="eln.ClientSecurityHelper">
    <SingletonComponent>
        <Implementation>
            <ClientSecurityHelperConfig>
                <AutoLoginCredentialProviders OnMiss="Add" OnExist="Fail">
                    <!--
                    <CredentialProvider Priority="1">
                        <ProviderName Name="Security.CertificateCredentialProvider"/>
                    </CredentialProvider>
                    <CredentialProvider Priority="2">
                        <ProviderName Name="Security.SSPICredentialProvider"/>
                    </CredentialProvider>
                    -->
                    <!-- Mexico Novell -->
                    <CredentialProvider Priority="1">
                        <ProviderName Name="Security.NovellCredentialProvider"/>
                    </CredentialProvider>
```

```
                </AutoLoginCredentialProviders>
            </ClientSecurityHelperConfig>
        </Implementation>
    </SingletonComponent>
</ComponentDefinition>
```

Web Services – VaultConnection.configunit

File: C:\InextorX\yyyyWS\bin\VaultConnection.configunit. Configure the file similarly to the website configuration, as described in Section Vault Web site - VaultConnection.configunit.

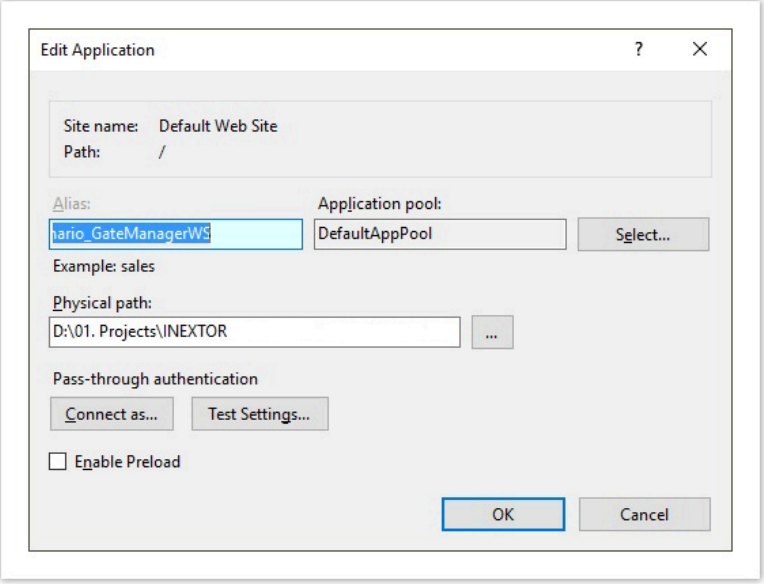
Changing Virtual Directory

In case of an upgrade, you might want to deploy the new version in another folder than the default one. In this case, you have to change Virtual Directory for the ASP.NET Applications. This can be done with the following procedure, valid on both Windows 2008 and 2012 server (***Note**: as of January 14, 2020, Windows 2008 is obsolete)*

Web Service Application (WS - Virtual Directory)

Perform the following steps to change the virtual directory.

- Open "Control Panel", "Administrative Tools" and "Internet Information Services".
- Click on the WS directory.
- Click on "Basic Settings..." in the menu to the right.
- Click on "..." to browse and search the new folder of the updated application.
- Click OK.

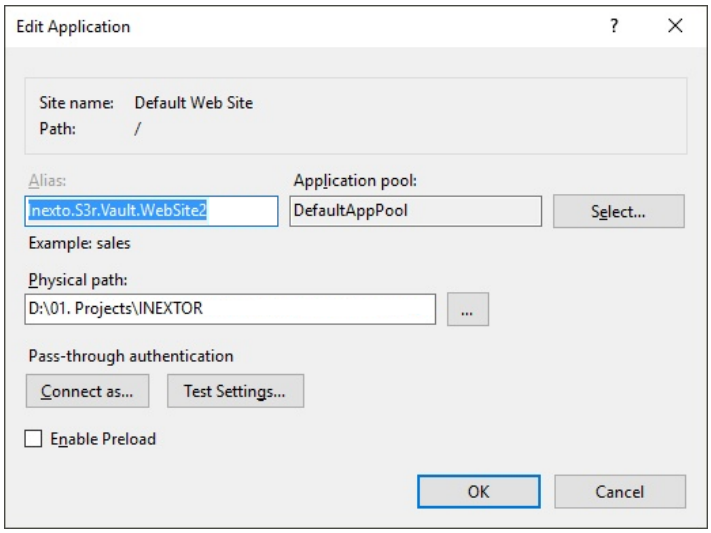


Changing Virtual Directory

Web Service Application (Web - Virtual Directory)

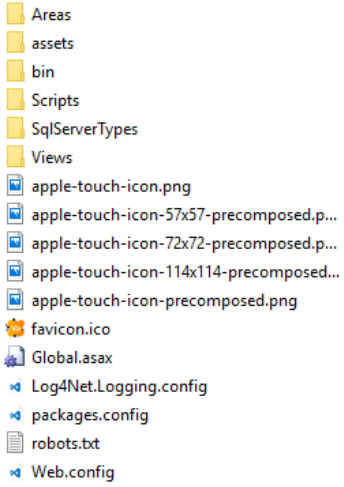
Perform the following steps to change the virtual directory.

- Open "Control Panel", "Administrative Tools" and "Internet Information Services".
- Click on the Web directory.
- Click on "Basic Settings..." in the menu to the right.
- Click on "..." to browse and search the new folder of the updated application.
- Click OK.



Changing Virtual Directory

Click on the Web directory and then click "Explore" in the menu to the right. Check that the Web directory has the structure according to below.



Directory Structure

Web Service Application (WebWin - Virtual Directory)

Perform the following steps to change the virtual directory.

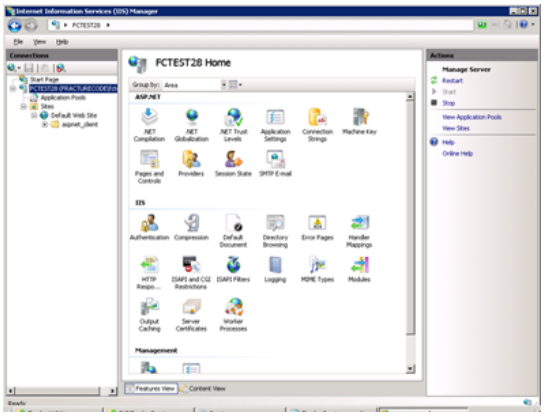
- Open "Control Panel", "Administrative Tools" and "Internet Information Services".
- Click on the WebWin directory.
- Click on "Basic Settings..." in the menu to the right.
- Click on "..." to browse and search the new folder of the updated application.
- Click OK.

Click on the WebWin directory and then click "Explore" in the menu to the right. Check that the WebWin directory has the structure according to below.

Starting IIS

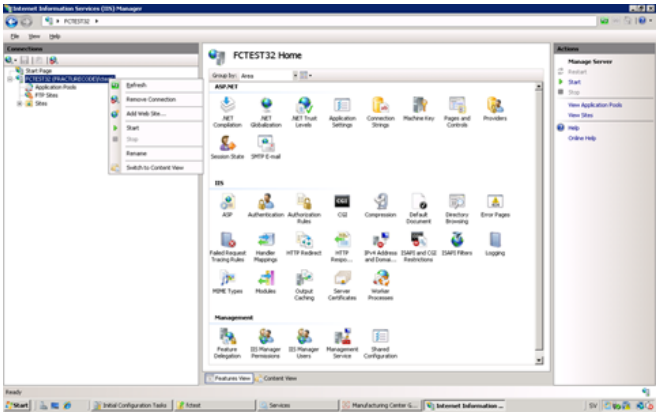
To start the IIS, it is necessary to have administrative rights over the website that is going to be started. Perform the following steps to start the IIS:

- Open Administrative Tools and click Internet Information Services (IIS) Manager.
- On the left side of the screen there is a Menu showing the Connections, search the server that you want to manage.



Connections Menu

- Right-click the server and choose the Start option.



Start option

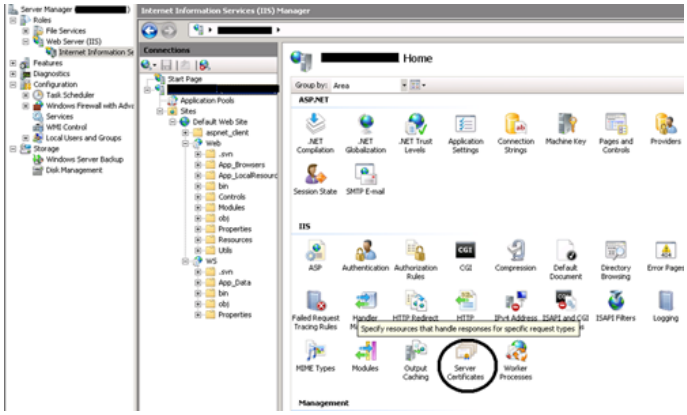
Enable SSL

To enable SSL on the Vault website, you need to create a certificate request in IIS and then define a new binding for the website with the signed certificate. This page describes the corresponding procedure to follow.

The IIS certificate can be created on one of the presentation servers, then exported and imported to the other presentation servers, as described in Enable SSL on Multiple Servers.

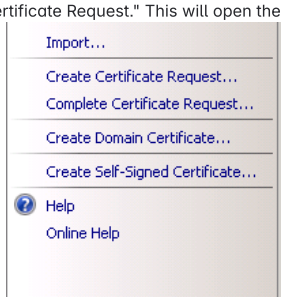
The procedure described here for the creation of IIS certificate is valid for both Windows Server 2008 and 2012.

1. Click Start, Administrative Tools, and then Internet Information Services (IIS) Manager.
2. Choose the root leaf (the one showing the computer name). From the center menu, double-click the "Server Certificates" button in the "Security" section, near the bottom of the menu.



Server Certificates

3. Next, from the "Actions" menu (on the right), click on "Create Certificate Request." This will open the Request Certificate wizard.

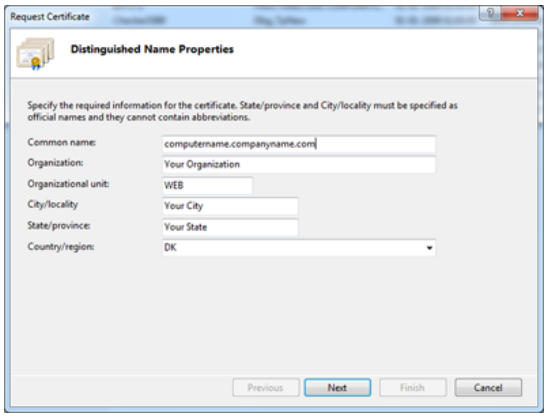


Create Certificate Request

4. In the "Distinguished Name Properties" window, enter the information as follows: **Common Name** - The name through which the certificate will be accessed. Enter the fully qualified domain name of your site. The common name should be equal to the DNS-name for the Vault in the form:
   - <HostName>.<DomainName>

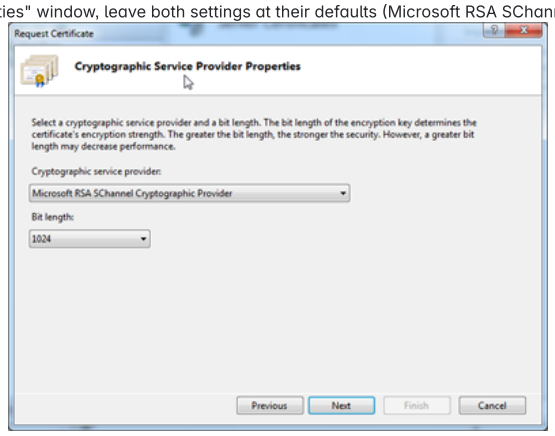
   For instance that could be:
   - Vault.companyname.com (for an internet domain)
   - computername.companyname.com (for an internet domain)
   - computername.companyname.local (for an intranet domain)
   - clusterVault.companyname.local (for an intranet domain)

   Organization - The legally registered name of your organization/company. **Organizational unit** - The name of your department within the organization (frequently this entry will be listed as "IT," "Web Security," or is simply left blank). **City/locality** - The city in which your organization is located. **State/province** - The state in which your organization is located. **Country/region** - Country Code



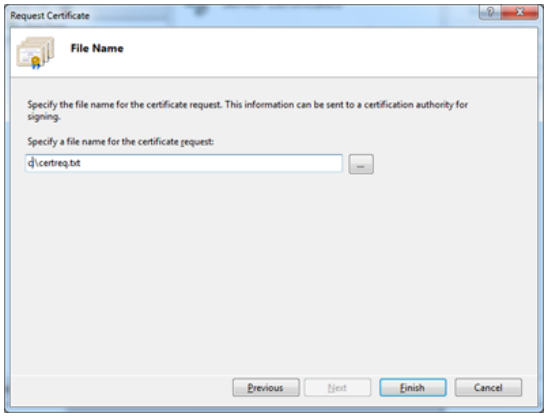
Name Properties

5. Click Next.
6. In the "Cryptographic Service Provider Properties" window, leave both settings at their defaults (Microsoft RSA SChannel and 1024) and then click next.



Cryptographic Service Provider Properties

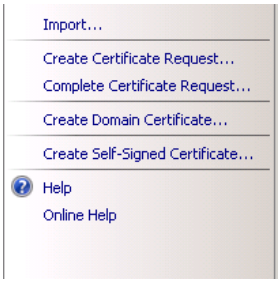
7. [Windows 2012 only] Click Next.
8. Enter a file name and a location to save your SSL Certificate Signing Request (CSR). The file should be saved as a text file (.txt) Remember the filename that you choose and the location to which you save it.



Certificate Signing Request

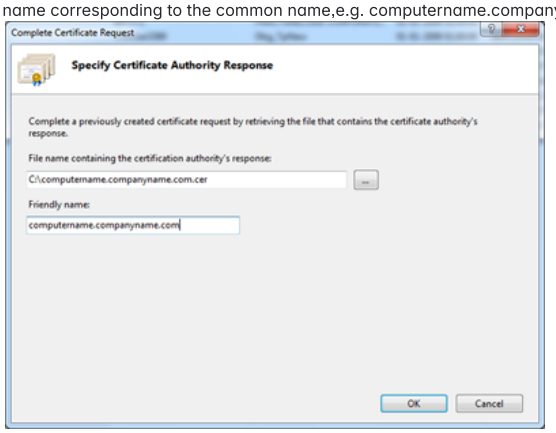
9. Click Finish to generate the file.
10. The request (the text file) is then used for creating/signing a pending certificate by either using:
    - A certificate authority, such as Verisign
    - The Inextor Certificate Generator (see /wiki/spaces/INEXTOR/pages/395812035 for instructions to use the tool)
    - Another tool

When the certificate is received press "Complete Certificate Request..."



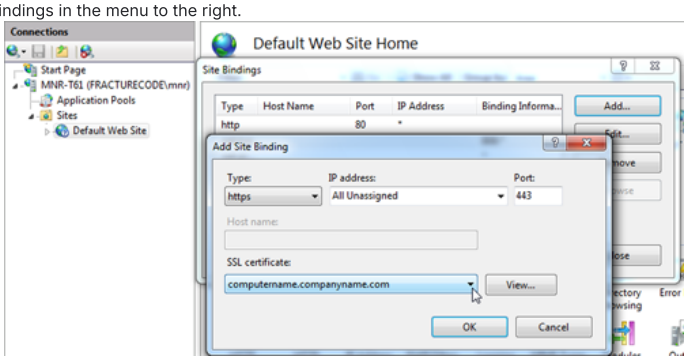
Complete certificate Request

11. Specify the certificate file and give it a friendly name corresponding to the common name,e.g. computername.companyname.com.



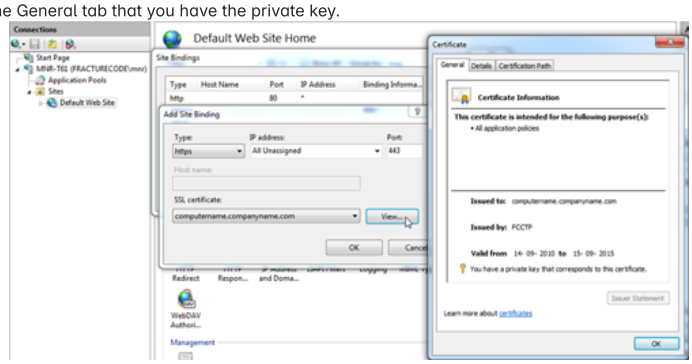
Certificate File

12. Select Default Web site and click on Bindings in the menu to the right.



Default Web Site

13. Click the Add button and select https: in the Type combo box.
14. Select the SSL certificate that you previously installed.
15. Click the View button and check on the General tab that you have the private key.



Add Site Binding

16. Also check the Certificate Path tab to verify that the certificate path is ok.
17. Click OK in the Add Site Binding dialog box.



Certificate Path

Note: Since the IIS is run under the Local Computer, the certificates that were used when signing the certificate are automatically installed into Windows certificate store under "Local Computer" account.

Enable SSL on Multiple Servers

Table of Contents

- Introduction
- Export certificate
- Import certificate

Introduction

If you have a high availability solution (with two Presentation components) you will most likely use the same certificate on both servers. You then have to export the certificates generated on the primary server (as described in Enable SSL) to the secondary server. Furthermore, export might be useful for getting backups of the certificates. Import will also be used in case of disaster recovery.

Export certificate

- Press "Start", "Run" and enter mmc.
- From file menu, select "Add/Remove Snap-in".
- Press "add" and select "certificates" and press "add" in the form.
- Select the Computer Account.
- From file menu, select "Add/Remove Snap-in".
- Press "add" and select "certificates" and press "add" in the form.
- Select the User Account.
- Select folder "Computer Account - Personal - Certificates"
- Right click on the Certificate and select "All Tasks" and "Export"
- Follow the wizard and include the private key.
    - Note: When exporting a certificate with private key you might have to enter a password. This password should then be used when importing the certificate.
- Locate the depending certificates (folders: Intermediate Certificates Authorities and Trusted Root Certificates Authorities)
- For every depending certificate, right click on the Certificate and select "All Tasks" and "Export" and follow the wizard

Import certificate

- Press "Start", "Run" and enter mmc.
- From file menu, select "Add/Remove Snap-in".
- Press "add" and select "certificates" and press "add" in the form.
- Select the Computer Account.
- From file menu, select "Add/Remove Snap-in".
- Press "add" and select "certificates" and press "add" in the form.
- Select the User Account.
- Select folder "Computer Account - Trusted Root Certificates Authorities - Certificates"
- Right click on Certificates and select "All Tasks" and "Import"
- Select the root certificate
- Follow the wizard
- Select folder "Computer Account - Intermediate Certificates Authorities - Certificates"
- Right click on Certificates and select "All Tasks" and "Import"
- Select the Intermediate certificate
- Follow the wizard
- Select folder "Computer Account - Personal - Certificates"
- Right click on Certificates and select "All Tasks" and "Import"
- Select the certificate
- Follow the wizard
    - Note. You might be required to have a password that was defined during export of the certificate.

Verify Installation

This section describes how to check, at the end of installation, that the Vault website and web services are available. If you are performing a new installation, refer to section Finalizing a New Installation.
The information is structured as follows:

- Finalizing a New Installation
- Checking the Website
- - Log in the Vault website by using a certificate
  - Log in the Vault website by using a windows account
- Checking Web Services
- Troubleshooting

Finalizing a New Installation

If you are performing a new installation, you must install the Admin certificate and its depending certificates in the Windows Certification Store of your computer in order to be able to log on to the website. This is because no users are defined in the Vault database at this point, so that you will use the Admin certificate to log on to the website and will then be able to define new users as described in the Vault Business Scenarios

For instructions on how to install certificates in the Windows Store, refer to Install Certificates.

Checking the Website

You have 3 ways to log in the Vault website:
- Use a standard user account
- Use a certificate
- Use a windows account

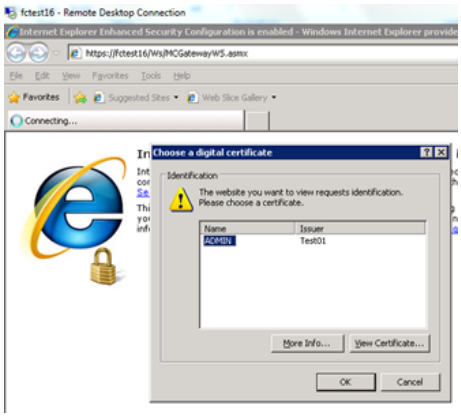
To log in as a standard user, you can either use the default **admin** account in Inextor or create your own user like shown in the Create user account userguide.
The admin account password is '**admin**', if you want to use it please modify its password for a better security.

To check if your installation succeeded with certificate and user account, please refer to the sections below.

1. Log in the Vault website by using a certificate

- Start Internet Explorer and enter the website address:
  - https://[VaultAddress]/Web if the website is using Certificate authentication
  - https://[VaultAddress]/WebWin if the website is using Windows authentication.

- If the Admin certificate pops up, select and click OK.



Certificate Pop-up

- You should now see the main page on the Vault.

2. Log in the Vault website by using a windows account

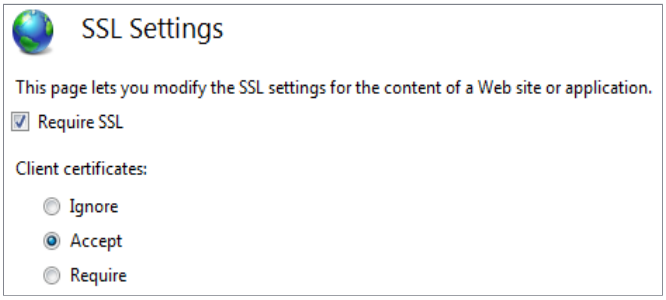
Note:

1. To test the log in by windows account, you must create a user account of type WindowsUser. To do so, you can use the admin account (which password is given above) to log in and use the Create user account userguide to create your WindowsUser account.

2. If there are multiple methods to log in your Vault website, the system will try each method with a specific order. Therefore it might be possible that the certificate method stands before the windows account method. To be sure the system will not consider your certificate instead of your windows account, delete your personal certificate in the mmc.exe application.

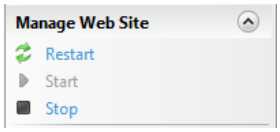
Configure your Internet Information Service (IIS):

- In IIS, select your Vault website on the left side and double click on the **SSL Settings** icon - Make sure that the client certificate is not set to Require



SSL settings client certificates not set to require

- Restart IIS web site

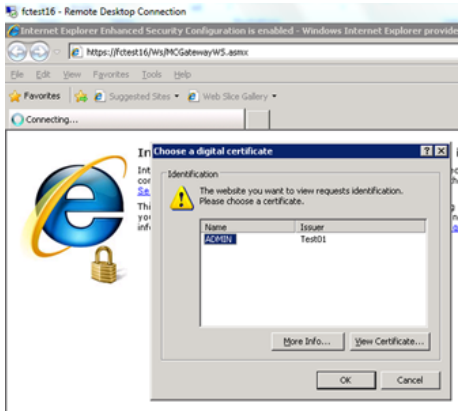


Restart web site

Try to log in the Vault website and check that log in is successful and that the windows user previously created is used.

Checking Web Services

- Start Internet Explorer and enter https://[VaultAddress]/Ws/GateWcfWS.svc
- If the admin certificate pops up, select and click OK.



Certificate Pop-up

If you encounter any problem, please see the Troubleshooting section below related to the certificate issue.

- You should then see the methods that are exposed by the Web Server on the Vault.

If you encounter any problem, please see the Troubleshooting section below related to the basic authentication.

Troubleshooting

See page Vault Installation Problems.

Authentication Methods

The Vault exposes the following web service interfaces for certificate authentication:

- Gate Certificate: Authentication and authorization is made by the Vault. The Gate supplies the Gate certificate for the authentication and authorization.
- Connection Certificate: Authentication is made by an external device (e.g. a reverse proxy server) or by the IIS. The certificate is then mapped to a Windows user. If authenticated, the request is forwarded to the Vault. The Vault authorizes the Gate by checking the Windows user that was mapped to the certificate.

Starting from version 10.1.0.2330, it is possible to authenticate a Gate or a user by sending only the certificate serial number, in a HTTP header (whose name is configurable). This is recommended only if the Vault APIs are exposed from Azure API Management, as certificate validation is fully delegated to this Azure component. See .Azure API Management driver for further details

The Vault exposes an interface for using Windows users/groups from Active Directory. In this way, it is possible to create users on the Vault that use their Windows credentials to log on. The Vault must be deployed on a Windows domain environment and users/groups must exist in the same domain where the Vault servers are registered. To combine Windows users/groups with Standard users, separate websites must be setup in the IIS. Otherwise, no special configuration in Inextor is needed for this feature. Windows Users/group are for Intranet users only. The Vault also exposes a form field injection interface. This interface should be combined with a portal that authenticates the user and enables the access the Vault. The portal then provides a role and user. The user types are as follows:

- **Standard user:** Standard user is a user where credentials to logon on the Vault is defined on the Vault Web Site.
- **Windows user:** Windows user is an existing Windows user in a Microsoft Active Directory. The user will access the Vault using their Windows credentials to logon.
- **Windows group:** Windows group is an existing Windows group in Active Directory. The users use their Windows credentials to logon to the Vault.

For information on how to set up a Vault website using Windows authentication, refer to Vault Website with Windows Authentication.

For information on how to set up a Vault website using certificate authentication, refer to Vault Website with Certificate Authentication.

For information on how to expose Vault web site and web services outside of the company domain, refer to Exposing Vault Web Services.

Refer to Database Administration procedures for database configuration settings in each case.

Configuration Files (Vault Configuration)

This section gives an overview of the configuration files for the Vault. They are described in greater detail in the following sections:

- Configuration Files - Vault Service
- Configuration Files - Vault Website
- Configuration Files - Vault Web Service

Configuration files for Vault NT Service

The list below briefly describes the configuration files for the Vault Windows service. For a more detailed description of each file, refer to Vault Service.

Configuration file	Description
ApplicationGroupOperationDefinitions.xml	The file must exist but is not used.
RoleOperationMapping.xml	This file contains the rights for each operation that each specific role have.
WindowsUserGroupMapping.xml	The file must exist but is not used.
ExtendedFields.xml	Some objects (such as account) can be given additional properties such as address. The extended fields to be used is defined in this configuration file.
Import.xml	This file defines which columns are used when using the Import function on various web pages on the Vault. For instance, instead of creating every SKU on the Vault Web site, several SKUs could be imported at the same time with the Import function.
MSKPolicy.configunit	Defines rules for MSK, such as how often Lizs should report volume information to the Vault. It also controls what kind of rules are associated with generating Inextor codes. Normally, this file should never be changed.
VaultService.exe.config	Vault NT Service application configuration file.
GateLicenses.configunit	Defines which license versions should be available to select when generating Gate license files on the Vault website. If this file does not exist, then the license version is determined by the setting in VaultService.exe.config. Note: This file is optional and might not exist on the Vault.

Configuration files for Vault Website application

The list below briefly describes the configuration files for the Vault Windows service. For a more detailed description of each file, refer to Vault Website.

Configuration file	Description
InfragisticsGrid.xml	Vault website result grid configuration. On many Vault pages, you can define a search criterion and click the search button to populate a grid with data. This file defines which columns to show in the result grid.
Web.config	Vault website application configuration file
VaultConnection.configunit	Defining which Vault NT service(s) should serve this client for load balancing.

Configuration files for Vault Web Service application

The list below briefly describes the configuration files for the Vault web service. Equivalent files exist for each web service and can be configured similarly. For a more detailed description of each file, refer to Vault Web Service.

Configuration file	Description
Web.config	Vault Web service application configuration file
VaultConnection.configunit	Defining which Vault NT service(s) should serve this client for load balancing

Locations

We summarize below the list of files described in this page, along with the location where each of these files can be found.

Configuration file	Description
ApplicationGroupOperationDefinitions.xml	C:\[DeploymentFolder]\CVSVaultservice \Security\Authorization
RoleOperationMapping.xml	C:\[DeploymentFolder]\CVSVaultservice \Security\Authorization
WindowsUserGroupMapping.xml	C:\[DeploymentFolder]\CVSVaultservice \Security\Authorization

Configuration file	Description
ExtendedFields.xml	C:\[DeploymentFolder]\CVSVaultservice\ExtendedFields
Import.xml	C:\[DeploymentFolder]\CVSVaultservice\ExtendedFields
MSKPolicy.configunit	C:\[DeploymentFolder]\CVSVaultservice\RuleEngine
VaultService.exe.config	C:\[DeploymentFolder]\CVSVaultservice
GateLicenses.configunit	C:\[DeploymentFolder]\CVSVaultservice
InfragisticsGrid.xml	C:\[DeploymentFolder]\Cvs.Vault.WebSite\Resources
Web.config	C:\[DeploymentFolder]\Cvs.Vault.WebSite
VaultConnection.configunit	C:\[DeploymentFolder]\Cvs.Vault.WebSite\bin
Web.config	C:\[DeploymentFolder]\GateWS
VaultConnection.configunit	C:\[DeploymentFolder]\GateWS\bin

Vault Service

This page gives details about the following configuration files for the Vault Service:

1. ApplicationGroupOperationDefinitions.xml
2. RoleOperationMapping.xml
3. WindowsUserGroupMapping.xml
4. MSKPolicy.configunit
5. VaultService.exe.config
6. ExtendedFields.xml
7. Import.xml
8. CheckPRODStatus.vdx
9. GateLicenses.configunit

The configuration files 6 to 9 are included here for completion but you **should not** need to change them. Modifying these files can disrupt the functionality of the Vault.

ApplicationGroupOperationDefinitions.xml

The file must exist but is not used. It is intended for future functionality.

RoleOperationMapping.xml

This file contains the rights that each role has according to the table below. For instance, the value of 21 for Liz corresponds to the rights (List, Edit, Validate. That is 1+4+16).

Rights	Value
List	1
Create	2
Edit	4
Delete	8
Validate	16
Manage	32

Below, the default settings in the role operation mapping are shown. For a detailed description of all operation rights, see /wiki/spaces/INEXTOR/pages/395816844. Changes in this file should be made with care, since there are other rules that are predefined in the system. For instance setting SKU operation to 0 for LE and to 15 for MAN and IMP role will only get the effect that LE cannot create SKUs and neither can MAN or IMP since the SKU must be created by the LE role. Hence any change should be followed up with a test that the expected functionality is achieved.

> Only the Vault owner might change this file and normally the default settings (or very similar settings) are used.

WindowsUserGroupMapping.xml

The file must exist but is not used. It is intended for future functionality.

MSKPolicy.configunit

This file defines rules for prohibiting the Liz from generating codes and determining how often the Liz will supply production batch information to the Vault.

> The parameter ProductionUpdateTimeoutPeriodStr is set to 0 by default. That means this function is disabled. Setting this parameter to another value will prevent Lizs connected the Vault from producing when the timeout period is exceeded. Before enabling this function, you should be aware of the consequences. Hence, a business risk investigation should be performed before enabling. Also note that you have to state any change in this file in government regulations to all manufacturers using the Vault.

For private Vaults it is abnormal to make any changes in this file except for testing purposes. For settings in the MSKPolicy file to be registered by the Liz, the settings must be made before requesting an MSK i.e. you must request a new MSK if you make any amendments.

Parameter	Description
ProductionUpdatePeriodStr	Time interval for each transaction log
ProductionUpdateTimeoutPeriodStr	Timeout time (that is a Liz that have not transmitted any transaction logs for the last X days, where X is the time out time will be unable to generate any Inextor codes).

Parameter	Description
EstampPolicyAction, Warning	When remaining Volume Control units are below threshold e.g. 1000, a warning is shown on Liz.
EstampPolicyAction, Termination	When remaining Volume Control units are below threshold e.g. 0, the batch is terminated on Liz.

The threshold for the remaining E-stamps may not be acted upon exactly when the threshold is reached. This is because it is dependent upon how often communication between the Gate and Liz takes place.

VaultService.exe.config

The format for the file is defined by a schema for configuration files for applications using .NET Framework (http://msdn.microsoft.com/en-us/library/1fk1t1t0.aspx). You have to supply information into mandatory placeholders in the file (Marked as ######). The only mandatory placeholder to enter information for is the connection string (CertificateDBConnectionString) to the database that is going to be used (see the Vault Service Installation page). Parameters in section “appSettings Vault Settings” are described below. The sections Visio rule file and NxBRE is related to the rule engine (CheckPRODStatus.vdx). The logging is configured by default in file Logging.Serilog.json.

```
<configuration>
    ...
    <appSettings>
        <!-- Vault Settings -->
        <!-- Visio rule file -->
        <!-- NxBRE configuration -->
    </appSettings>
    ...
</configuration>
```

The table below shows the most important keys available to be configured in the Vault Service. The keys with a default value below should be left unchanged unless there is a specific need for another setting. The keys below with an empty default value should always be determined for your Vault specific needs.

There are more possible keys, visible in the Vault Application Settings (menu *Configuration*). The description of a key can be visualized as a tooltip, by hovering over the key name in the application.

Updating the value of the keys below should now be done in the Application Settings tab of the Vault administration program. It is still possible to update the configuration file of the Vault server, but such a direct change will be lost during upgrades.

Key	Default value	Description
AbsoluteDuration	1d	Lifetime of session until it is expired, it is used when session is created, and set a absolute expired time of session (this time can not be change in the future by renewing and it differs from expired time above). Session will be expired if it reaches expired time or absolute expired time. Put negative value if we don't want use this value and related functions.
CertificateDbDataProvider	provider=System.Data.SqlClient	Data Provider Name is used to connect Certificate Database. Values: • provider=System.Data.SqlClient: certificates are stored in SQL database • provider=System.Data.SqlServerCe.4.0: certificates are stored in SQL server compact file, the version 4.0 here can be changed depend on the used version of SQL server compact.
CodeDuplicateProtectionEnabled	False	When set to True, finding a duplicate code in a TPD order prevents a chunk of codes from being sent to the Gate. See Configuring Materialization for more details. This feature is currently only supported when Azure Table Storage is used to store TPD codes.
CryptoDatabaseSafeType	DataProtect	Specify the mechanism used to protect data store in database. Value could be: • None: data is not encrypted • DataProtect: encrypt symmetric key using Data Protection, and then use symmetric key to encrypt DB • SingleRsa: encrypt symmetric key using Rsa algorithm with one 'specific' key, and then use symmetric key to encrypt DB. • DoubleRsa: encrypt symmetric key using Rsa algorithm with one 'specific' key and again for one 'global' key, then use symmetric key to encrypt DB. • RsaDataProtect: encrypt symmetric key using Data Protection and again for one 'global' key (using rsa algorithm),then use symmetric key to encrypt DB.
DatabaseHealthReceivedDataPath	Empty value	Specifies a directory where the Vault will cache the received DACPACs from the connected services. The files are generally stored for troubleshooting purposes.

Key	Default value	Description
		If the value is empty, the path used is: {RootApplicationData}\Health\Database\ReceivedData (where RootApplicationData is another configuration entry).
DatabaseHealthReferenceBlobStorageConnectionString	An endpoint mapped to a Blob storage account.	Connection information to Azure blob storage account where reference data is stored. This setting should only be changed after discussing with the development team.
DatabaseHealthReferenceBlobStorageContainer	gate-reference-dacpac	Container name in Azure blob storage account where reference data is stored. This setting should only be changed after discussing with the development team.
DatabaseHealthReferenceDataPath	Empty value	Specifies the directory where the Vault caches the downloaded from Azure reference DACPAC files for health checks. If the value is empty, the path used is: {RootApplicationData}\Health\Database\ReferenceData (where RootApplicationData is another configuration entry).
DataSetValidation	./ExtendedFields/LizDataSetValidation.xml	Path to the file containing the validation rules for all input data types
DataValidationRaiseException	false	Specify if error in validation cause exception or only logging.
DBConnectionString		Name of the connection string for Certificate in DB. Must be configured directly in the Vault configuration file.
DeadlineForRenewal	7d	Interval of time for renewal before the certificate is marked as expired.
DefaultFunctionOrigin	Employee	The default type of check when checking codes on the Vault Web site. Possible values: Authorities, Customer, Customs, Distribution, Employee, Media, QA, Retailer, Test.
DeleteLogsBefore	180d	Logs that are older than the specified duration are entirely deleted from database.
DeletePRODsUpdatesBefore	180d	PRODsUpdates that are older than the specified duration are entirely deleted from database.
EStampSupport	false	Whether volume control is enabled on Vault web site.
GateNotifyType	SKU PROD	The default policy for Gate notifications to Vault PROD. • None: nothing is notified • SKU: notify all SKUs from Gate to Vault • PROD: notify any SKUs from Gate that relate to PRODs sent to Vault • HostName: notify the host machines and port numbers that run the Gate service to the Vault (only if Vault is PRIV and factory role type MAN).
GateReportType	AdditionalQty	The default policy for Gate reporting to Vault PROD (None or AdditionalQty).
GeoNameDBConnectionString		ConnectionString to the GeoName database. Must be configured directly in the Vault configuration file.
GeoNameDBDataProvider	System.Data.SqlClient	DataProvider for ConnectionString to the GeoName database. Must be configured directly in the Vault configuration file.
HideAssemblyResolutionFailure	(XmlSerializers)|(resources)	Skips these exceptions, without trying to localize them. Resources are missing for these system dlls and that would cause infinite recursion in the EL4Net exception hook.
IgnoreNonTransactionalUpdates	false	The Vault ignores non transactional Updates (UpdateType.None: the production changes is sent with type None) During the production changes, for each interval defined by ProductionUpdatePeriodStr, the Production changes will be send to Vault with type "VolumeUpdate". For each interval of UploadInterval, the Liz will check information for uploading, if the value of ProductionUpdatePeriodStr is not reached, the production changes will be sent with type "None" ProductionUpdatePeriodStr is defined in MSK license. UploadInterval: Sets how often PRODs, SKUs, pending Liz registrations, registration updates and deregistrations are sent to Vault.
ImportStructureFile	.\ExtendedFields\Import.xml	File describing the imports.
InitialDuration	40M	Lifetime of session until it is expired. it is used the first time session created (if session is renewed, expired time of session is set again by using renewalDuration token).

Key	Default value	Description
InterfacesBindingType	System.ServiceModel.NetNamedPipeBinding System.ServiceModel.NetTcpBinding	The service URI is: net.tcp://localhost:<CheckerServicePort>/InterfacesChecker/ICodentifyInterface MEX uri: net.tcp://localhost:<CheckerServicePort>/InterfacesChecker/mex
KeepOneEveryXHour	4	One PRODsUpdates record is preserved in the database for every period of x hours
KeepPRODsUpdatesAfter	90d	PRODsUpdates that are more recent than the specified duration are entirely preserved in database. PRODsUpdates that are more recent than DeleteLogsBefore and older than KeepLogsAfter are deleted according to the KeepOneEveryXHour parameter.
ListenPort	8086	Listen port of Vault Service.
LizApprovalRequired	false	The default policy regarding Liz approval. • true: Liz register request must be approved manually, Serializerstatus is Pending. • false: Serializerstatus is automatically marked as Authorized.
MinimalAuditDataActionRequirement	None	All AuditDataAction points will require at least this minimal requirement. This setting is useful for development/testing purpose. Changes the data action requirement of all data actions. Convenient to enable all data actions in a single place. • None: Turns off recording of data action events that do not have specific requirement set in the database • Recorded: The audit events will be recorded in the audit log (see Audit Configuration)
MinimalAuditEntityChangesRequirement	None	All AuditDataAction points will require at least this minimal requirement. This setting is useful for development/testing purpose. Changes the entity changes requirement of all entity changes. Convenient to enable all entity changes in a single place. • None: turns off recording of entity change events that do not have specific requirement set in the database. • Recorded: the audit events will be recorded in the audit log (see the documentation for more details about how to configure the audit log).
MinimalAuditOperationRequirement	None	All AuditOperation points will require at least this minimal requirement. This setting is useful for development/testing purpose. Changes the audit operation requirement of all audited operations. Convenient to enable all audited operations in a single place. • None: turns off recording of audit operation events that do not have specific requirement set in the database. • Recorded: the audit events will be recorded in the audit log (see the documentation for more details about how to configure the audit log). • Reason: in addition to recording the event, the application will require the user to enter a business reason that requires his action. • Signature: in addition to asking for a reason and recording the event, the application will require the user sign his action.
MSKApprovalRequired	false	The default policy regarding MSK approval. • true: Users have to manually approve MSK request at request page. • false: MSK requests are approved automatically.
OutdatedStatusBlocked	true	Block status for outdated production.
PermutationMaximumCorrections	3	Maximum number of corrections in permutations. Value must be in the range of 1 to 12.
PrimaryCheckCleanupDataMonthsToPreserve	6	Number of months worth of Primary Check data to preserve, from 2 to 11, 2 meaning keep only current and previous one
ProductionOutdatedInterval	1d	Production will be outdated if it does not reach Vault within time interval set by this value.
ProgramName	Vaultservice	Name of the Program.

Key	Default value	Description
ProgramUnitName	Vaultservice	Name of the Program unit built by this deployment.
ProgramVersion	10.0	Version of the Program for this deployment.
ProtectedDBConnectionString	false	Specify if the connection string is protected or not.
RemotingCustomErrorsModes	Off	Defines how to handle custom exceptions. Three possible values (On, Off, RemoteOnly) describe when exception filtering is activated.
RenewalDuration	30M	Lifetime of session until it is expired, it is used when session is renewed and expired time of session is set again by this value.
RoleApprovalRequired	false	The default policy regarding role approval. • true: Pending roles have to be manually approved by user. • false: Pending roles are automatically approved.
RootCertificateApprovalRequired	true	The default policy regarding Root Certificate approval. • true: Root Certificate in chunk of registered Liz certificates is marked as "not trusted" when registering Liz. • false: Root Certificate in chunk of registered Liz certificates is marked as "trusted" when registering Liz.
SKUApprovalRequired	None	Types of SKU require approval by users, they can be combined together. • None • Export: Export SKU require • Local: Local SKU require • Import: Import SKU require
SolutionCompany	Inexto	Name of the Company providing this Solution.
SolutionName	Inextor	Name of the Solution.
Path	.\SSRS	Path to the folder from which the Vault is loading the report path. The default path is relative to the VaultService installation folder. This path can be changed to an absolute path outside of the Vault installation folder (e.g. to folder ProgramData) so that the user doesn't have to redeploy the report files after each upgrade.
TaxLevelSupport	false	Whether tax level support is enabled on Vault web site. If changed, you need to restart Vault website. (recycle App Pool in IIS)
TpdForecastForceManualOrders	false	When set to true, all forecasts received from any Gate goes into 'manual import needed' state. Can be used if another entity than the Vault handles ordering to ID issuers.
TpdForecastHandlerProcessingInterval	5M	Processing of new forecast message is done at a time interval given by this parameter.
TpdMaxAutomaticApprovedQuantity	100000000	Sets a limit to the maximum amount of codes that can be approved automatically. If this property is defined on the Vault, all TPD factories will be affected. To set a limit at a factory-level, choose a facility, and provide the value in the attributes in the field. This will override the value defined on the Vault.
TpdReleasedOrdersProcessingInterval	5M	Processing of approved orders is done at a time interval given by this parameter. *Note: This setting has been replaced, in version 4100, with the configuration of the underlying job.*
TpdWaitTimeBeforeAutomaticOrderRelease	0M	Time to wait before automatically approving an order. If set to 0, the orders with a quantity lower than TpdMaxAutomaticApprovedQuantity are immediately approved and processed. This wait time also applies to orders which have the 'immediate' flag; when the involved ID Issuer has a 'process order immediately' policy of "immediate with explicit confirmation', the 'immediate' flag is clear when this wait time is elapsed.
VaultHost	localhost	Host name of Vault Service.
VaultOnlyCheckCodes	false	When set to true, the Vault will not run any background job (message processing disabled) and will only respond to incoming calls from iTrack / Inextrack systems. Must be configured directly in the Vault configuration file.

Key	Default value	Description
VisioPRODRuleFile	.\RuleEngine\CheckPRODStatus.vdx	Path to the Visio file which defines rules interacting with production change.
VisioPRODRulePage	MSK, SKU, SKUActor, Gate, Liz, PROD, Helper, GateActor, SKU2Actor, SKU2Gate, Estamp, TaxLevel, Query	Which pages in visio file are used to compute the rule.
WasteTypesApprobation	SKUDestroy PRODDestroy SUPIDestroy	The default policy regarding Waste supported. It defines the types not requiring approval. Possible values: SKUWaste SKUDestroy PRODWaste PRODDestroy SUPIWaste SUPIDestroy.

See sections Business Configuration Parameters and /wiki/spaces/INEXTOR/pages/395826584 for details.

Warning: The configuration files below are included here for completion but you **should not** need to change them. Modifying these files can disrupt the functionality of the Vault.

ExtendedFields.xml

This file defines extended fields that can be used on the Vault Web site; that is, you can extend what kind of properties are associated to a certain object (see table names below). By default, a number of predefined properties (e.g. address property for Actors) are defined in this file, but you can change the file according to your own needs. Note that just changing the file will not have any effect; the file simply defines what properties you can use and hence will be stored in the database. You also need to adjust the related web pages, making it possible to enter the properties and also making it possible to view them.

Only the Vault owner might change this file.

Example: For an actor (account or Gate) you also want the account to enter an Address field:

- Check that the Address field is defined in the ExtendedFields.xml file. If not add the field under Actor and define the number of characters the field can have (e.g. varchar(50)). The require field should always be set to yes, which just means that it will always be stored in the database even if no value is entered.
- Edit the associated web pages and add a label and a textbox. For adding labels and text boxes, knowledge of ASP.NET is required as well as of the Inextor project structure. How to do this is out of the scope of this documentation.
- The field will not automatically become mandatory for anyone creating an account (or Gate) on the Web site. To make filling the field mandatory, you have to write source code for business rules as well (which requires knowledge in ASP.NET, C# and Inextor source code project structure). For this you need to recompile the Inextor project files. How to manage this kind of changes is out of the scope of this documentation.
- You might want to have the address field visible in the result grid on the Account page. In this case, you have to configure the Infragistics.xml file accordingly.

```
<?xml version="1.0" encoding="utf-8" ?>
<!--

Use one of the following type
Possible types:
- bit: converted as boolean /bit
- integer: converted as int / int
- number: converted as double / float
- Anything else is converted as string / (same as specified type)
-->

<Tables>
  <Table Name ="Actors">
    <Column Name="LongName" Type="nvarchar(60)" Require="yes"/>
    <Column Name="ShortName" Type="nvarchar(60)" Require="yes"/>
    <Column Name="InternalCode" Type="nvarchar(10)" Require="yes"/>
    <Column Name="GeoNameID" Type="integer" Require="yes"/>
  </Table>
  <Table Name="Roles">
    <Column Name="Comments" Type="nvarchar(100)" Require="yes"/>
    <Column Name="GeoNameID" Type="integer" Require="yes"/>
  </Table>
  <Table Name="GatesDetails">
    <Column Name="ShortName" Type="nvarchar(20)" Require="yes"/>
    <Column Name="LongName" Type="nvarchar(60)" Require="yes"/>
    <Column Name="GeoNameID" Type="integer" Require="yes"/>
  </Table>
  <Table Name="UserPreferences">
    <Column Name="FirstName" Type="nvarchar(50)" Require="yes"/>
    <Column Name="LastName" Type="nvarchar(50)" Require="yes"/>
    <Column Name="Email" Type="nvarchar(50)" Require="yes"/>
    <Column Name="EmailSubscribed" Type="bit" Require="no"/>
  </Table>
</Tables>
```

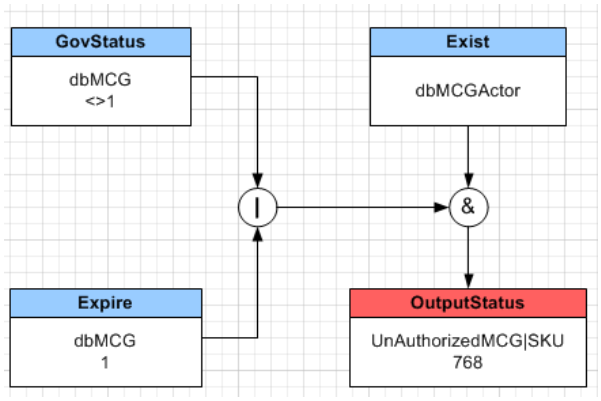
Import.xml

This file defines which columns are used when using the Import function on various web pages on the Vault. The file contains four tables:

- SKU: On the Vault Web Site on the SKU page it is possible to Import SKUs, see Vault Website - Products - Import. This table defines the Columns in the Excel document that pops up when "Download Template" is selected.
- SKUAssignedActor: On the Vault Web Site on the Account page it is possible to Import SKUs to the selected Account, see document "Inextor Operation Guide" in the section "Account – Details – SKU". This Table defines the Columns in the Excel document that pops up when "Download Template" is selected.
- SKUAssignedGate: On the Vault Web Site on the Manufacturing Centers page it is possible to Import SKUs to the selected Gate, when "License all SKUs" are unmarked, see Vault Website - Factory - Import. This table defines the Columns in the Excel document that pops up when "Download Template" is selected.
- Tax: On the Vault Web Site on the Product Groups page it is possible to import Product Groups, see Product Groups and Products. This table defines the Columns in the Excel document that pops up when "Download Template" is selected. Note: With the default configuration in RoleOperationMapping.xml, only ADM can create Product Groups.

CheckPRODStatus.vdx

This Visio file defines rules for the status field and related error codes for production batches (Page Production, Batches on Vault Web Site). The file is a third party business rule component. The tool and manual for managing the Visio file can be downloaded on http://sourceforge.net/projects/nxbre/ For being able to change setting in the file detailed knowledge of the tool is required as well as knowledge about the source code and database tables used by Inextor. For instance, a Gate that is not Authorized or is expired and the actor that owns the Gate exists then a production batch coming from that Gate will get error code equal to UnAuthorizedGate, see sample code below.



Visio screenshot

GateLicenses.configunit

This file is optional, refer to Migration for further details.

Vault Website

Web.config (Vault Website)

The format for the file is defined by a schema for configuration files for applications using .NET Framework (http://msdn.microsoft.com/en-us/library/1fk1t1t0.aspx). Usual configuration settings of the ASP.NET platform apply (for example session timeout). Parameters in section "appSettings" are described below. The logging is configured by default in file Logging.Serilog.json. Normally the other section could be left with the default settings.

```
<configuration>
    ...
    <appSettings>
        <!—Vault Web Site Settings -->
    </appSettings>
    ...
    <SelectedConfigUnits>
        <!-- add credential providers -->
    </SelectedConfigUnits>
    ...
</configuration>
```

AppSettings: Vault Website settings. **SelectedConfigUnits:** Define which of the following credentials providers to use:

- **CertificateCredentialProvider**: Enables login on a Vault with Account Certificate (i.e. using the certificate that can be downloaded on the Account detail page of the Vault website or using the Admin certificate)
- **SSPICredentialProvider**: Enables login on a Vault with users stored in Microsoft Active Directory (Referred as WindowsUser and WindowsGroup on th Vault website user page)
- **LoginPasswordCredentialProvider**: Enables login with standard users (Referred as standard user on the Vault website user page).
- **AzureADCredentialProvider**: Enables login on a Vault with users stored in an AzureAD tenant. See VaultConnection.configunit section below for additional configuration.
    - Note 1: this provider is not compatible with SSPICredentialProvider, which must be disabled.
    - Note 2: Azure AD returns user names in the format user@domain (instead of domain\username for on-premises AD) and group names in the format groupName (instead of domain\groupName for on-premises AD). If you are migrating from on-premises AD, contact support in order to migrate the existing rights.

VaultConnection.configunit

The configuration defines if the website is served by one (Single Server) specific Vault NT service or several using the built-in load balancer. For each server, you define the host instance of the Vault NT service by defining the host name either with FQDN or with IP-addresses. Finally, you define which credentials providers to use and the priority order to use.

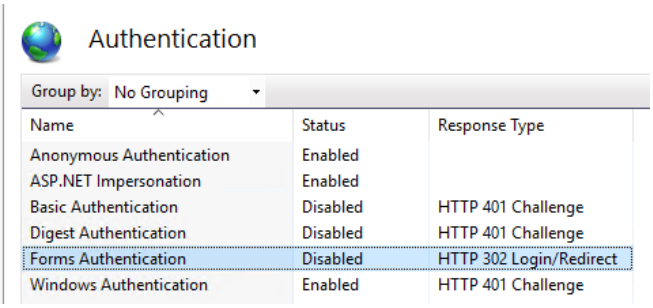
- **Single server configuration**: Define values for the keys <HostName> and <Port> below the comment <!-- In case of single server -->
- **Load balancing configuration**: Define values for the keys <HostName> and <Port> below the comment <!-- In case of single loadbalancing -->. You also have to define a unique Id for each connection (e.g. ConnectionId="0" or "1".
- **Connection providers**: Define priority for each provider. If having more than one provider, the one with the highest priority (lowest number for Priority) will be tested first. If authenticating fails with that provider, authentication will be tried with the next provider.
    - If the Azure provider is required, its declaration must be un-commented while the SSPI provider must be commented out. It is also necessary to register an application in the Azure portal and provide the Azure AD tenant id, the application id, an application secret and the public web site address in the configuration file. See Use Entra ID / Azure AD to authenticate users for information on how to register an application in Azure.
      If using a multi-tenant Azure AD credential provider, the list of accepted tenant ids must be set up.

The key <UseOrOverride>LoadBalancedSessionAffinity,... must be enabled if using load balancer or should be commented out if using single server.

Web Services

IIS configuration

The authentication mechanisms must be configured like this:



Web.config (Vault Web Service)

The format for this file is defined by a schema for configuration files files for applications using.NET Framework (http://msdn.microsoft.com/en-us/library/1fk1t1t0.aspx). Parameters in section "appSettings" are described below. The logging is configured by default in file Logging.Serilog.json. Normally the other section could be left with the default settings.

```
<configuration>
    ...
    <appSettings>
        <!--Vault Web Site Settings -->
    </appSettings>
    ...
    <SelectedConfigUnits>
        <!-- add credential providers -->
    </SelectedConfigUnits>
    ...
</configuration>
```

SelectedConfigUnits: Define which of the following credentials providers to use:

- **CertificateCredentialProvider**: Enables the Gate to authenticate on Vault with Gate Certificate (Default method)
- **SSPICredentialProvider**: Enables the Gate to authenticate on Vault with a connection certificate that is mapped to a user stored in Microsoft Active Directory). The connection certificate cannot be downloaded from Vault website; instead, the Vault owner must provide this certificate separately to manufacturers/importers owning Gates. How to setup an environment using connection certificate is out of the scope of this documentation.
- **AzureADHeadlessCredentialProvider**: Enables login on a Vault with users stored in an AzureAD tenant. See VaultConnection.configunit section below for additional configuration.
    - Note: Azure AD returns user names in the format user@domain (instead of domain\username for on-premises AD) and group names in the format groupName (instead of domain\groupName for on-premises AD). If you are migrating from on-premises AD, contact support in order to migrate the existing rights.

VaultConnection.configunit

The configuration defines if the web site is served by one specific Vault NT service or several using the built-in load balancer. Furthermore, you define address and port for the NT service(s). Finally, you define which credentials providers to use and the priority order to use. The possible configuration settings for the Vault web service are the same as for the Vault web site; see Vault Website - VaultConnection.configunit.

Note: using Azure AD IS compatible with the SSPI provider in this case. When using the above documentation, disregard the need to disable the SSPI provider. Also, there is no need to add configuration information for Azure (the configuration in the Vault web site is reused).

Configure Vault to use NetNamedPipeBinding (Configuration)

Introduction

By default, the Vault is configured to use NetTcpBinding for its connection. In all configurations today, the Vault Service and Web Services are installed on the same machine. So we can replace NetTcpBinding with NetNamedPipeBinding. This will avoid data serialization because the data is moved between processes using Windows shared memory, therefore improving the performance.

Configure Vault Service

The Vault Service can run as one of this following method:

- **CVSVaultserver:** The Vault is installed and run as a console application.
- **CVSVaultservice:** The Vault is installed using Windows InstallUtil and run as a Windows Service

To replace NetTcpBinding with NetNamedPipeBinding, we need to modify the **App.config** of CVSVaultservice/CVSVaultserver:

1. Find and open **App.config** that need to be modified
2. Navigate to the section **<appSettings>**
3. Add (or uncomment) this setting under section **<appSettings>**

```
<add key="BindingType" value="System.ServiceModel.NetNamedPipeBinding"/>
```

4. The final **App.config** will look like the following:

```
<?xml version="1.0"?>
<configuration>
   <configSections>
      ...
   </configSections>
   <appSettings>
      ...
      <add key="VaultHost" value="localhost"/> <!-- specify the host name of the Vault -->
      <add key="ListenPort" value="8086"/> <!--Listen port of Vault Service-->
      <add key="IsSecureChannel" value="true"/> <!-- specify the channel is secure or not -->
      <!-- Binding type of Vault Service.
           Values: System.ServiceModel.NetNamedPipeBinding, System.ServiceModel.NetTcpBinding
       Default value is NetTcpBinding.
       Uncomment below settings to configure Vault as NetNamedPipeBinding -->
      <add key="BindingType" value="System.ServiceModel.NetNamedPipeBinding"/>
      ...
   </appSettings>
   ...
</configuration>
```

Notice: In some configurations, the Vault Service can be configured with a configunit called *TimeoutSettings.configunit*. In this case, we need to modify this configunit as well (see the following sample configunit, notice the node called **BindingType**):

```
<?xml version="1.0"?>
<ConfigUnit>
    ...
    <ConfigFragment>
        <ComponentRepository>
            <ComponentDefinition Name="Cvs.Vault.VaultWSWcfManager" OnMiss="Add" OnExist="Replace">
                <ContractInterfaceName>Cvs.Vault.IVaultWSWcfManager,Cvs.Vault.Common</ContractInterface
                <StaticComponent>
                    <Implementation>
                        <TypeName>Cvs.Vault.VaultWSWcfManager, Cvs.Vault.Business</TypeName>
                        <VaultWSWcfManagerConfig>
                            <IP>@{VaultHost}</IP>
                            <Port>@{ListenPort}</Port>
                            <IsSecure>true</IsSecure>
                            <!-- Add BindingType to VaultWSWcfManagerConfig -->
```

```
                            <BindingType>@{BindingType}</BindingType>
                            <SendTimeoutStr OnMiss="Add" OnExist="Fail">@{DefaultWcfTimeout}</SendTimeo
                            <ReceiveTimeoutStr OnMiss="Add" OnExist="Fail">@{DefaultWcfTimeout}</Receiv
                            <OpenTimeoutStr OnMiss="Add" OnExist="Fail">@{DefaultWcfTimeout}</OpenTimeo
                            <CloseTimeoutStr OnMiss="Add" OnExist="Fail">@{DefaultWcfTimeout}</CloseTim
                        </VaultWSWcfManagerConfig>
                    </Implementation>
                </StaticComponent>
            </ComponentDefinition>
        </ComponentRepository>
        ...
    </ConfigFragment>
</ConfigUnit>

```

Configure Vault Website and Web Services

After configuring the Vault Service, you need to modify the **Vault Website** and **Vault Web Services** configuration as well, so that the Vault website and web services can communicate with the Vault Service via NetNamedPipeBinding.
Web services are services that are hosted in IIS. There are several kinds of Web Services:

- **VaultserviceWS**: Web Service to communicate with client (ex: iTrack)
- **GateManagerWS**: Web Service to communicate with Gateway
- **OfflineCheckingWS**: Web Service used by the factory to synchronize data with an offline vault
- **SsccWS**: Web Service providing SSCC codes management.

To configure the Vault Website and Web Services to use NetNamedPipeBinding, you must modify the corresponding **Web.config**:

1. Find and open **Web.config** that you need to modify
2. Navigate to the section **<appSettings>**
3. Add (or uncomment) this setting under section **<appSettings>**

```
<add key="BindingType" value="System.ServiceModel.NetNamedPipeBinding"/>
```

4. The final **Web.config** will look like the following:

```
<?xml version="1.0"?>
<configuration>
    <configSections>
        ...
    </configSections>
    <appSettings>
    ...
    <add key="TargetHost" value="localhost"/>
        <add key="TargetPort" value="8086"/>
    <!-- Binding type of Vault Service.
        Values: System.ServiceModel.NetNamedPipeBinding, System.ServiceModel.NetTcpBinding
        Default value is NetTcpBinding.
         Uncomment below settings if Vault was configured as NetNamedPipeBinding -->
    <add key="BindingType" value="System.ServiceModel.NetNamedPipeBinding"/>
    ...
    </appSettings>
    ...
</configuration>
```

Configuring Web Clients with Limiting Access

VaultRemotingWindowsIdentity/ VaultRemotingWindowsGroup configuration

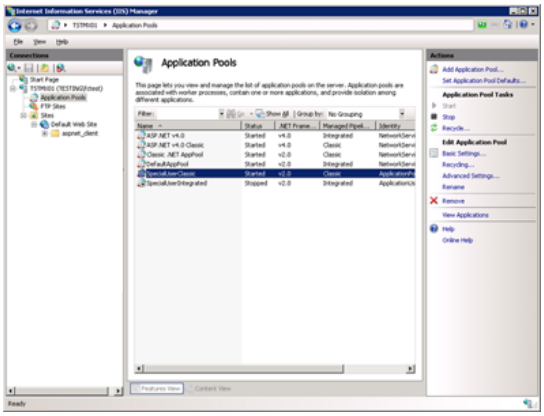
If the Vault NT Service is setup for limiting access, only clients with a certain Windows account or group are allowed to connect the Vault NT Service, hence the Web clients must be configured accordingly.

> **Note**: To set up the Vault NT service for limiting access, specify a value for **VaultRemotingWindowsGroup** and **VaultRemotingWindowsIdentity** in the **VaultService.exe.config** configuration file (see Configuration Files for the location of this file). If these values are specified, then the Vault service will only allow the specified account (for VaultRemotingWindowsIdentity) or users that are members of this group (for VaultRemotingWindowsGroup) to connect.

First of all the web.config of the website and web services have to be verified so that the impersonate element is set to false:

```
<authentication mode="Windows"/>
<identity impersonate="false"/>
```

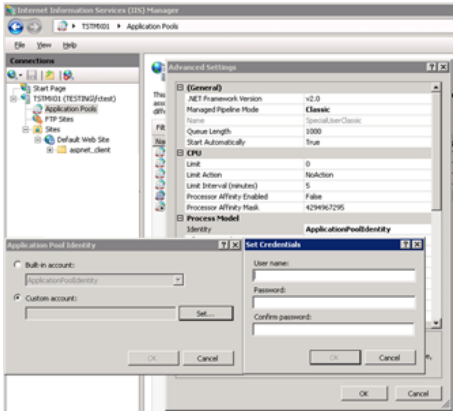
There is a key in the Vault service configuration file which allows the Vault service to verify that the calling application is allowed to access the service. In the setup with an IIS, this is done by setting the application pool user to a specific account. The easiest way to do this is to create new application pools for integrated (managed pipeline mode).



Application Pools Screen

- Then open advanced settings on the pools
- Click on the button next to identity
- Click on Set...

Preferably choose a domain user account or, if domain accounts are not used, a local user. [Domain]\[User] for a domain account Or [User] for a local account



Accounts

Then go to the applications (Web site and web service) under Default Web Site and choose basic settings and select the appropriate application pool. Now when the web applications are running under a specific account, this account has to be specified in the configuration file of the Vault NT service. If the account is a local one, the same account has to be created on the machine running the Vault NT service. **It is important that the password is the same on both machines.**

Exposing Vault Web Services

In order to expose web applications to external systems or users from outside the domain on a Vault that uses Windows authentication, it is necessary to enable and configure oneToOneMapping in IIS which allows mapping a certificate to a windows account. The procedure is described in this page. Typical external system which can use such WS are a remote public webpage or the AIT Portal. Those external systems connect to the CIS through a web service hosted in IIS. The virtual directory must point to the project folder (source code) named "VaultWS".

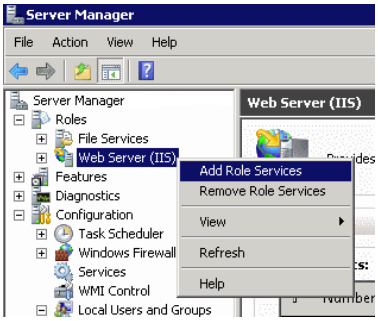
The following steps must be executed in order to expose the CIS web service to external systems:

1. Adding role services "IIS Client Certificate Mapping Authentication"
2. Configuring the web service application
3. Creating the Windows user that will be mapped against the X509 client certificate
4. Creating the client certificate
5. Making the client certificate valid for IIS
6. Exporting the client certificate to base64 format
7. Setting up the client certification mapping authentication in IIS
8. Setting ACL on the web site folder and files for the mapped user
9. Configuring the virtual directory of the web site
10. Creating the Inextor CIS user on the CIS web site

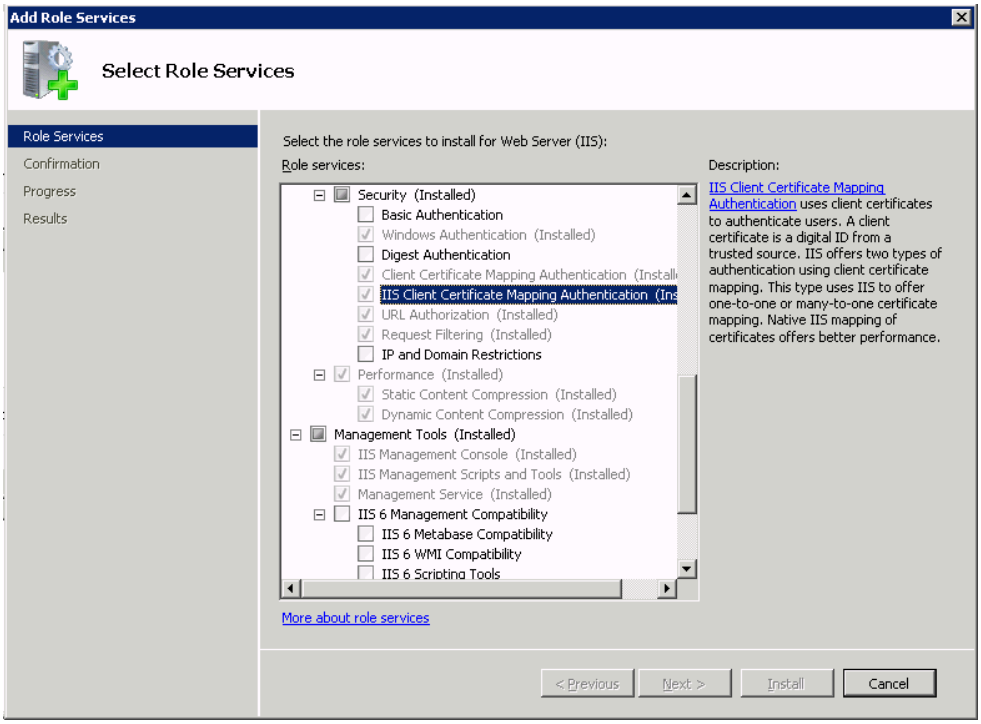
Adding role services "IIS Client Certificate Mapping Authentication"

In order to enable certificate mapping authentication, it is necessary to add the role service "IIS Client Certificate Mapping Authentication" on the server hosting the Vault web application.

In Server Manager, navigate to the node "Web Server (IIS)" under "Roles", right-click ans select "Add Role Services" as depicted by the image below:



In the "Add Role Services" window, check "IIS Client Certificate Mapping Authentication" and click "Install":



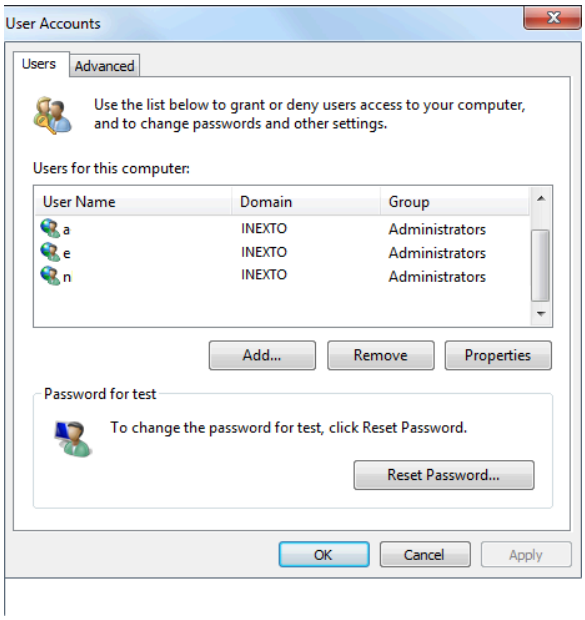
Configuring the web service application

The web service is configured by editing the file **web.config**, following the same principles as for the **web.config** file of the CIS web site:

```
<SelectedConfigUnits>
  <Unit>VaultserviceRemote</Unit>
  <Unit>LeavingAppDomainInvoker</Unit>
  <Unit>LoginWorkflowRemote</Unit>
  <Unit>LogoutWorkflowRemote</Unit>
  <Unit>SecurityFacadeRemote</Unit>
  <!-- Add credential Providers -->
  <Unit>CertificateCredentialProvider</Unit>
  <Unit>SSPICredentialProvider</Unit>
</SelectedConfigUnits>
```

Creating the Windows user that will be mapped against the X509 client certificate

The user can be a local user or a domain user. In the example below, it is a local user of the machine pmichlauw07817.



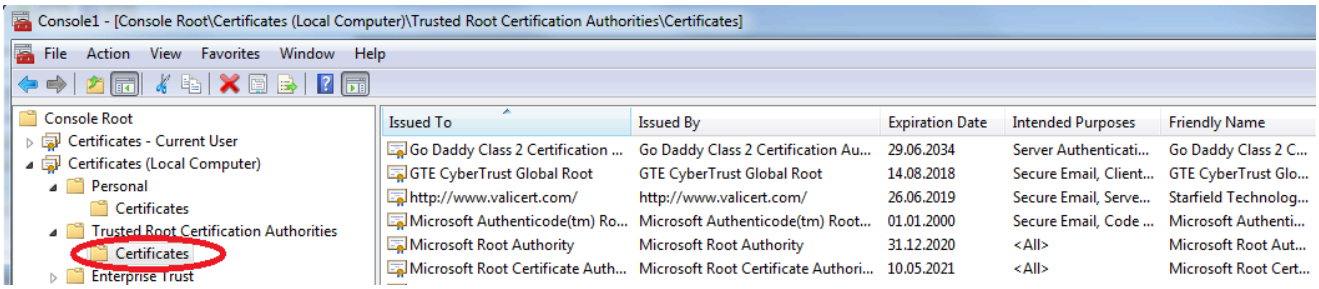
CIS Local User

Creating the client certificate

The client certificate can be generated by any certification issuer tool.

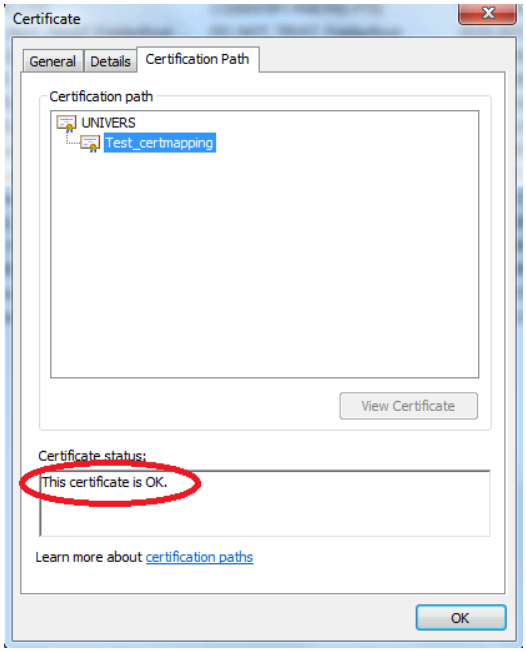
Making the client certificate valid for IIS

The client certificate must be signed by a valid certificate authority (CA). The root certificate of the certificate chain must be a member of the “Trusted Root Certification Authorities” in the Local Computer Windows Store for both the client machine and the Vault server.



CIS Trusted Root

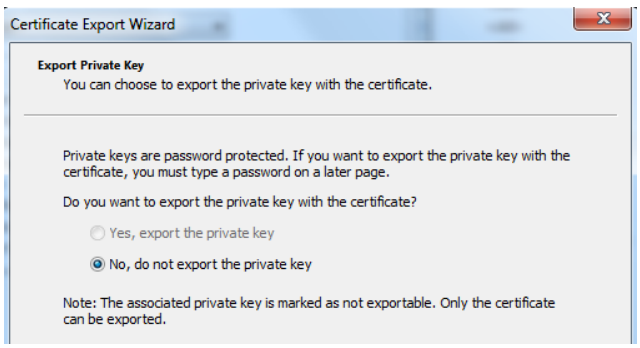
You can verify the validity of the client certificate by installing it into the Current User Windows Store and click on the tab “Certification Path”:



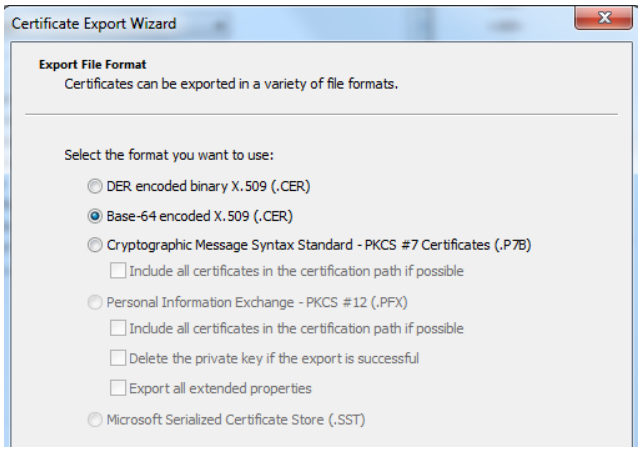
CIS Certification Path

Exporting the client certificate to base64 format

You can go to the tab “Details” of your client-certificate and click the button “Copy to file...” to run the “Certificate Export Wizard”. This will allow exporting the certificate to base64 encoding (cer file).



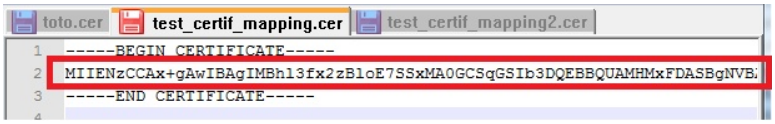
Certification Export Wizard



Certificate Format

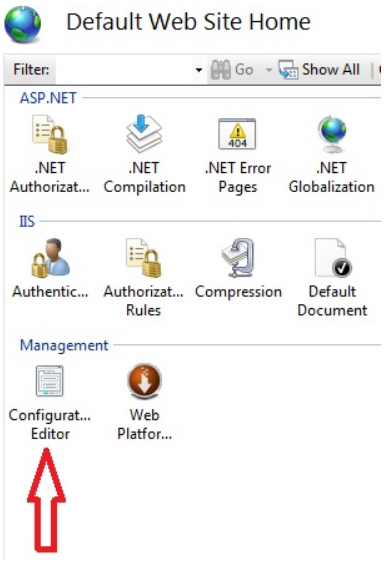
This will create a file with extension .cer. Open the file into notepad++, and remove all the line breaks and the -----BEGIN CERTIFICATE----- and -----END CERTIFICATE----- .

```
-----BEGIN CERTIFICATE-----
MIIENzCCAx+gAwIBAgIMBhl3fx2zBloE7SSxMA0GCSqGSIb3DQEBBQUAMHMxFDASBgNVBAYTC1N3aXR6ZXJsYW5kMQ0wCwYDVQQIEwRWYXVkMREw
DwYDVQQHEwhMYXVzYW5uZTEUMBIGA1UEChMLVU5JVkVSU19ZR0ExDTALBgNVBAsTBFJvb3QxFDASBgNVBAMTC1VOSVZFU1NfWUdBMCAXDTEyMDYx
NzIzMDAwMFoYDzIwOTkwMTAxMDAwMDAwWjBzMRQwEgYDVQQGEwtTd2l0emVybGFuZDENMAsGA1UECBMEVmF1ZDERMA8GA1UEBxMITGF1c2FubmU
xFDASBgNVBAoTC1VOSVZFU1NfWUdBMQ0wCwYDVQQLEwRSb290MRQwEgYDVQQDEwtVTklWRVJTX1lHQTCCASIwDQYJKoZIhvcNAQEBBQADggEPADC
CAQoCggEBANPfZbZa44VebVZJdyMhMpONxFv3kSu+UCQ6OXD2KG2Xapn4wGTLFejQ+SkBHLYuFzAQ0tlcPRDRb52+oN82VoMDWU1itfYiA7VcUSNUhXTEb
5ndUwjh9j5FqKK1aYWPbfADbeZ1wiBS8cG9JSMFwPQ5qBpJd3X49g60kKAHbgASoVFrBRrqM1YlYglDIcjJWIl+kn26hj1yfRoxDSWtblUcw46Qf8ZkE/puy
MGurs0e1204M0vKD8lwVkg34SRuvYmXO8qKrIPC6tfuqGlREnHuBk4gh72c6FxsRUgCLm1Wg9uWWA5qK0hv/ZriMBbA6pEVZSdPpZ5/lbfa9kyECAwEAAa
OByDCBxTASBgNVHRMBAf8ECDAGAQH/AgEDMIGuBgNVHQkBAQAEgaMwgaAwgZ0GgZg8TGljZW5zZUJhc2UgeG1sbnM6eHNkPSJodHRwOi8vd3d3Lncz
Lm9yZy8yMDAxL1hNTFNjaGVtYSIgeG1sbnM6eHNpPSJodHRwOi8vd3d3LnczLm9yZy8yMDAxL1hNTFNjaGVtYS1pbnN0YW5jZSI+PFZlcnNpb24+NC4zLjAu
MDwvVmVyc2lvbj48L0xpY2Vuc2VCYXNlPjEAMA0GCSqGSIb3DQEBBQUAA4IBAQACbl07hm/TCddNTVNijlBrzteSQDvkkAaxReMQ6av3fiobrERBSD4erqg
2mJvtozVzJN60BvT2VR5CojQ/E+fRaqpSxEERIngtKYe669dXn4SEGy1JBdKu+kE2Atymuyh2ZZOuNelapNFdRI+PzioKjU7eZ4LvJq6Mg7u7ArwefRNrhij
DPIsXOQfcYhJmXQQOnHXBugpWpiO2yqW05qTj8erE/HI8tLdtoXv46G0/yDrACefU6PW3WySS+r90MEi6JfQ3KQk7pps8MS9QeShyKpxuJTd0MCwKqJ219b
mRNQ5RnR4La5FLnUjxPq8SqkEdN1TygsDt0mcVcN8Fe
-----END CERTIFICATE-----
```



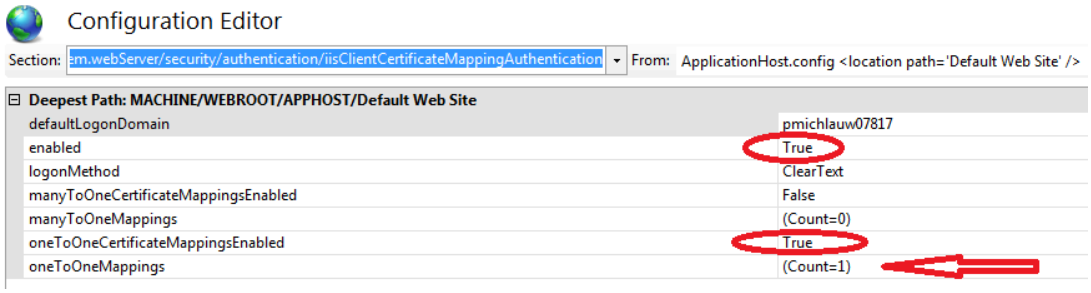
Certificate Content

Setting up the client certification mapping authentication in IIS

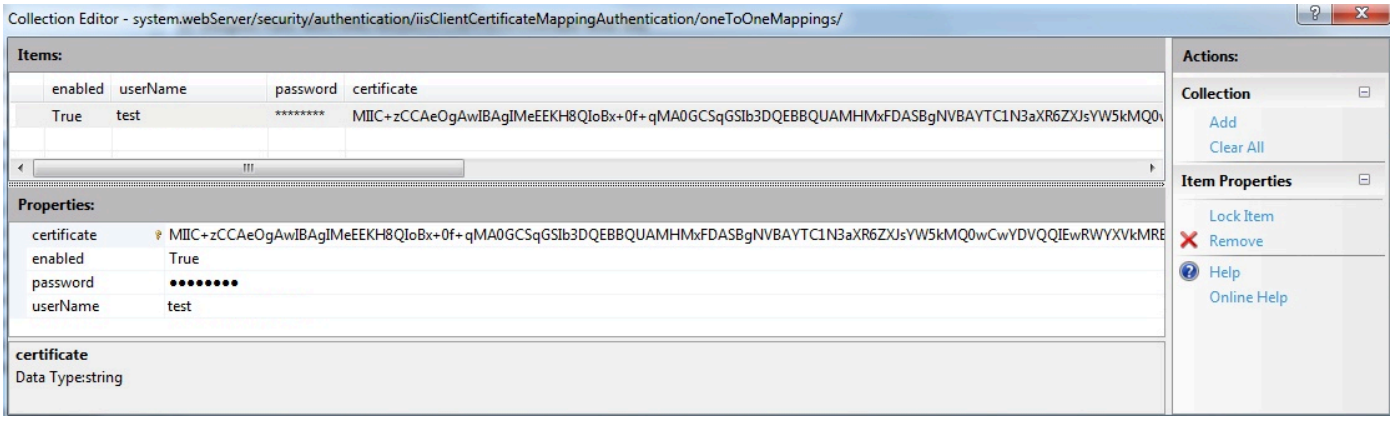


IIS Configuration

IMPORTANT: make sure that you select the web site when clicking the « configuration Editor ». If you select the IIS top node or any virtual directory, you can still click the "configuration Editor" and edit the settings but they will just be ignored.



IIS Configuration



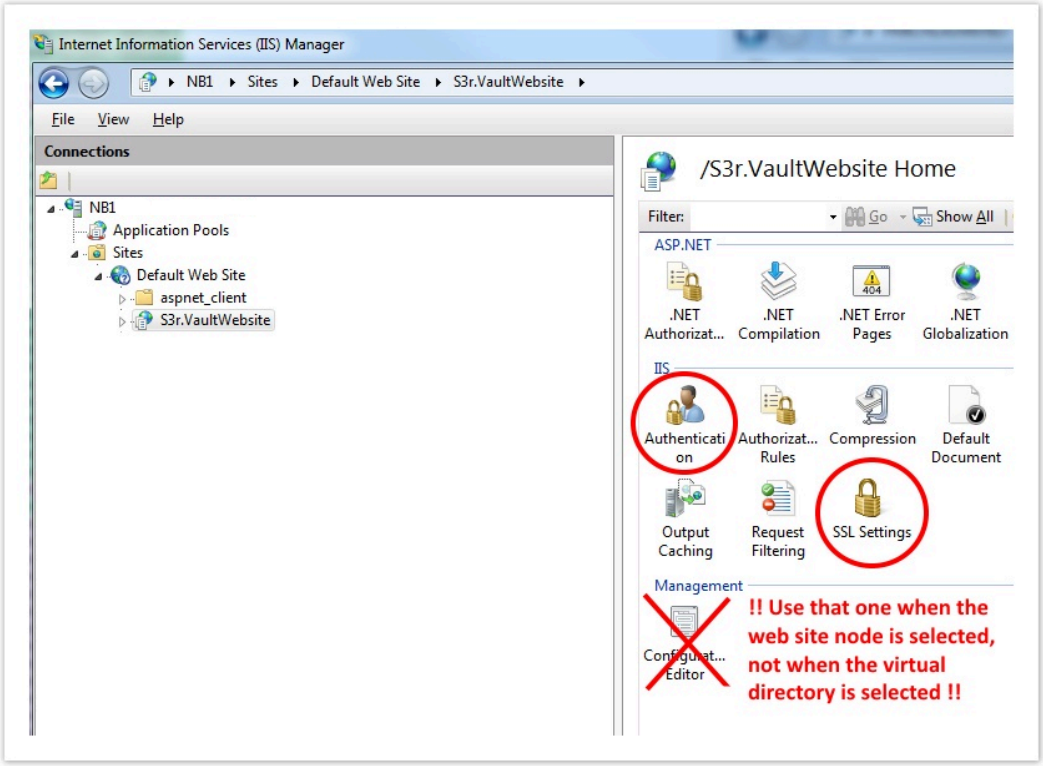
IIS Configuration

You must copy the whole base64 string of the certificate into the field "Certificate".

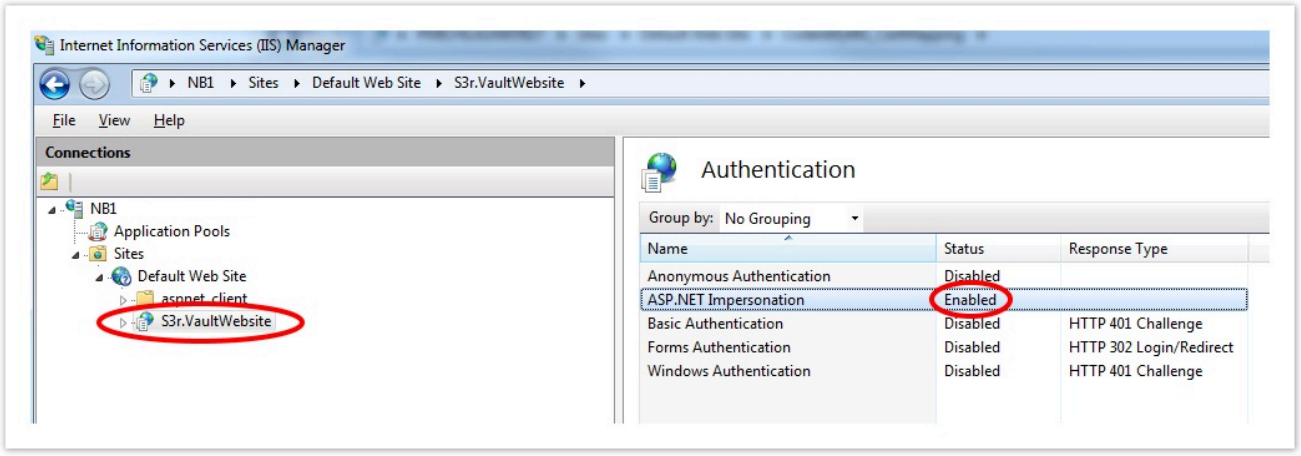
Setting ACL on the web site folder and files for the mapped user

It is important that the user has read-access permissions (ACL) to the files of the web site in NTFS otherwise you will receive HTTP ERROR 401.3. You should give read permissions to the user on the folder of the web site.

Configuring the virtual directory of the web site



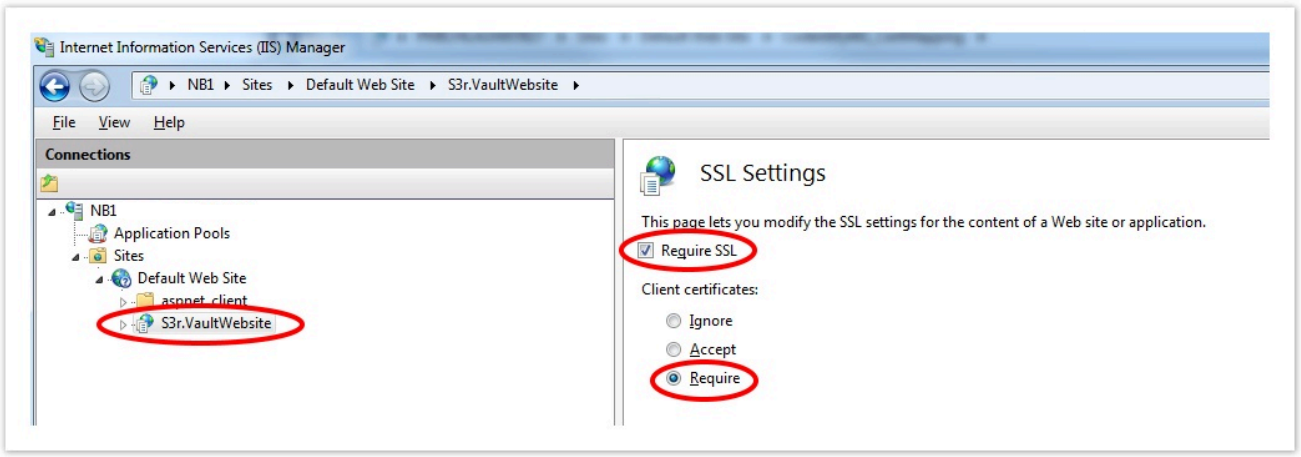
IIS Configuration



IIS Configuration

Note: since we use Client-certificate mapping authentication, we can disable:

- Anonymous authentication
- Windows authentication



IIS Configuration

Creating the Inextor CIS user on the CIS web site

Browse the CIS web site, and click on the entry Users, under the menu Accounts. Click the “New user...” button. Fill the fields as follows:

- The Actor Code must be an actor that has the GOV role (i.e. the Administrator of the CIS).
- The Actor Code Login must be the windows user account mapped to the client-certificate.
- The Type must be set to “WindowsUser”.

Alternatively, it is possible to set the Actor Code Login to a windows group name and set Type to “WindowsGroup”. In that case, the windows user must be a member of the windows group.

Finally, the permissions for the external system must be granted as follows:

- CheckSUPI: List, Create, Edit, Delete, Manage, Validate
- CheckUsed: List, Create, Delete, Validate
- CheckOrigin: List, Validate
- CheckPermutation: List

Migration (Configuration)

A Vault could support different versions of Gates. On the Vault, the version of the MSK (that would be issued to any Liz requesting one) is also defined. Migration to a newer version must always be undertaken top-to-bottom (Vault first, then Gate, and finally Lizs). A Gate can be connected to multiple Vaults, and a Vault can be connected to multiple Gates; therefore it is important to ensure correct versioning of the Gate license files that are generated on a Vault so that they are compatible with the targeted Gate.

MSK and Gate certificate version support

The version of the MSK and Gate certificate version is defined in the configuration file VaultService.exe.config with the setting LicenseVersion, see below.

```
<Cvs.Vault.Properties.Settings>
    <Cvs.Certificates.Common.Properties.Settings>
        <setting name="KeyLength" serializeAs="String">
            <value>2048</value>
        </setting>
    </Cvs.Certificates.Common.Properties.Settings>
    <Cvs.Properties.Settings>
        <setting name="LicenseVersion" serializeAs="String">
            <value>3.0.0.0</value>
        </setting>
    </Cvs.Properties.Settings>
```

Check with your technical provider to know which value to have for the license version and which versions of Gate and Liz comply with this value, since the license version is not necessarily the same as the Inextor version of the Gate (or Liz). The license version defined will be shown on the Gate page of the Vault website, and the license version should be selected prior to renewing the Gate certificate. If the Vault should support multiple Gate certificate versions, then you need to configure this in a separate configuration file (GateLicenses.configunit), see **Multiple Gate Certificate Versions Support** below. However, you still need to set the LicenseVersion in VaultService.exe.config since it will be used by the Vault when creating MSK (or to be more precise, the certificate contained in the MSK). The certificate inside the MSK is used for signing PROD updates that is sent from Liz to Vault via the Gate.

Multiple Gate certificate versions support

If the Vault should support multiple license versions for the Gate, then this is configured in the file GateLicenses.configunit, see below.

```
<VaultserviceConfig>
    <GateLicenseVersions OnMiss="Add" OnExist="Replace">
        <GateLicenseVersionSupported DisplayedLicenseVersion="3.0.0.0">3.0.0.0</GateLicenseVersionSupported>
        <GateLicenseVersionSupported DisplayedLicenseVersion="3.1.0.0">3.0.0.0</GateLicenseVersionSupported>
        <GateLicenseVersionSupported DisplayedLicenseVersion="3.2.0.0">3.2.0.0</GateLicenseVersionSupported>
        <GateLicenseVersionSupported DisplayedLicenseVersion="4.3.0.0">4.3.0.0</GateLicenseVersionSupported>
    </GateLicenseVersions>
</VaultserviceConfig>
```

Check with your technical provider to know which value to have for the license version and which versions of Gate and Liz comply with this value, since the license version is not necessarily the same as the Inextor version of the Gate (or Liz). The License Versions defined in the this file will be shown on Gate page on Vault web site and license version should be selected prior to renewing the Gate certificate (see Vault Operation Guide). You could, if you want to have another value displayed on the Vault Web site by defining another DisplayedLicenseVersion. This is for instance the case for version 3.1.0.0 since license version 3.1.0.0 does not exists. Instead it uses version 3.0.0.0.

Security (Configuration)

Communication

The communication from users or factories (Gate) to the Vault Web site/Web service is secured (SSL). For having the communication secured, a SSL-certificate for the Vault address must be created during the Installation process of the Vault. Preferably the Vault address should be a signed by CA (e.g. VeriSign) for government Vaults. For private Vaults a self-signing tool could be used (for instance the /wiki/spaces/INEXTOR/pages/395812035).

Renewing SSL-Certificate: It is possible to renew the SSL-certificate, but the following should be considered:

- The Gate might validate the certificate that is used for signing. By using another certificate for signing, if that certificate is not installed on the Gate server, the Gate will not connect to the Vault.
- For the website, it is still possible to login with account certificate, but when accessing the website the user will get a security warning that the certificate is not trusted, if the certificate used for signing is not installed.

Hence, if you use the same certificate for signing, then you do not need to inform any actor in the system. However if you change the certificate for signing you should inform all actors about it and supply them with the certificate that was used for signing.

Vault certificate

Certificates are created on the Vault Web site for the Gate and for accounts. The certificates are signed with the Vault certificate. The Gate will then use its certificate to authenticate itself on the Vault but the Gate certificate is also used for signing messages between the Gate and the Vault. The Vault will sign messages to the Gate with the Vault certificate. The Vault certificate is created with a Inextor tool, the /wiki/spaces/INEXTOR/pages/395812029.

Renew Vault Certificate

It is a complex task to renew the Vault certificate.

Renew Vault certificate using the same Vault name

If you renew the Vault certificate, Gates will no longer be able to connect to the Vault. All certificates on the Vault have to be renewed. Even though it is possible to change certificate, it is a complex task to execute the change. A detailed procedure has to be defined prior to deploying a new Vault certificate.

Renew Vault certificate using another Vault name

This requires even more considerations, since all registered SKUs will be invalid, since a SKU is partly defined by Vault name. A detailed procedure has to be defined prior to deploying a new Vault certificate with another Vault name. Practically, this is almost an impossible task. The expiration date for the Vault certificate is year 2099 and hence does not have to be considered. For a government that wants to change Vault certificate, the government should work with its implementation partner to coordinate such a procedure.

Vault Certificates, summary

The following certificates are deployed in the Vault Windows Certification Store at deployment:

Certificate	Type	Description
Inextor root	Public	The root for this Inextor universe. Created by Vault owner. Used for signing Inextor TP certs.
Inextor TP	Public	A technical provider within the universe. Created by Vault owner. Used for signing Inextor TP certs.
Inextor Vault	Public	The Vault certificate
CA root	Public	The root for CA universe. Used for signing the CA intermediate
CA intermediate	Public	Certificate used for signing SSL-certificate.
SSL Certificate	Private	Certificate used for establishing a secure link to actors (Gate, Account and Users)

The following certificates are deployed in Vault SQL Server database at deployment:

Certificate	Type	Description
Inextor root	Public	The root for this Inextor universe. Created by Vault owner. Used for signing Inextor TP certs.
Inextor TP	Public	A technical provider within the universe. Created by Vault owner. Used for signing Inextor TP certs.
Inextor Vault	Private	The Vault certificate
Admin	Serial number	Account for GOV-role that is administrator of the system. Only serial number is stored for Account certs,

The following certificates are deployed in Vault SQL Server database during its life cycle:

Certificate	Type	Description
Additional root	Public	Root certificate for other Inextor universe.

Certificate	Type	Description
Additional TP	Public	TP certificate from other Inextor universe.
Inextor Liz certs	Public	One for every Liz registered on the Vault
Gate certs.	Public	One for every Gate on the Vault
Accounts	Serial	One for every account created on the Vault
MSK certs.	Public	One for every MSK registered on the Vault

Protection

Data Protection

By setting the key CryptoDatabaseSafeType to None, all certificates with private keys are stored without encryption in the database. This means that anyone getting hold of the Vault database are able to extract all certificates from the database. By using encryption (CryptoDatabaseSafeType set to DoubleRSA), then all certificates that are stored in the Vault database that has a private key are protected. This means that only those that have the keys for encrypting the certificates will be able to extract certificates with private keys from the database. For using data protection, you must define the following keys in VaultService.exe.config (see Create Keys for Data Protection for details on how to create the keys):

```
<add key="CryptoKeyProtected" value="true"/>
<add key="GlobalRSAPublicKey" value="XXX"/>
<add key="CryptoSpecificRSAKey" value="XXX"/>
<add key="CryptoDatabaseSafeType" value="DoubleRsa"/>
```

For disabling data protection, you define the key CryptoDatabaseSafeType to none:

```
<add key="CryptoDatabaseSafeType" value="None" />
```

Central Validation

To enable the processing servers to raise an exception in case of validation error, set the key ValidationRaiseException in VaultService.exe.config to true. Enabling central validation will guarantee that the Vault does not store data that does not satisfy the Vault business rules. Disabling the raise exception will only log validation errors to the Vault service log file.

Limiting access to the Vault NT Service

By default, any client can connect to the Vault NT service. You could limit the access for clients (e.g. Web site or Web service) by defining that only clients that authenticate themselves with a Windows user account or group will be granted access to the NT service by using any of the following keys in VaultService.exe.config:

- VaultRemotingWindowsGroup
- VaultRemotingWindowsIdentity

The format for the values for the keys is: DOMAIN\name COMPUTERNAME\name Where name is either the group name (when using VaultRemotingWindowsGroup) or user account name (when using VaultRemotingWindowsIdentity).

Patch for RRP

Introduction

The RRP production stores information in a specific table that was previously not used on the Vault (before 10.1.01882) and therefore not migrated. The RRP code check will work on the Vault without the migration of this table but will not provide all the information. Below is the patching steps necessary to have those data on the Vault. The patch will migrate the data from the Checker to the vault and the new version of the continuous migration service will include this table.

Please note that the patch does not include the creation of the new table. and needs to have version 10.1.0.1882 or higher for it to work.

Patching steps

1. Stop the continuous migration service
2. Update the binaries with the new version of the service. Make sure to keep the config files (both *Inexto.S3r.ContinuousMigrationService.exe.config* and *log4net.config*)
3. Change the service config file to add : **<add key="patchForRrp" value="true" />** in the appSettings part
4. Start the service
5. Check the logs : it should display *RRP patch operation : operation completed with success*
    - Location of the log file can be found in *log4net.config*
6. Stop the service after one run - log displays : *Continuous migration successful [...]*
7. Remove the line added in step 3 from the service config file.
8. Start the service, check the logs that no error occurs for one run

Configuration Protection Tool

There might be additional elements that need protection in the configuration file of the Vault NT service (VaultService.exe.config), for example, the Database connection string. This can depend on different types of providers for authentication or modifications for a particular environment. In these cases the ConfigurationProtectionTool can be used to encrypt these elements.

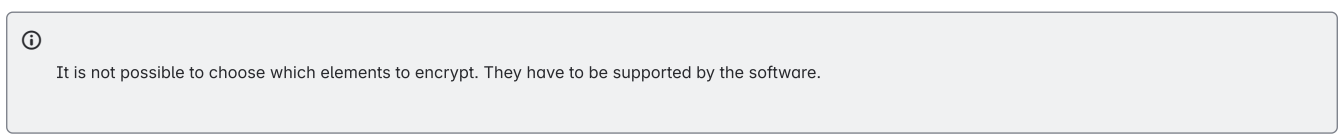
Example – to encrypt the Database connection string:

```
<add key="CertificateDBConnectionString" value="Data Source=#########;Initial Catalog=#########;Integrated Sec
<add key="ProtectedDBConnectionString" value="false"/>
```

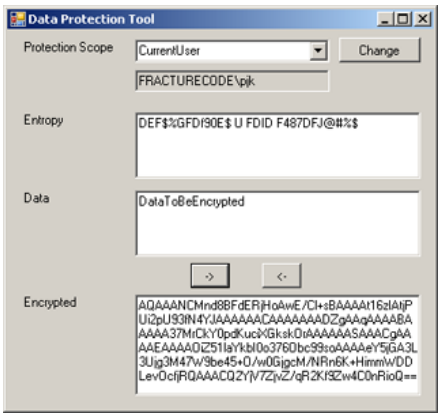
1. Encrypt the value (marked as bold).

2. Set the parameter ProtectedDBConnectionString to true.

```
<add key="AQAAANCMnd8BFdERjHoAwE/Cl+sBAAAA3Hg53UA6mkGaSFqCZbRyqQAAAAACAAAAAAADZgAAqAAAABAAAAAvRp/yU+kam+TU160z
<add key="ProtectedDBConnectionString" value="true"/>
```



It is not possible to choose which elements to encrypt. They have to be supported by the software.



Source Control: SVN

The DataProtection tool uses Microsoft Data Protection (the same as the CertificateManager).

The encryption algorithm used is unique for each server and also unique for the account under which the Vault NT Service is running under. This has the consequence that you have to create encrypted data for every instance of Vault NT Service.

Furthermore, if the account used for running the Vault NT Service is changed, you have to create the encrypted data again.

The user running the program must be the same as the one running the Vault Service.

Encrypting data

Make sure that you are logged on as the user that will be running the Vault Service on the same computer that will be hosting the Vault Service. If it is not possible to log on to the computer with that account, start the ConfigurationProtectionTool by running the following command from the command prompt in the same directory as the ConfigurationProtectionTool tool (replace VaultserviceUserName with the account that will be running the Vault Service):

```
runas /user:VaultserviceUserName .\ ConfigurationProtectionTool.exe
```

Entropy: The entropy is an additional security to make sure that one application cannot read configuration elements of other applications running under the same user on the same computer. Since this separation can be done using different users for other applications potentially using Microsoft Data Protection, there is no need to change this value. Therefore the suggested value is the default one.

If the entropy is changed, the configuration file for every Vault service using the encryption will have to be modified to specify the new entropy. Also, since the CertificateManager uses the same Data Protection, the configuration file of the CertificateManager must also be changed to use the new entropy.

The following entry has to be added under <appSettings> in both VaultService.exe.config and CertificateManager.exe.config to change the entropy.

```
<add key="DataProtectionEntropy" value="DEF$%GFDf90E$ U FDID F487DFJ@#%$" />
```

> When changing the entropy on an existing configuration, the data protection has to be renewed for the RSA protection and all other protected configuration elements.

Data: Enter the data to be encrypted, e.g. the Connection string to the database

Press the → button to encrypt.

Encrypted: Here the data that are encrypted will be displayed. These data must be entered for the parameter, for which the encryption is done, in the configuration file (e.g. VaultService.exe.config)

Decrypting data

During troubleshooting it might be a good idea to investigate the current configuration. In such a case the parameter(s) can be decrypted to show the value being used by the Vault Service.

Use Entra ID / Azure AD to authenticate users

Introduction

The Vault Web site, as well as the Vault/Gate/Liz/Inexpress administration programs support the use of Azure AD (now called Entra ID) to authenticate users and get their group memberships.

When this is configured, a user coming to the Vault Web site is redirected to Azure, where actual authentication happens. At Azure, any policy can be configued, e.g. multi-factor authentication. When the user successfully authenticates on Azure s/he is redirected back to the Vault Web site.

The administration programs for Vault, Gate, Liz and Inexpress can also be configued to use this same authentication mechanism. As they are desktop applications and this mechanism is Web-based, they open a Web page where the user can be authenticated.

A big advantage of this mechanism is single sign-on (SSO). If the user had previously used any other application compatible with Azure AD authentication, this authentication is transparent. The user may note a small delay while the browser is redirected to Azure and back, but s/he does not need to enter any credentials again.

On the Vault Web site, setting up Azure authentication removes Windows authentication, but keeps the ability to enter an explicit username and password.

General Setup

There are 4 main steps to enable Azure AD authentication.

1. Create an application registration on Azure
2. Reconfigure the security on the Vault Web site
3. Configure the Vault Web site to support Azure authentication
4. Configure an administration program to use Azure authentication

These steps will now be covered in detail.

Detailed Setup

Application Registration

Azure will only accept to authenticate users if an administrator creates what is called an 'app registration'.

This entity declares:

- Who can be authenticated. More precisely, from which company can employees use Azure to be authenticated.
- What rights the authenticated user will have.
- Which Web site can ask Azure to authenticate users

The page Create an Application Registration on Azure has all the detailed steps to create an app registration on the Azure portal.

But there are already 2 existing registrations which can be used. It is recommended to use one of them, but if the customer insists on having its own registration, let him go ahead.

When he has finished the creation, he needs to give us:

- his tenant identifier (a GUID, i.e. an identifier in the form of xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx),
- the application id of the registration (again a GUID),
- The client secret created in steps 9 to 11 of the creation instructions.

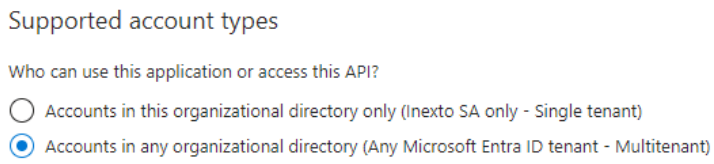
Note that Inexto needs to provide the Web site public address, as it is required in step 3.

If an app registration is reused, this information is available from the infrastructure or development team. In this case, the only required step is to add the public Web site address to the 'Redirect URI' section of the app registration (see step 3).

The 2 existing registrations serve 2 different purposes:

- If only Inexto employees need to use Azure authentication, the registration named 'Inextor Credential Provider' should be used.
- If employees from more than one company need to use Azure authentication, the registration named 'Inextor Vault' should be used.

On the Azure side, the only difference between these 2 registrations is the choice selected on step 3:



Reconfigure Security on Vault Web Site

On the Vault Web site, users and groups, when using Azure authentication, are still considered as 'Windows users' and 'Windows groups'

But there is a difference in naming:

- Users are identified by their e-mail address, instead of the Windows way with *domain\user name*.
- Groups are identified by their simple name, instead of the Windows way with *domain\group name.*

Note that this is the way Windows and Azure authentications work and cannot be changed by Inexto.

As a consequence, the security configuration on the Vault Web site must be changed.

Every user who is assigned a right must get a new assignment, using the e-mail format for his/her name.

Every group which is assigned a right must get a new assignment, using the simple name to identify the group.

Configure the Vault Web site

Enabling Azure AD authentication requires to update the Web.config and bin\VaultConnection.configunit files in the installation directory of the Web site (typically Inexto.S3r.VaultWebsite2).

The Web.config file has comments to explain what should be done.

It is easier to search for "Azure AD" in the document, but here is a summary of the changes:

1. The Web site supports several authentication methods. Using one involves activating a specific component. For Windows authentication, the component is 'SSPICredentialProvider'. For Azure authentication, the component is 'AzureADCredentialProvider'. So, this step involves commenting out (thus deactivating) the first one and uncommenting (thus activating) the second one.
2. Configure ASP.NET to enable anonymous authentication, instead of forcing Windows authentication (this gets technical: the Web framework used to program the Vault Web site has no direct suport for Azure authentication, so we are asking it to let us handle authentication). This is done by replacing the *mode* attribute to '*none*' (from '*Windows*') on the *authentication* tag.

3. Configure the Web Server (IIS) to enable anonymous authentication (same reason as above). This is done by uncommenting a section of the file.

The bin\VaultConnection.configunit also has comments to explain what should be done.

Anyway, here is a description of the changes:

1. As when updating the Web.config file in step 1, it is necessary to activate the required authentication methods that the Web site uses to log in the user. Under the definition of the *eln.ClientSecurityHelper* component, it is required to comment out the Windows part and uncomment the Azure AD part.
2. Information about the app registration must be entered. This is done in the definition of the component named *Security.AzureADCredentialProvider*. There are 2 example definitions of this component in the file, one for each registration type (single- or multi-tenant, as explained above). By default, these definition are commented out.
   Uncomment one of them and enter the appropriate data, for the following tags:
   a. TenantId. In case of a single-tenant registration, this is the Azure tenant where the registration has been created. If reusing the existing registration, keep the value starting with 73d4101a (found in the first example), it represents the Inexto tenant. If using a multi-tenant registration, use the special value 'common' (like in the second example).
   b. ApplicationId. This represents the app registration itself.
   c. ClientSecret. This represents the secret generated during registration creation.
   d. PublicWebSiteAddress. This represents the URL of the Vault Web site. Care must be taken to match the exact the same letters as what was entered in the 'Redirect URI' section of the app registration, including casing and a final slash in the URL at the end. In other words: https://hello.world.com/ and https://Hello.world.com/ are not identical; https://hello.world.com and https://hello.world.com/ are not identical either.
   e. AcceptedTenantIds. This is necessary only in the multi-tenant case. It represents the list of companies (tenants) that have employees using the Vault Web site with Azure AD authentication. Typically, we need first the Inexto tenant id (which is already in the example file). If employees from the customer or their partners will use the Vault Web site (or the administration programs) with Azure AD authentication, an entry must be added for each tenant.

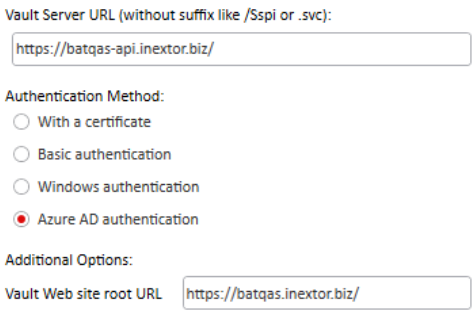
Try then to log in to the Vault Web site. It should redirect you to Azure and back after authentication, if everything has been done correctly.

Please make sure to have this part working before trying to configure any administration program.

Configure an administration program

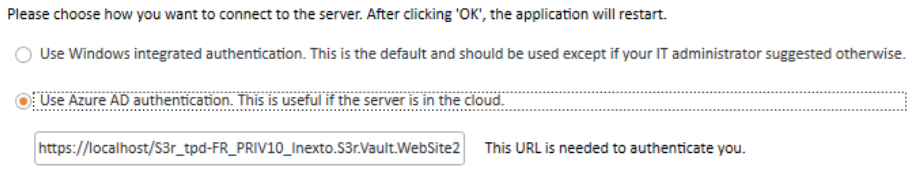
Vault

On the Vault administration program, activate the 'change connection' screen and set the authentication method to 'Azure AD authentication'. Enter the Vault Web site address, as necessary in this case.



Gate / Liz

On the administration program, activate the 'change connection' screen and set the authentication method to 'Azure AD authentication'. Enter the Vault Web site address, as necessary in this case.



Inexpress

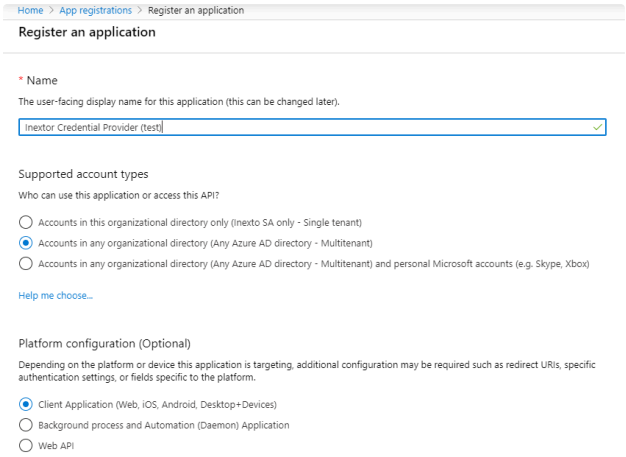
See https://inexto.atlassian.net/wiki/spaces/INEXPRESS/pages/1021575169 Request access

Create an Application Registration on Azure

This page shows the necessary steps to register an application in the Azure portal. Note that INEXTRACE is sometimes mentioned in these screenshots (it has the same requirements); please substitute it with INEXTOR instead.

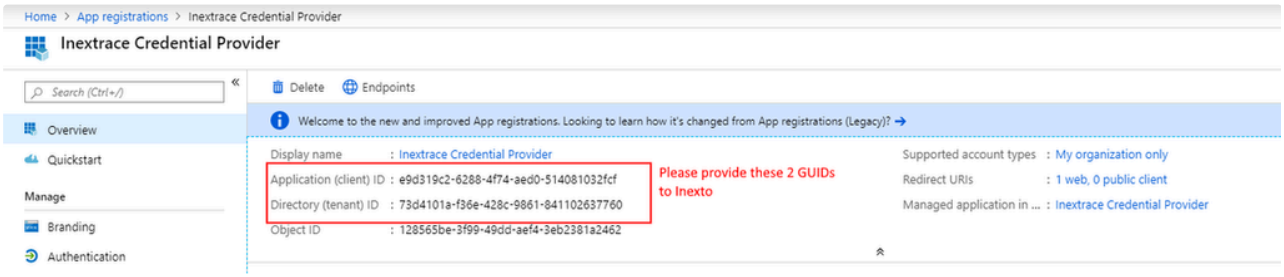
⚠ The user interface of Azure changes very frequently. The actual user interface may vary from these screenshots.

1. Go to Azure portal, App registrations, click on ‘New registration’

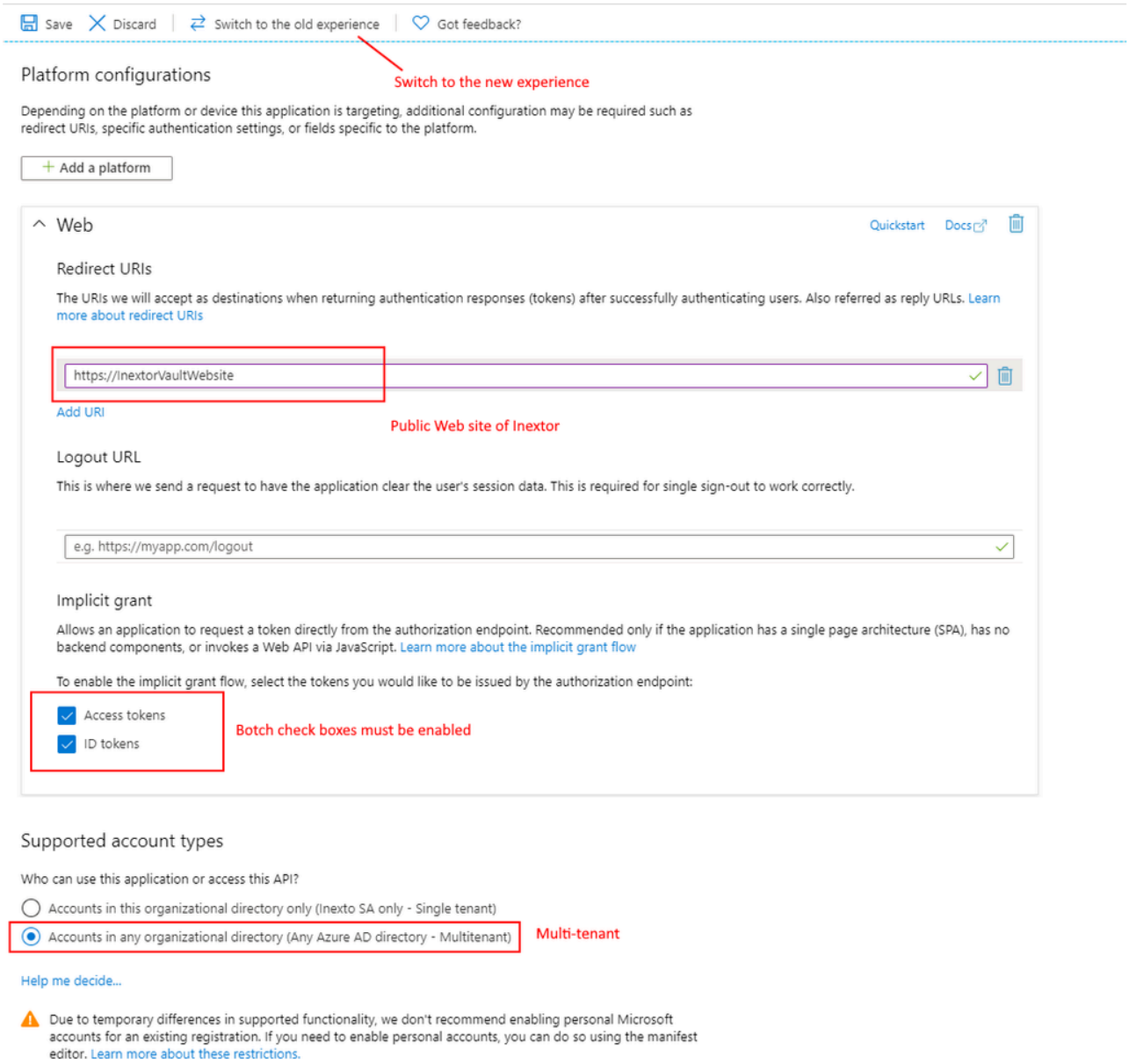


Inexto will provide the public Web site address. The address, set below under ‘Redirect URIs’ section, must match exactly what is declared in the ‘PublicWebSiteAddress’ parameter of the Vault Web site configuration.

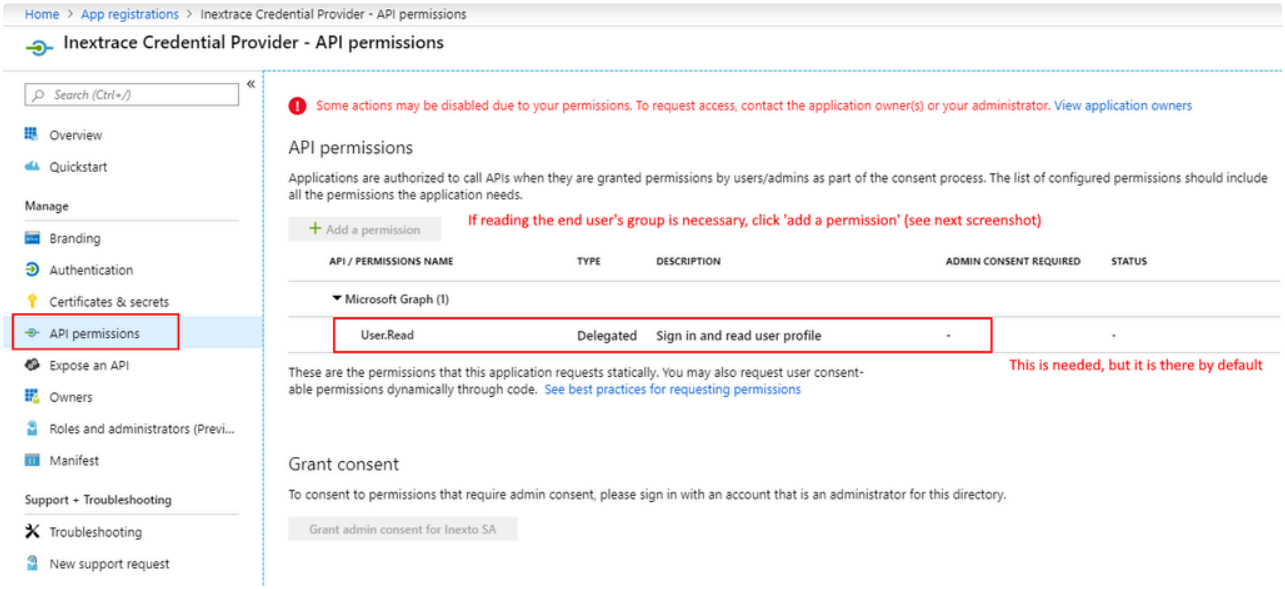
2. In the portal, go to the Overview page. Please give the 2 GUIDs to Inexto:



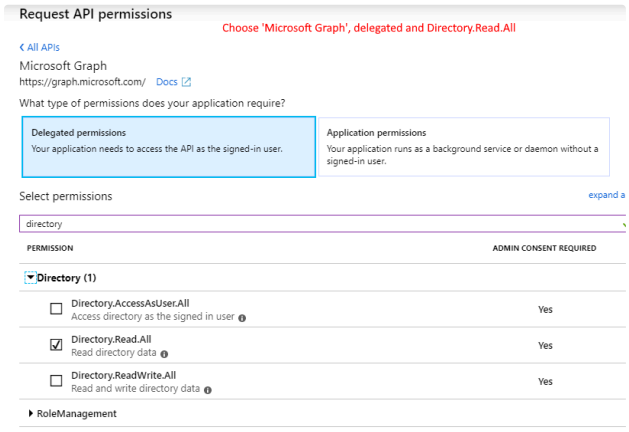
3. Go to authentication tab (switch to the new experience, if not already done), click on 'Add a platform', choose Web applications / Web and enable 'Access tokens' and 'ID tokens':



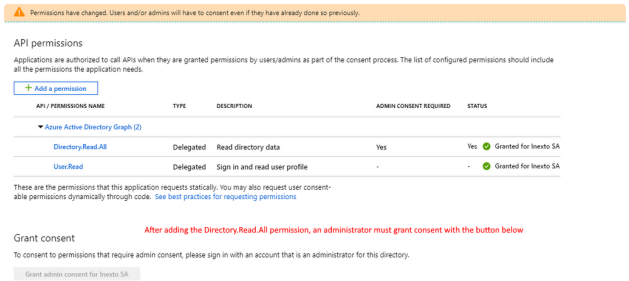
4. Go to API Permissions. There is 1 permission to add, as the groups list of the logged-in user must be read.



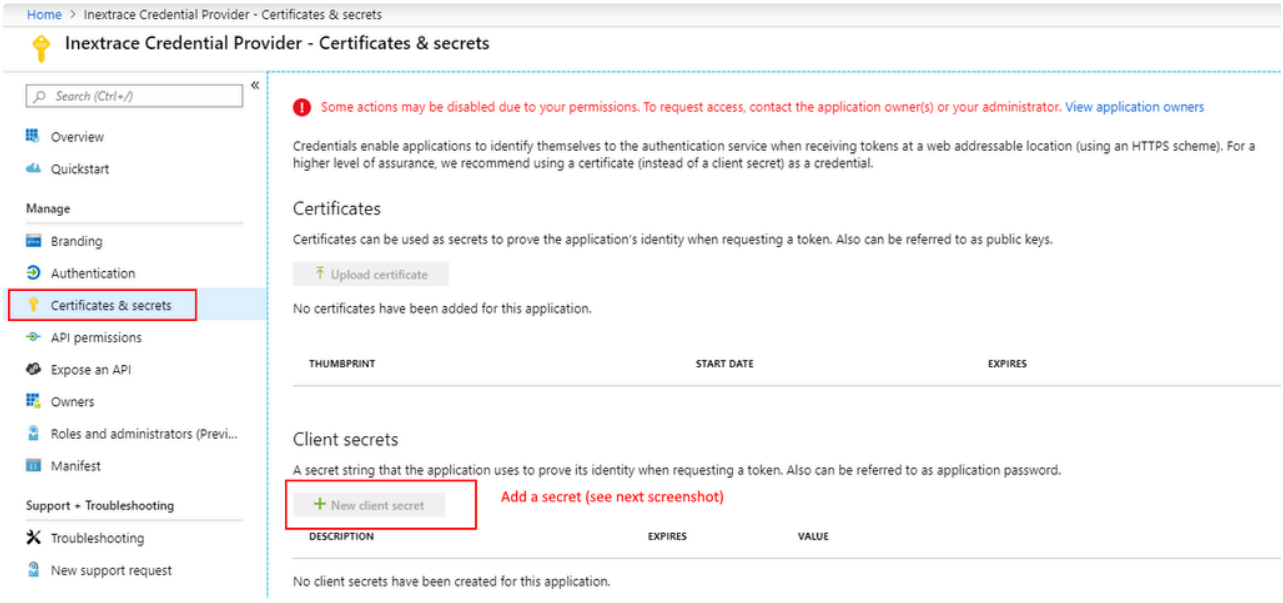
5. Click on 'Add a permission' and choose 'Microsoft Graph':



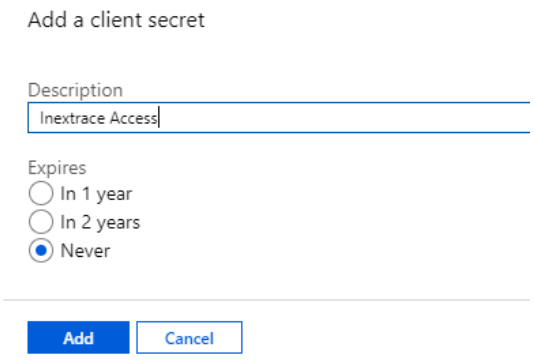
6. Choose 'Delegated permissions' as permission type and find and select 'Directory.Read.All' in the list of permissions.
7. Click on 'Add permissions'.
8. Click on 'Grand admin consent for...':



9. Go to 'Certificates & Secrets'. Click on 'New client secret':

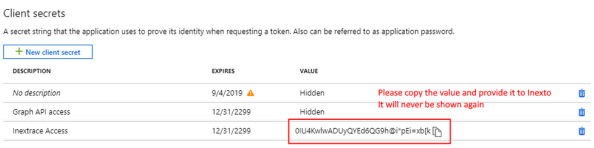


10. Give the secret a name and choose an expiration period; 'never' is recommended, unless you have in place a procedure to replace secrets. Note that the secret can be revoked at any time.



The secret is then displayed.

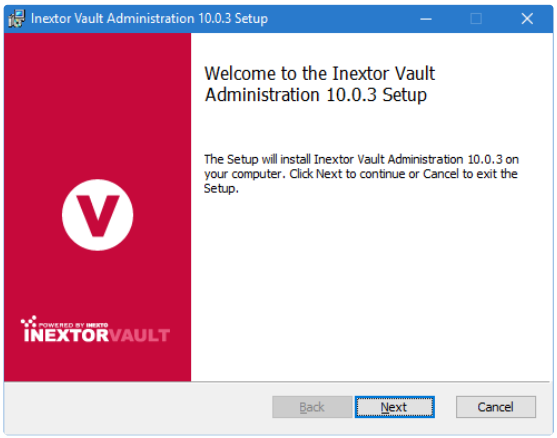
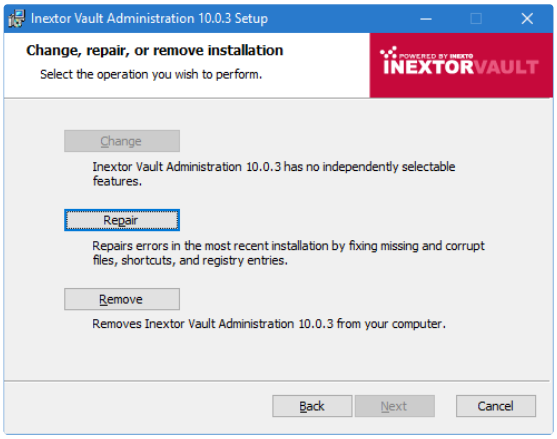
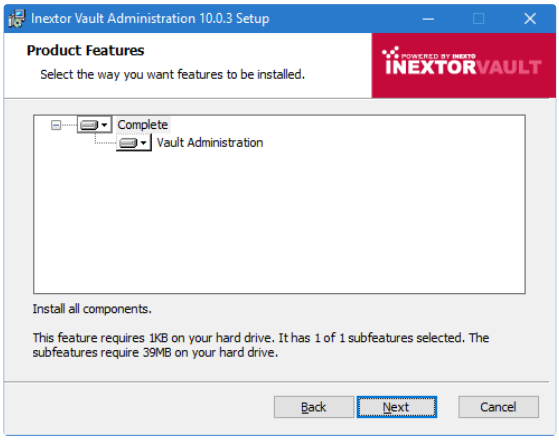
11. Take note of it and give it to Inexto. Note that it will NEVER be shown again:

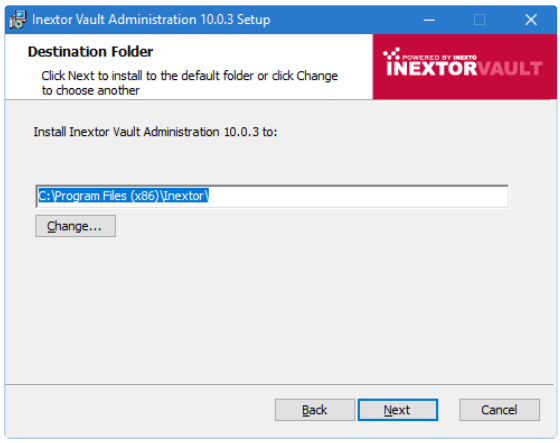
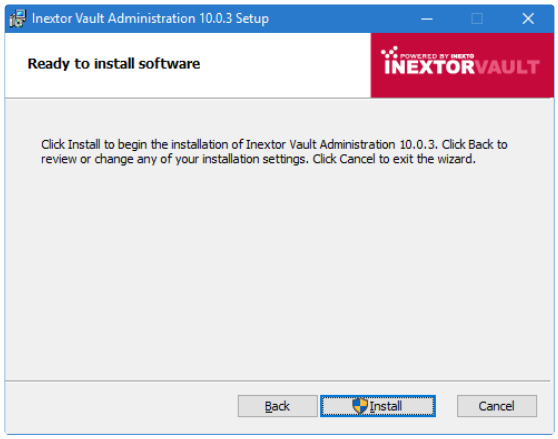
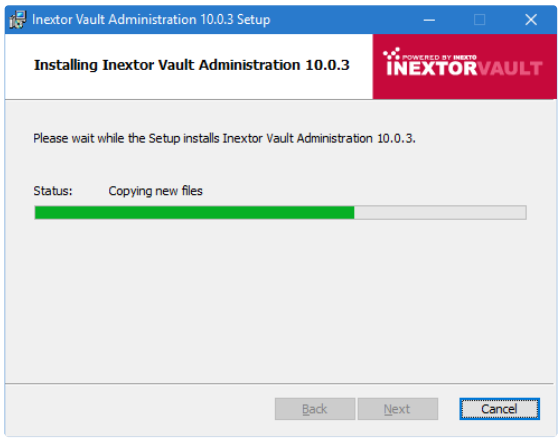
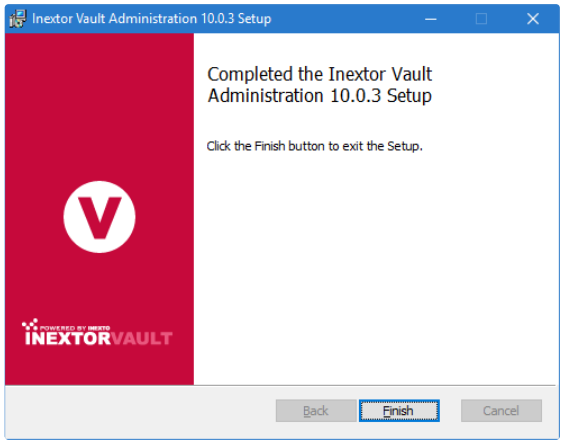


The application registration is now ready.

Vault Administration UI

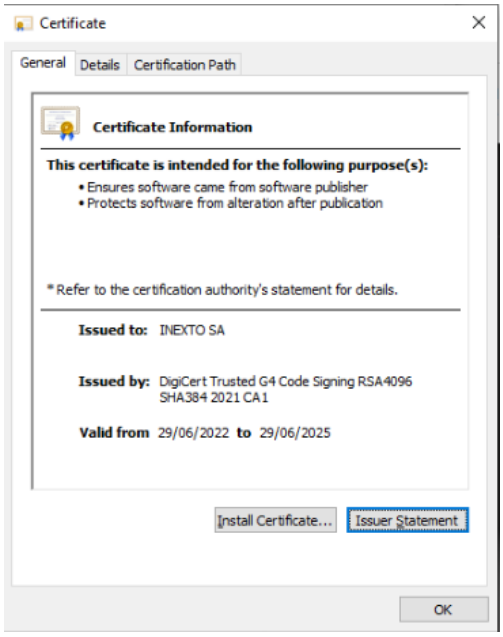
The following procedure describe the installation of the Vault Administration UI.

1.	Run the setup file **InextorVaultAdministration-<version>.msi**. Click **Next**.	
2.	In case of an existing Vault Administration installation the system will propose to do either a side by side installation or a complete upgrade. The preferred approach is to do a complete upgrade.	
3.	The installer contains one single component, Vault Administration. Click **Next**.	

4.	Use this window to change the destination folder. Click **Next** when you are satisfied with the destination folder.	
5.	Use the **Back** button to return to any configuration windows. Click **Install** to start the installation. Installation of the software package requires elevated privileges and you will be prompted to confirm you accept it.	
6.	Installation is now in progress...	
7.	Click Finish to complete the installation. Note that you have the possibility to display the installation log file.	

See Configuring the Connection to the Vault Server in order to configure the connection to the Vault Server

The Vault Administration is Digitally Signed.



IIS Configuration - Setup

The IIS configuration should be made on all those servers who will have a Web Site or/and the Web Service even if they are in multiple servers (presentation and business layer servers). This configuration procedure is valid on both Windows 2008 Server and Windows 2012 Server.

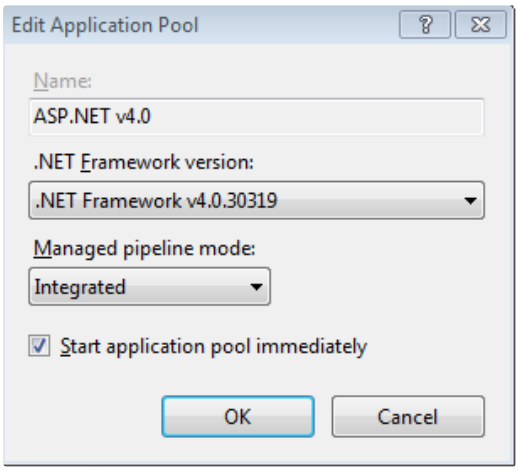
The structure of the document is as follows:

- Configuring Application Pool
- Web Service Application (WS)
- - Adding Application
  - Setting Secure Communication Parameters
  - Authentication Settings
- Website Application Not Using Windows Authentication (Web)
- - Adding Application
  - Setting Secure Communication Parameters
  - Authentication Settings
  - Setting Parameters for Certificate Negociation
- Website Application for Windows Users (WebWin)
- - Adding Application
  - Setting Secure Communication Parameters
  - Authentication Settings
  - Exposing Web Applications to External Systems Using Certificate Mapping
- Setting a Different Website for Web Services
- Starting IIS

Configuring Application Pool

To access the ASP .NET on the IIS Manager, perform the following steps:

- Click Start, Administrative Tools, and then Internet Information Services (IIS) Manager.
- Choose the root leaf (the one showing the computer name).
- Click on Application Pools.



ASP.NET application pools

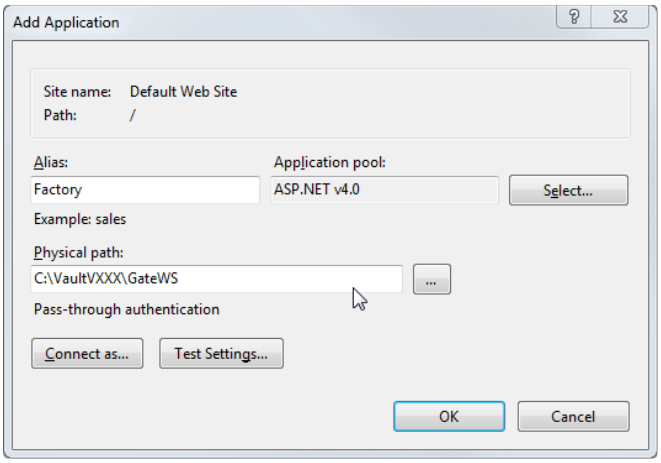
Make sure there is an application pool with the parameters shown above (.NET Framework version set to 4.0 and Managed pipeline mode to "Integrated"), potentially run by a service account defined previously. This is the application pool that you should select when adding applications for the website and web services.

Web Service Application (WS)

For all web service applications, do the following steps.

Adding Application

Right-click on Default Web site and choose Add Application.



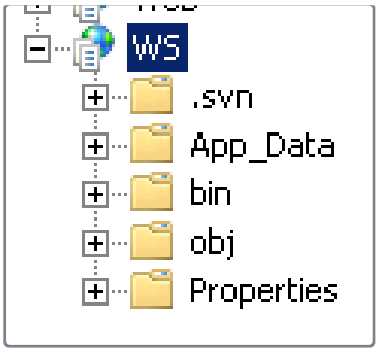
Choose WS as your alias and select the path to the WebServices directory.

(e.g.folder [Drive]:\InextorX\GateWS).

Select Application pool (ASP .NET v4.0 .or the Application pool created for VaultRemotingWindowsIdentity/ VaultRemotingWindowsGroup).

Click OK.

Check that the WS directory is structured as shown below.



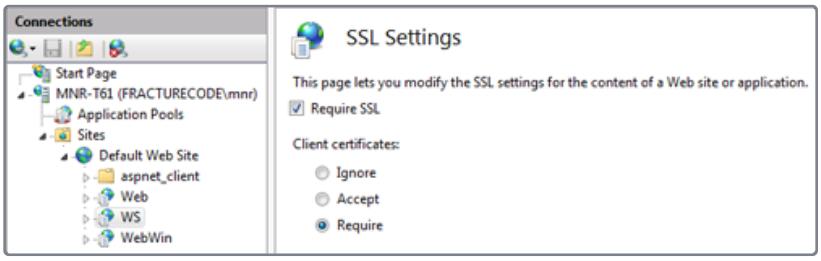
Directory Structure

Setting Secure Communication Parameters

Select the WS application and double-click on SSL settings. Set:

- Require SSL if using SSL
- Client Certificate:
  - Ignore: If not allowing Gate Certificate for authentication.
  - Accept: If Gate certificate for authentication is allowed but other mechanism as well
  - Require: If only using Gate certificate for authentication

For each WS application, it is recommended to uncheck Require SSL and to select Accept option.



SSL Settings

Authentication Settings

On the right side, under the IIS section, double-click on the Authentication item.



Authentication

If using Windows Integrated authentication, make sure Windows authentication is enabled.

If using Certificate authentication, make sure Anonymous Authentication is enabled (SSL must also be enabled in 2.2 above).

In all cases, make sure Impersonation is disabled and Basic Authentication is enabled.

For each WS application, it is recommended to enable

- **Anonymous Authentication**
- **Basic Authentication**
- **Windows Authentication**

and disable the other types of authentication.

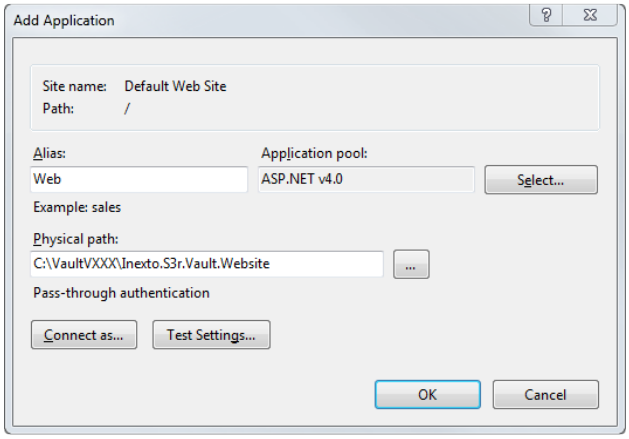
Ensuring that messages from Gate can reach the Vault

Large messages from the Gate can be rejected by IIS if they are larger than the default of 49 K (49'152 bytes). It is necessary to change the uploadReadAheadSize parameter to 100 MB, for the GateManagerWS Web application, as described in section 2 of certificate negotiation.

Web Site Application not using Windows Authentication (Web)

Adding Application

Right-click on Default Web site and choose **Add** Application.



Choose Web as your alias and select the path to the website directory.

(e.g.folder [Drive]:\InextorX\VaultWebsite)

Select Application pool (ASP .NET v4.0 or the Application pool created for VaultRemotingWindowsIdentity / VaultRemotingWindowsGroup)

Click OK.

Install Root Certificate in the Web Server Certificate Store

The Root certificate of the ADMIN account should be installed in the Web application server's certificate store in the "Trusted Root Certification Authorities" folder.

If this is not the case, IIS will not trust the ADMIN certificate and will return an HTTP error 403.16.

To install a certificate in the certificate store, please refer to CertificateInstall.

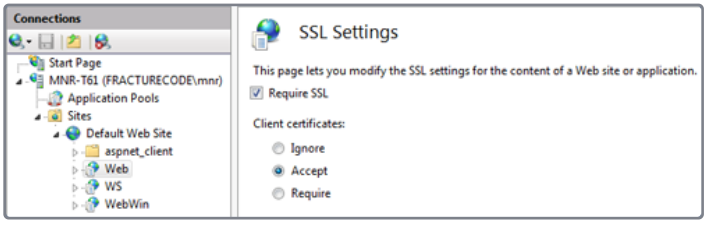
Setting Secure Communication Parameters

Select the application and double-click on SSL settings. Set:

- Require SSL if using SSL
- Client Certificate:

- Require client certificates (If account login only should be used)
- Accept client certificates (if the account and other mechanism are allowed)
- Ignore client certificates (if not allowing account login).

Note Account login: Log on to the website with a certificate (belonging to an account on the Vault).



SSL Settings

Authentication Settings

On the right side, under the IIS section, double-click on the Authentication item.



Authentication

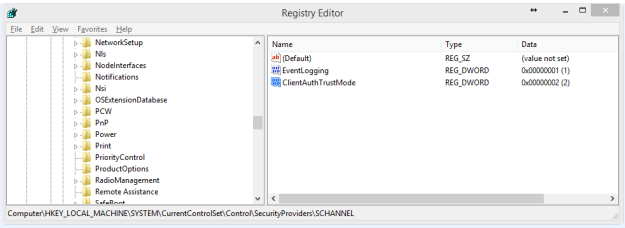
Make sure only Anonymous Authentication is enabled. Make sure impersonation is disabled.

Applies To: Windows 8.1, Windows Server 2012 R2, Windows Server 2012, Windows 8

There have been changes in functionality in the Schannel Security Support Provider (SSP), which includes the Transport Layer Security (TLS), the Secure Sockets Layer (SSL), and the Datagram Transport Layer Security (DTLS) authentication protocols, for Windows Server 2012 R2, Windows Server 2012, Windows 8.1, and Windows 8.
https://technet.microsoft.com/en-us/library/hh831771.aspx

Symptoms: After installing all certificates in the correct places and everything configured properly in IIS you kept getting 403.16 2148204809 error in the IIS log when you try to login using the client certificate.
To fix it:

1. Execute regedit.exe
2. Add a new DWORD value to HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL name: ClientAuthTrustMode Value type: REG_DWORD Value data: 2

3. 

Setting Parameters for Certificate Negociation

You should always set the parameters for Certificate negotiation, so suggestion 1 must be executed. If using multiple website servers, then this has to be done on both servers.
When sending large messages, for instance production changes during coherency, the IIS can report error 413 in the IIS log file, meaning that the content is too large. This could be the case when too many productions are sent at a time (5000+). In the configuration file for the Gate Service (see document Configure the Gate section "Gate Service (GateService.exe.config)") the parameter "UploadPRODsMaxChunkSize" is set. If this parameter is set to report max. 1000 and the message still appears the reason could be negotiation with the client certificate. What happens is that the server demands re-negotiation and in the process interrupts the large message. There are two ways to remedy this, and it is recommended to follow the steps in suggestion 1:

1. Set the parameter SSLAlwaysNegoClientCert to true. This makes the client negotiate the certificate prior to sending the large message.
   - Make sure that all IIS management tools are closed.
   - Execute the command at c:\inetpub\AdminScripts.
     cscript adsutil.vbs set w3svc/1/SSLAlwaysNegoClientCert true
   - To verify that the setting is active execute:
     cscript adsutil.vbs get w3svc/1/SSLAlwaysNegoClientCert

   If the folder AdminScripts does not exist, then install the "IIS Management Scripts and Tools" and "IIS 6 Scripting Tool". Go to "Programs and Features" and click "Turn Windows features on and off".
2. Set the parameter uploadReadAheadSize to a value (bytes) large enough for the largest message. One production order synchronization message could take up to 1 Megabyte, so a value of 100 MB would handle 100 such messages.
   - Go to folder: C:\Windows\System32\inetsrv
   - Execute the following command:

   ```
   appcmd.exe set config -section:system.webServer/serverRuntime /uploadReadAheadSize:"100000000"
   /commit:apphost
   ```

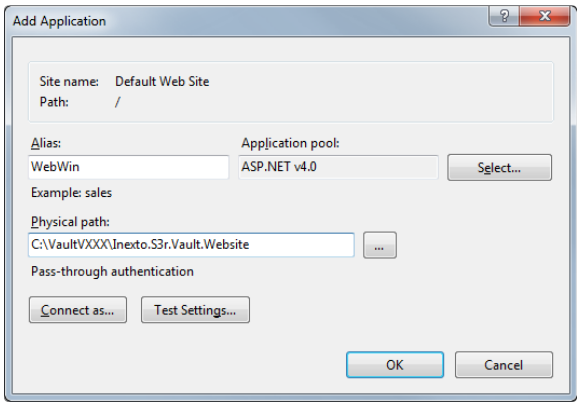
3. Set the parameter maxAllowedContentLength to the same value as uploadReadAheadSize. This parameter can be found under system.webServer/security/requestFiltering/requestLimits

Web Site Application for Windows Users (WebWin)

Note: This application should only be configured if Windows Authentication for users is used and is only recommended for Intranet solutions.

Adding Application

Right-click on Default Web site and choose **Add** Application.



Select ASP .NET v4.0 Alias: WebWin Physical path: the directory where the VaultWebsite folder is located (e.g.folder [Drive]:\InextorX\ VaultWebSite)

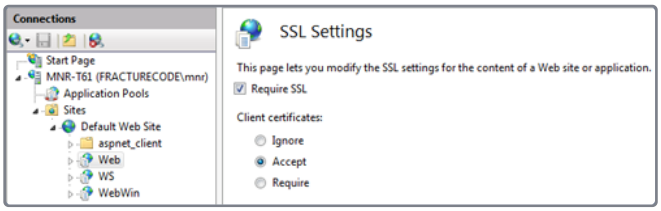
Click OK.

Setting Secure Communication Parameters

Select the application and double-click on SSL settings. Set:

- Require SSL
- Client Certificate:
  - Require client certificates (If account login only should be used)
  - Accept client certificates (if the account and user login should be used)
  - Ignore client certificates (if user login only should be used).

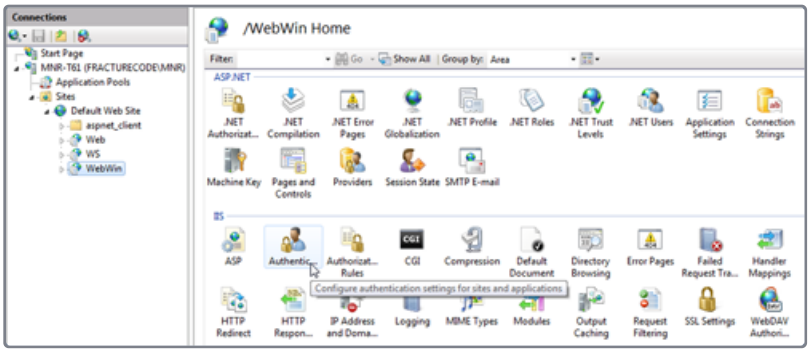
Note User login: Log on to the website with user name and password. Account login: Log on to the website with a certificate (belonging to an account on the Vault). Usually "Accept" is chosen.



SSL Settings

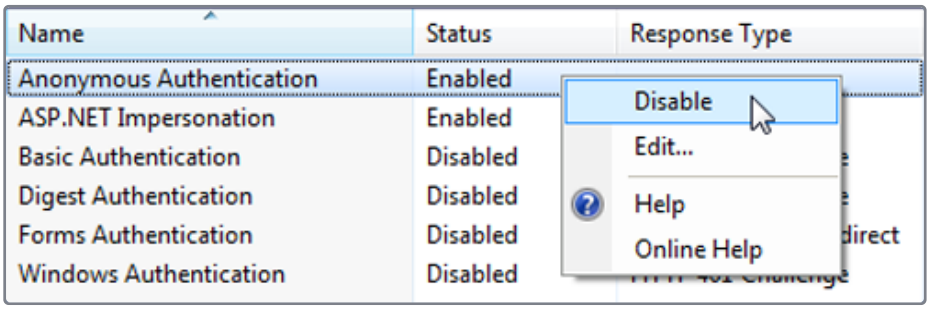
Authentication Settings

Click on WebWin and double-click on Authentication.



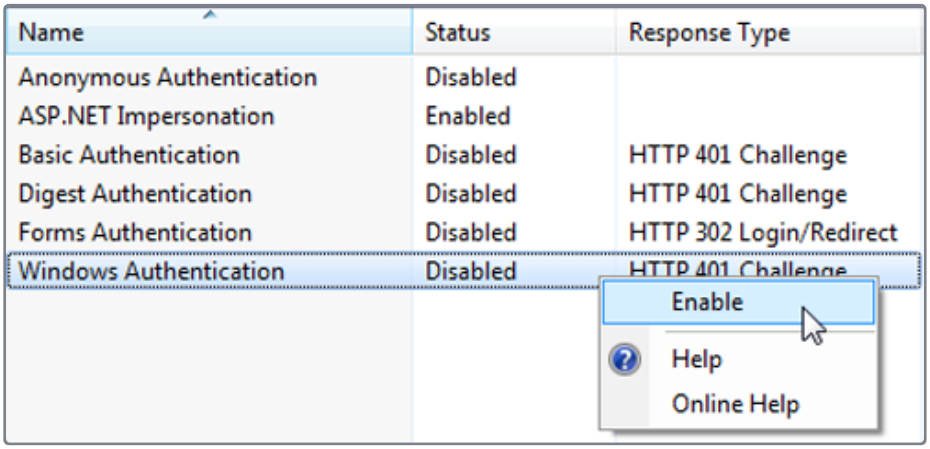
Authentication

Ensure Impersonation is enabled. Disable Anonymous Authentication.



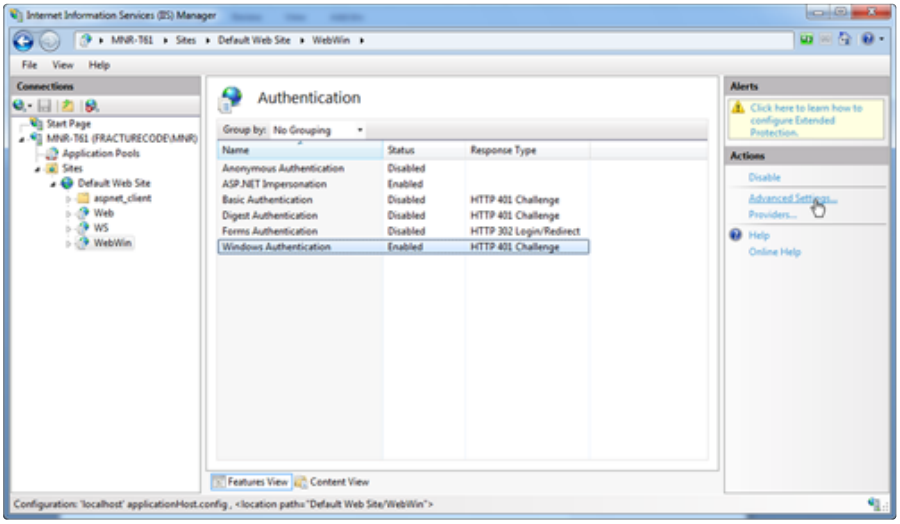
Anonymous Authentication

Enable Windows Authentication



Windows Authentication

For Extended Protection click "Advanced Settings..."



Advanced Settings

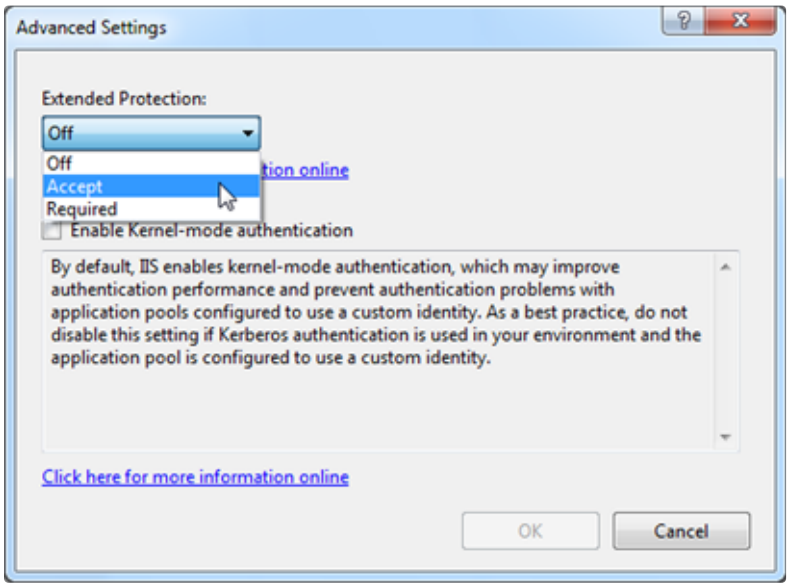
Here you can select the desired Protection Mode.

By default this is set to "Off". This value specifies that IIS will not perform channel-binding token checking. This setting emulates the behavior that existed before extended protection was available.

"Accept" specifies that channel-binding token checking is enabled, but not required. This setting allows secure communication with clients that support extended protection, but still supports clients that are not capable of using extended protection.

"Required" specifies that channel-binding token checking is required. This setting does not provide support for clients that do not support extended protection.

It is recommended to set this value to "Off" or "Accept", so that clients without Extended Protection can be supported.



Extended Protection

Exposing Web Applications to External Systems Using Certificate Mapping

In order to expose web applications to external systems or users from outside the domain on a Vault that uses Windows authentication, it is necessary to enable and configure OneToOneMapping in IIS which allows mapping a certificate to a windows account. The procedure is described here: Exposing Vault Web Services For External Systems.

Setting a Different Website for Web Services

In some cases, you may want to define a different website for the web services. This allows you for example to keep the web services private while the Vault website is public. In this case, a new website has to be defined in the IIS to which some or all of the web services can be added as applications. This also makes it possible to specify another port for the communication. Please refer to IIS documentation (Web Application and Enable SSL) to accomplish this.

Starting IIS

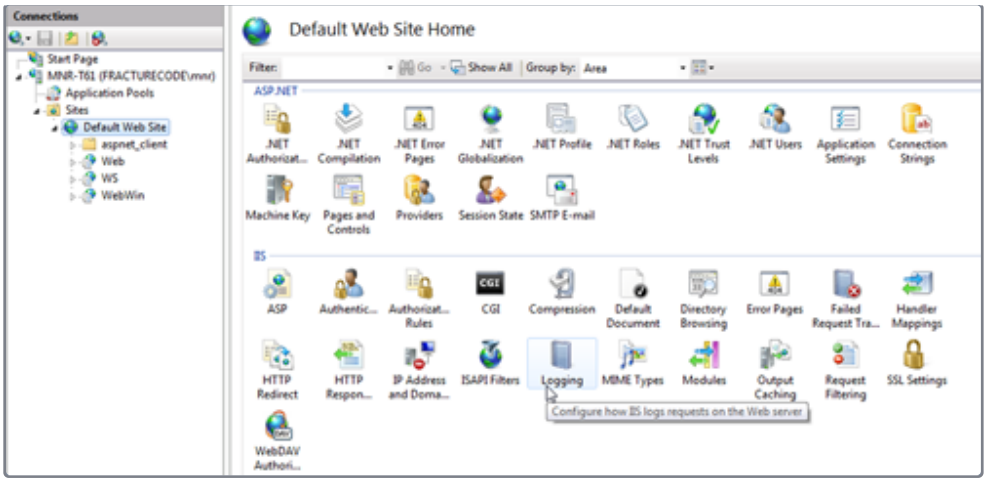
Start the IIS (See Web Application - Starting IIS)

If the IIS is not starting, check that the service "World Wide Web Publishing Service" is running.

Perform the following steps to check that the service is running:

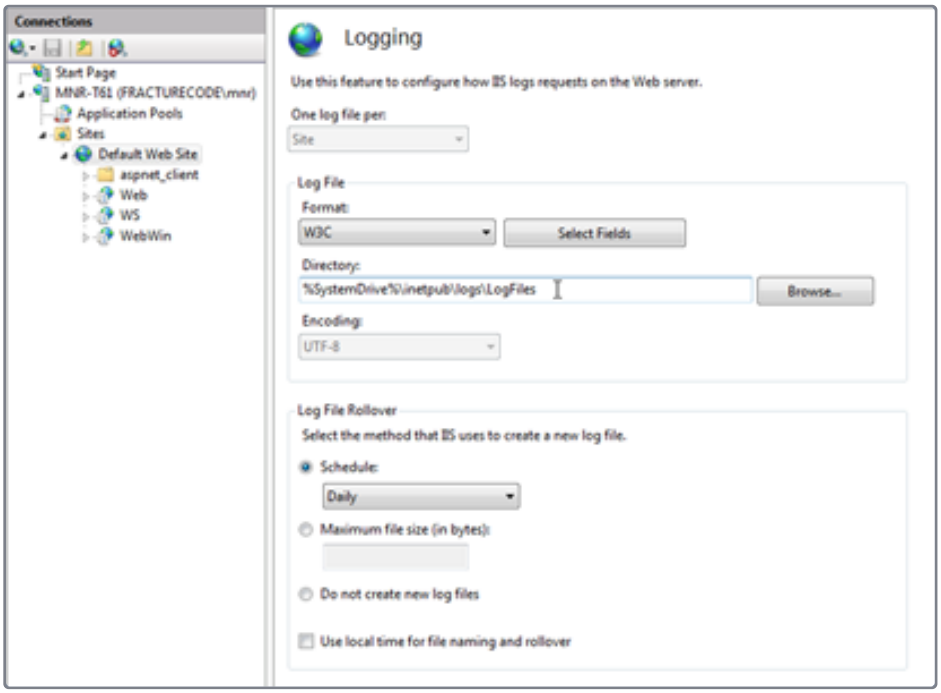
- Open the services.msc GUI by typing it into the Run option in the Start Menu or it can be located on Administrative Tools→ Services
- Locate the World Wide Web Publishing Service and on the column Status it should say Started, if not right click the service.
- If the option Start is enabled, it means that the service is stopped and has to be started.
  - To start the service, right click the Service and choose Start option.
- If the option Start is disabled, it means that the Service is running or there is another problem with it, in this case you should restart the service.

Check the IIS log file. The path to the IIS log file can be found by clicking on "Default Web Site". Click "Logging".



Logging

Find the path to the IIS log file (%SystemDrive%\inetpub\logs\LogFiles).



Logging Path

Windows Administrators: Liz

LIZ Installation and Setup

Setup the Liz

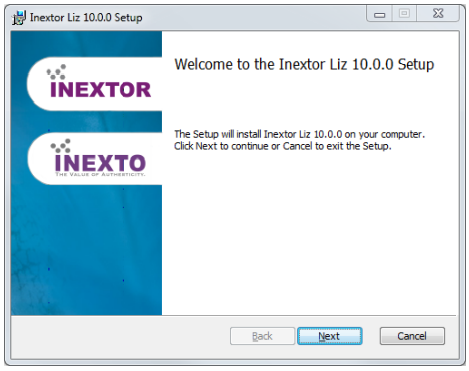
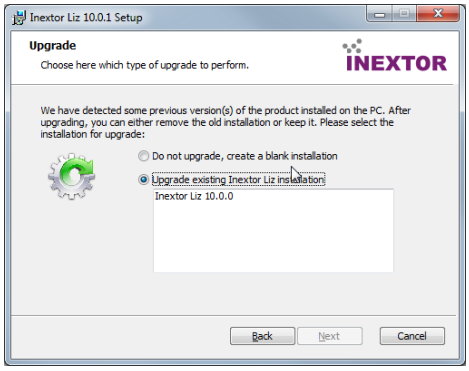
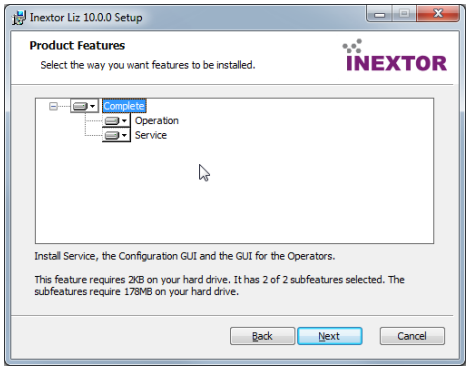
This page describes the procedure in order to install and configure a Code Generator. The information is organized as follows:

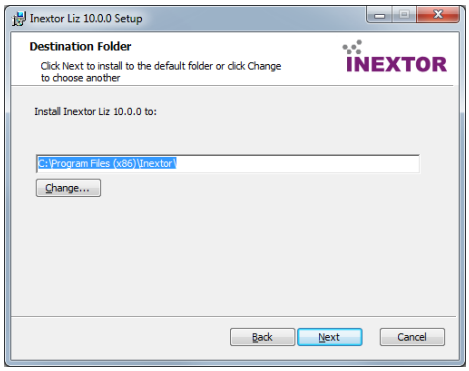
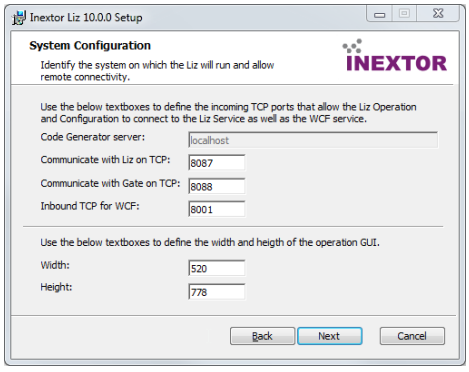
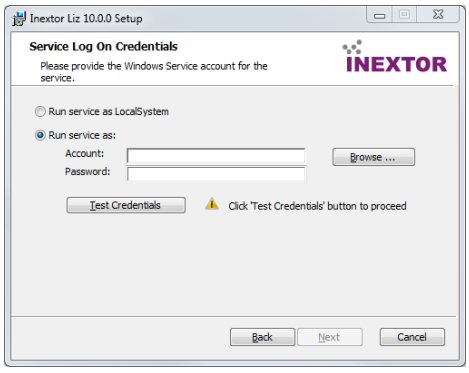
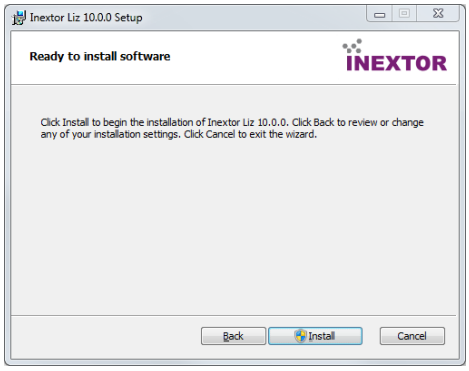
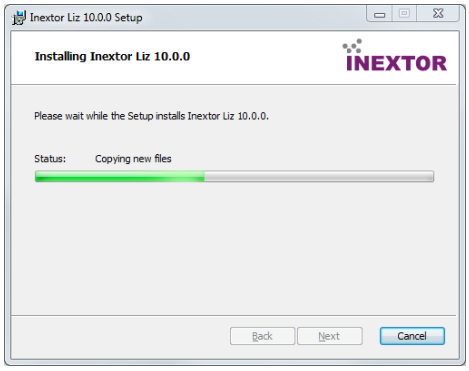
1. Check the Prerequisites
2. Install the Serializer
3. Assess the Serializer Installation
4. Configure the Serializer.
5. Install the Serializer in silent mode

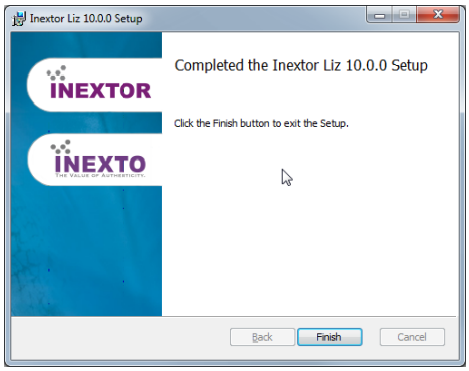
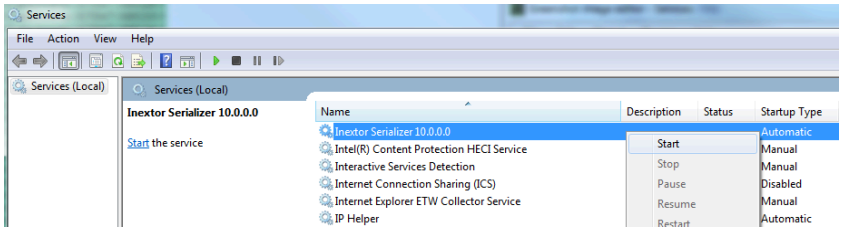
Check the Prerequisites

See the General Installation Procedures: Technical Overview, Manufacturers Checklist, IT Infrastructure: Hardware and Software Requirements, Liz Procedures: Prerequisites for Implementation of the LIZ and Liz network: Required firewall settings.

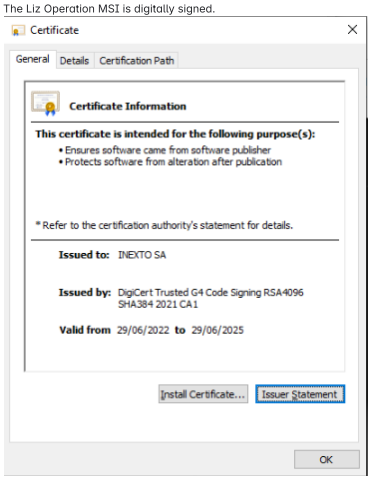
Install the Liz

1.	Run the setup file **InextorLiz-<version>.msi**. Click **Next**.	
2.	In case of an existing 4.5 installation the system will propose to do either a side by side installation or a complete upgrade. The preferred approach is to do a complete upgrade. ⓘ Prior to doing the upgrade, you will need to have first migrated the Gate to version 10.0.0.	
3.	The Liz contains two components, Operation and Service. Operation is shipped with the following languages: British English, French, Dutch, German, Polish, Russian, Spanish and Mexican Spanish. The default language is English (en-US). Click **Next**. ⓘ This dialog is not proposed in case of an upgrade from a 4.5 version.	

4.	Use this window to change the destination folder. Click **Next** when you are satisfied with the destination folder.	
5.	The Installation Wizard will propose some configuration settings with the default values. • Liz host name (only activated if the service is not installed for separate GUI installation). • Liz TCP Port Number • Gate communication port (only activated for a service installation) • Liz WCF Production change port (only activated for a service installation) • Liz Operation GUI Width (only activated when Operation GUI is installed) • Liz Operation GUI Height (only activated when Operation GUI is installed) ⓘ Note that using the GPO applied on the AD Groups will automatically open these ports for you. Click **Next**.	
6.	Use this window to specify which account the Liz Service should use to run. If you do not want to use a domain account, you can select the LocalSystem account. If you select a domain account, you must click on **Test Credentials** before continuing. Click **Next**	
7.	Use the **Back** button to return to any configuration windows. Click **Install** to start the installation. ⓘ Installation of the software package requires elevated privileges and you will be prompted to confirm you accept it.	
8.	Installation is now in progress...	

9.	Click Finish to complete the installation. Note that you have the possibility to display the installation log file.	
10.	From the Control Panel, go to Administrative tools and start the application Services. In the Services window, locate Inextor Serializer Service 5.X.X, right click on it and select Start in the context menu.	

The Liz Operation MSI is digitally signed.



Assess the Liz Installation

Verify that the **Inextor Liz Service 10.0.0** is running with the account you specified during the setup. If the service failed to start, make sure that the account is set properly and that this account can actually write to the folder Database. If the Liz Operation component has been selected during installation, the Liz Operation application should also appear in the programs installed on the computer in the start menu.

Configure the Serializer

The first time you launch the Liz User Interface, you must configure the Liz to run properly, by importing a license file, configuring the serialization, attaching to an Gate, assigning a link-up setup, and registering with a Vault and obtaining an MSK. For instructions on how to perform this one-time setup, refer to Configure the Liz on the First Run. For more configuration information, refer to the general Liz Configuration page.

Parameters	Description
SERVICE_LOGON_TYPE1	Service logon
SERVICE_USERNAME	Service username
SERVICE_PASSWORD	Service password of the user
TCPLIZ	TCP port Liz
TCPGATE	TCP port Gate
INSTALLDIR	Directory of the installation
INSTALL_TYPE	Type of installation: • Complete • Client • Service

Install the Liz Client and Service in silent mode

The msi can be installed via command, without needing to use the installer UI.

When upgrading the Liz, whether you are using the msi in UI mode or silent command-line mode, any change you made to the Liz configuration file (Service\AppDomain.config, LizService.exe.config and LizOperation.exe.config in the installation folder) will not be preserved. If you have made changes to this file after installation, you need to:

- copy it to a different location before the upgrade
- perform the upgrade
- check for differences between your old configuration file and the new configuration file that was installed during the upgrade
- Apply any changes to the configuration file in the installation folder as needed, or copy your old file over the new one if no setting will be lost.

To do a silent installation/upgrade, use one of the commands below. The commands include sample values, which need to be changed to reflect your environment, including the name of the msi file and the installation location.

When upgrading the Liz using the silent msi installation command, you need to specify all the settings you would enter in the UI (described above) in the command, unless you know you are using the default value.

After the installation is complete, a log file will be generated (LizInstallLog.txt) which allows troubleshooting in case the installation failed.

Install client and service to run as LocalSystem:

Minimum command line (if all settings have default values):

- `msiexec.exe /qn /i InextorLiz-[Tobacco]-10.1.0.1700.msi SERVICE_LOGON_TYPE1=ServiceLocalSystem /l*vx LizInstallLog.txt`

All settings specified (can omit any that have default value):

- `msiexec.exe /qn /i InextorLiz-[Tobacco]-10.1.0.1700.msi SERVICE_LOGON_TYPE1=ServiceLocalSystem TCPLIZ=8087 TCPGATE=8088 OP_WIDTH=520 OP_HEIGHT=778 INSTALLDIR="C:\Program Files (x86)\Inextor\" INSTALL_TYPE=Complete /l*vx LizInstallLog.txt`

Install client and service to run under a different account

Minimum command line (if all settings are as default):

- `msiexec.exe /qn /i InextorLiz-[Tobacco]-10.1.0.1700.msi SERVICE_LOGON_TYPE1=ServiceAccount SERVICE_USERNAME="domain\user" SERVICE_PASSWORD="#####" /l*vx LizInstallLog.txt`

All settings specified (can omit any that have default value):

- `msiexec.exe /qn /i InextorLiz-[Tobacco]-10.1.0.1700.msi SERVICE_LOGON_TYPE1=ServiceAccount SERVICE_USERNAME="domain\user" SERVICE_PASSWORD="#####" TCPLIZ=8087 TCPGATE=8088 OP_WIDTH=520 OP_HEIGHT=778 INSTALLDIR="C:\Program Files (x86)\Inext`

Configuration Files (Liz Configuration)

The following configuration files define the settings for the Liz service and operation.

Configuration file	Description	Location	Documentation
LizService.exe.config	Configuration file containing all settings for the Liz Service.	Inextor\Liz 10.0\Service\	Liz Service - Configuration
LizOperation.exe.config	Configuration file containing all settings for the Liz Operation UI.	Inextor\Liz 10.0\Operation\	LizOperation.exe.config

After changing any setting in the application configuration file, the service must be restarted otherwise the changes will not be effective.
These settings have to be configured correctly for the service to start and work properly.

Before version 3486
Before version 3486, the following configuration files defined the settings for the Liz service and operation.

Configuration file	Description	Location	Documentation
AppDomain.config	Configuration file containing all settings for the Liz Service.	Inextor\Liz 10.0\Service\LizService	Liz Service - Configuration
LizServer.exe.config	This file was deprecated, please use Liz Service - Configuration instead.	Inextor\Liz 10.0\Service\	
LizOperation.exe.config	Configuration file containing all settings for the Liz Operation UI.	Inextor\Liz 10.0\Operation\	LizOperation.exe.config

Liz Service - Configuration

The table below shows the keys that can be modified in the LizService.exe.config file.

Key	Default value	Description
DBConnectionString		Connection string to the Liz database

⚠ Updating the value of the keys below should now be done in the Application Settings tab of the Liz Operation program. It is still possible to update the configuration file of the Liz service, but such a direct change will be lost during upgrades.

Key	Default value	Description
AllowCreationOfProductionOrders	true	Specifies if users can create a production order from the Liz UI
CertificateDbDataProvider	provider=System.Data.SqlServerCe.4.0	Data Provider Name is used to connect Certificate Database. The possible values are: • *provider=System.Data.SqlClient*: certificates are stored in SQL database • *provider=System.Data.SqlServerCe.4.0*: certificates are stored in SQL server compact file, the version 4.0 here can be changed depend on the version of SQL server compact used
LizAdmHost	localhost	Host name of LizAdministration service
CodeCommissioningByFileEnable	false	Only used if the INEXPRESS operation driver is enabled. If true, additional functionality is available from the Inexpress operation driver, for commissioning codes on Inexpress directly from the Liz.
CryptoDatabaseSafeType	None	Specifies the mechanism used to protect data store in database. The possible values are: • None: data is not encrypted • DataProtect: the symmetric key is encrypted using Data Protection, and the symmetric key is used to encrypt the DB • SingleRsa: SingleRsa: SingleRsa: SingleRsa: encrypt symmetric key using Rsa algorithm with one 'specific' key, and then use symmetric key to encrypt DB • DoubleRsa: encrypt symmetric key using Rsa algorithm with one 'specific' key and again for one 'global' key, then use symmetric

		key to encrypt DB • RsaDataProtect: encrypt symmetric key using Data Protection and again for one 'global' key (using RSA algorithim),then use symmetric key to encrypt DB.
DataSetValidation	./ExtendedFields/LizDataSetValidation.xml	Path to the file containing the validation rules for all input data types.
DataValidationRaiseException	false	Specifies if error in validation cause exception or only logging.
DefaultLoadAllProducts	true	By default load all product from Gate or filter them.
DeleteLogsBefore	10d	Logs that are older than the specified duration are entirely deleted from database.
EnabledDllFileDetection	true	Specifies if dll file change detection is enabled.
EStampsWarningLevel	5000	Operation GUI raises a warning when the remaining number of Volume Control units falls below that level.
FileInterfaceMonitoringInterval	5s	Sets how often to invoke function that processes files in input directory.
FileInterfaceNotificationInterval	1M	Sets how often to invoke function that saves Liz status into file.
HideAssemblyResolutionFailure	(XmlSerializers)|(resources)|(Cvs.ShiftCode)	Skips these exceptions, without trying to localize them. Resources are missing for these system DLLs and that would cause infinite recursion in the EL4Net exception hook. (not available in application settings screen)
IsLizSecureChannel	false	Specify if the Liz exposes a secure channel for Gate or not.
ListenPort	8087	Listening port of Liz service.
LizHost	localhost	Host name of Liz service.
MaxGateDisconnectionPeriod		[duration] If a duration is specified, and the Liz has not established connection wih the Gate for specified duration, an error will show in the Dashboard Global Logs
MSKPath	..\MSKs	Folder where decryption keys for MSK are stored.
PrinterConnectionRetryDelay	20s	If printer connection fails, sets the interval after which the Liz will try reconnecting.
PrinterCounterInterval	500ms	If buffer in code pusher not required, f buffer in code pusher not required, the Liz will refresh printer counters regularly at this interval.
ProgramName	LizServer	Name of the Program. (not available in application settings screen)
ProgramUnitName	Serializerserver	Name of the Program unit built by this deployment. (not available in application settings screen)
ProgramVersion	10.0	Version of the Program for this deployment. (not available in application settings screen)
ProtectedDBConnectionString	false	Specifies if the connection string is protected or not.

ReservedEStamps	10M	Number of Volume Control units that are reserved in advance. Applies only to pessimistic mode. Volume Control units are reserved when production updates (transactional or non-transactional) are issued. Value can be expressed in minutes ("M"), in codes ("C") or in transactional updates ("U"). • Expressed in "M": reserve the max number of Inextor codes that can be printed during M minutes. If M is smaller than the period for production updates (transaction and non-transactional), it is overridden by this period. • Expressed in "C": reserve a fixed quantity. If C is smaller than the max number of Inextor codes that can be printed between 2 production udpates, it is overridden by this period. • Expressed in "U": reserve the max number of Inextor codes that can be printed between 2 transactional production updates.
ShowUnassignedOrders	true	Specify if unassigned POs are shown when selecting a PO to prepare.
SolutionCompany	Inexto	Name of the Company providing this Solution. (not available in application settings screen)
SolutionName	Inextor	Name of the Solution. (not available in application settings screen)
UploadInterval	5M	Determines how often the Liz will send a non-transactional production updates to the Gate. If the value is higher than the ProductionUpdatePeriod (MSK policy), only transactional production updates will be sent.
RecoveryLimitPeriod	30d	This setting is used in the context of automatic recovery of productions in the case of abrupt interruption of production (e.g. because of a service restart or a crash). The last production is automatically recovered if the time between when the interruption and the time when the Liz Service has been restarted is lower than RecoveryLimitPeriod. The value 0 is used to disable the automatic recovery feature.

Liz Operation - Configuration

LizOperation.exe.config

The table below shows the keys that can be modified in the LizOperation.exe.config file.

Key	Default value	Description
LizAuthenticationMethod	None	Authentication Method used to communicate with the Liz (only if the communication is secure)
HideAssemblyResolutionFailure	(XmlSerializers)|(resources)	Skips these exceptions, without trying to localize them. Resources are missing for these system dlls and that would cause infinite recursion in the EL4Net exception hook.
ProgramName	LizOperationExe	Name of the Program.
ProgramUnitName	LizUserInterface	Name of the Program unit built by this deployment.
ProgramVersion	10.0	Version of the Program for this deployment.
SolutionCompany	Inexto	Name of the Company providing this Solution.
SolutionName	Inextor	Name of the Solution.
TopologyView	logical isometric	Defines the the way the dashboard is displayed. Possible values are: "isometric", "logical", "isometric logical", "logical isometric"
EnableStartFromBarcode	false	Allows to start a production by scanning a barcode.

Unattended and Mass Liz Deployment

To ease the Liz deployment we have prepared a PowerShell script that allows to deploy Liz to all the Liz PCs from the Gate Server. Please take your time to read the instructions and don´t hesitate to contact support if you have any question.

Overview

Purpose

The script push and installs msi files on selected PCs from a centralized server (In this case the Gate Server). **Works for updating already existing Lizs, not for new installations**.

Requirements:

- **PowerShell 5** or greater. Powershell Core (7) is also valid.
- WinRM service configured and running in the destination PCs.
- **Traffic on port 5985 allowed** between the Gate and the Liz. This is the port for WinRM.
- Destination PCs should be considered TrustedHost by the Gate Server.
- Decent code editor (Notepad ++, Visual Studio Code, etc) with Syntax check for JSON files.
- Account with administrative rights on destination PCs.
- **Windows Vista or lower is NOT supported**. We can try to execute the script but there is no guarantee that it will work. Please upgrade!
- **Windows 7 requires Service Pack 1 installed.** Please upgrade!

First Run

The complexity of the script lays on it´s first run, as it requires some data gathering from the support agent.

JSON config file

The first and mandatory step is to fill the JSON file.

config.json

```
{
    "logFullPath":"C:\\InextoSupport\\Inextor\\Deploy\\deployliz.log",
    "isDomainNetwork":false,
    "domainShortName":"",
    "useGlobalUser":false,
    "globalUserName":"",
    "globalPasswordHash":"",
    "enablePowerShell7Remote":false,
    "backupProgramDataLizFolder":false,
    "backupNoCompression": true,
    "deleteOldBackups": false,
    "AppDomainConfigEdit":false,
    "LizOperationConfigCopy":false,
    "GrantLogonAsServiceRights": true,
    "msiInstallationType":"LocalSystem",
    "msiOptionalParameters":{
        "LizTcpPort":"8087",
        "GateTcpPort":"8088",
        "OpWidth":"520",
        "OpHeight":"778",
        "InstallDir":"C:\\Program Files (x86)\\Inextor\\",
        "Domain":".",
        "User":"CDFUser",
"PasswordHash":"01000000d08c9ddf0115d1118c7a00c04fc297eb010000005ac065d88b2df24f86c2184101f04d1a000000000200
    },
    "AppDomainConfigOptionalParameters":[
        {"name":"ProductionPurgeInterval", "value":"30d"}
    ],
    "workstations":[
        { "computerName":"10.193.42.101", "userName":"Administrator", "passwordHash":"01000000d08c9ddf0115d11
    ]
```

```
}
```

Let´s review one by one the parameters.

- **logFullPath:** The path where the log will be saved. Please mind that in JSON files you need to use double \ (\\) as escape character.
- **isDomainNetwork:** If set to **true** (case-sensitive), the credentials will be constructed with the **domainShortName** string. This is useful for clients inside a domain network. In a domain network all the users have the structure *domainname\username*, while outside domain are *computername\username*.
- **domainShortName:** Short name for the domain. Only works with **isDomainNetwork** set to **true**. Eg: Inexto
- **useGlobalUser:** If set to **true** the same credentials (defined on **globalUserName** and **globalPasswordHash**) will be used to connect to all the workstations, ignoring the per-client configuration.
- **globalUserName:** Name of the global user that will be used to connect to all workstations. Only works if **useGlobalUser** is set to **true**.
- **globalPasswordHash:** Password hash of user that will be used to connect to all workstations. Only works if **useGlobalUser** is set to **true**.**Note:** Please refer to the section "Generate Password Hash"
- **msiInstallationType:** Sets the type of installation for the Liz msi (defined here) It must have one of the four following values:
  - *LocalSystem*: This is the one that will be used most of the time. Use the default values and Local System as running user.
  - *LocalSystemParam*: This one will use the Local System as running user, but it will use the values inside **msiOptionalParameters (Domain**, **User** and **PasswordHash** will be ignored) for the msi**.**
  - *OtherUser*: Use default values for msi but allows to set a custom user to run Liz Service. **Domain**, **User** and **PasswordHash** inside **msiOptionalParameters** will be used for this purpose and **must be filled** (the rest of parameters will be ignored).
  - *OtherUserParam*: Use custom values for the msi and custom user to run Liz Service. You **must** fill all the values inside **msiOptionalParameters**.

> Avoid *OtherUser* and *OtherUserParam* installs on no domain networks as it will complicate things and you will need a json configuration file for each workstation.

- **enablePowerShell7Remote:** Enables (**true**) the script to create remote PowerShell sessions to PowerShell 7 endpoints. This is useful if the machine where the Liz is installed has an older version of Windows and the PowerShell version can´t run the entire script. False by default.
- **backupProgramDataLizFolder:** If enabled (**true**) the script will make a backup of the folder "*C:\ProgramData\Inextor\Liz 10.1.0*" . This folder contains, among other things, the Liz DB. Please mind that each backup could take up several GB.
- **backupNoCompression:** If enabled (**true**) the backup will be a raw copy of the folder. This is useful in machines with low RAM (<4GB) as the Compress-Archive cmdlet can fail if there is no enough memory. By default (**false**) the backup is compressed to a zip file. Requires **backupProgramDataLizFolder** set to **true**.
- **deleteOldBackups:** If enabled (**true**) the script will remove **all** previous zip files or folders with the backups.
- **AppDomainConfigEdit:** If set to **true** the keys listed on **AppDomainConfigOptionalParameters** will be added or modified on AppDomain.config
- **LizOperationConfigCopy:** If set to **true** LizService.exe.config will be preserved for the new version. Mandatory on ITG.
- **GrantLogonAsServiceRights: JTI**-specific. If set to **true**, after the msi installation (as **LocalSystem**), the script will manually change the user that will run the Liz service. This was done as msiexec was not capable to install with another user. It will use the user and password defined on **msiOptionalParameters**.

> **Warning**
> Be careful about about enabling **AppDomainConfigEdit.** Please check with Inexto first if there is any value that you should modify.

- **msiOptionalParameters:**
  - *LizTcpPort*: Liz service listening port. Default is 8087,
  - *GateTcpPort*: Port for communications with Gate. Default is 8088.
  - *OpWidth and OpHeight*: Relative to the window size. Most likely you won´t need to change this. Default values are 520 and 778.
  - *InstallDir*: The path where Liz will be installed. Be careful as you need to escape "\" character with another "\" character. Default value is C:\\Program Files (x86)\\Inextor\\
  - *Domain*: The domain of the user that will run the Liz Service. In case of a no domain network replace it with the computer name.
  - *User*: The name of the user that will run the Liz Service.
  - *PasswordHash*: The password hash of the user that will run the Liz Service.**Note:** Please refer to the section "Generate Password Hash"
- **AppDomainConfigOptionalParameters:** A collection of parameters to change on AppDomain.config. Needs **AppDomainConfigEdit** set to **true.**
  - *name*: Name of the key/property to change (Eg. ProductionPurgeInterval)
  - *value*: Value of the key/property to change (Eg. 30d)
- **workstations:** A collection of all the workstations where the Liz will be installed. If you need to add another one copy the line from the example and change the values.
  - *computerName*: The computer name or IP of the machine. In a network with no domain it is **highly recommended** to use the IP as if NetBios is not present it won´t be able to resolve the computer name.
  - *userName*: The name of a user that has Administrative rights over the destination machine. This will be ignored (and therefore can be set to "") if **useGlobalUser** is set to **true**.
  - *passwordHash*: The password hash of the user defined on **userName**. This will be ignored (and therefore can be set to "") if **useGlobalUser** is set to **true**.**Note:** Please refer to the section "Generate Password Hash"

Prepare workstations.

In the workstation

First thing: Make sure that the execution policy allows running scripts (this command must be run under the user that will be used to install the msi)

```
Set-ExecutionPolicy -ExecutionPolicy Unrestricted -Scope CurrentUser
```

The script ***Enable-WinRM.ps1*** handles WinRM configuration, including setting to *Private* the *Public* network connection profiles.

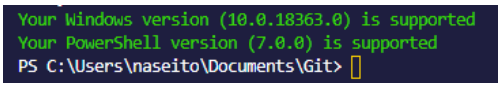
```
.\Enable-WinRM.ps1
```

You will be requested to input computer name (or IP), user and password. The script will take care of the conversion of the password to hash and saving the values to the json.

You can add as many workstations as you want (the script will keep running until you press 'n')

This script can also check the Windows and PowerShell versions requirements for you. Just add the **-CheckRequirements** switch on the execution:

```
.\Enable-WinRM.ps1 -CheckRequirements
```



> **Warning**
> Please note that if you execute the script -CheckRequirements and the PC doesn´t comply with them, the execution of the script will end.

In the Gate Server:

Now the script ***Add-ComputerConfiguration.ps1*** handles everything to include a new workstation in the json file.

```
.\Add-ComputerConfiguration.ps1
```

You will be requested to input computer name (or IP), user and password. The script will take care of the conversion of the password to hash and saving the values to the json.

You can add as many workstations as you want (the script will keep running until you press 'n')

The script have two optional parameters:

- *-CheckConnection*: If this parameter is set, a TCP connection test will be done to port 5985 (WinRM). If it fails the WinRM service is not reachable and the computer won´t be added to the list.
- *-AddTrustedHost*: If this parameter is set, the destination computer will be added to the local trusted host list.

I highly recommend enabling both parameters in **no domain networks**:

```
.\Add-ComputerConfiguration.ps1 -CheckConnection -AddTrustedHost
```

On **domain networks** enable only -CheckConnection

```
.\Add-ComputerConfiguration.ps1 -CheckConnection
```

Run the script:

> You **must** first rename the Liz msi file to something else and remove the [ ] characters as the script has issues reading the square brackets.

By this time you should have all the configuration and workstations filled in the config.json file.

The script accepts 3 parameters, and all are **mandatory**:

- -MsiPath: Path to the msi file.
- -DataPath: Path to the json configuration file.
- -Verbose: Enables the verbose of the script. With this you will be able to see the current step in the script at every moment.

```
.\Deploy-Liz.ps1 -MsiPath "C:\InextoSupport\Inextor\Liz-2140.msi" -DataPath "C:\InextoSupport\Inextor\Deploy Liz\co
```

The script logs to a deployliz.log file in the same folder than the script.

deployliz.log example

```
[04/05/2020 - 03:53:13:8138] [INFO] Creating a remote PS Session on GDCOMP
[04/05/2020 - 03:53:14:3496] [INFO] Copying InextorLiz-10.1.0.2105.msi to GDCOMP
[04/05/2020 - 03:53:24:2186] [WARNING] Liz service is running on GDCOMP
[04/05/2020 - 03:53:24:2216] [INFO] Stopping Liz Service and Liz Operation on GDCOMP
[04/05/2020 - 03:53:24:8155] [INFO] Installing MSI remotely on GDCOMP
[04/05/2020 - 03:54:06:5962] [INFO] Trying to start Liz Service on GDCOMP
[04/05/2020 - 03:54:56:5663] [INFO] Liz service was successfully started on GDCOMP
[04/05/2020 - 03:54:56:5818] [INFO] Remote PS Session closed on GDCOMP
```

Second and consecutive runs:

As all the workstation have been previously configured and the json file is correctly filled, you only need to run the script again.

```
.\Deploy-Liz.ps1 -MsiPath "C:\InextoSupport\Inextor\Liz-2140.msi" -DataPath "C:\InextoSupport\Inextor\Deploy Liz\co
```

Additional Notes

1. Powershell 5 is included with Windows 10 (first realease) or greater and Windows Server 2016 or greater. In case you have older OS you need to install a newer PS version.

a. *Powershell 5*:

> This is the recommended option (over PowerShell 7), but as PowerShell 5 will replace the current version installed, we supported PS7 in case you don´t want to do it.

> **Warning**
> This is the list of compatible OS: Windows 7 Service Pack 1, Windows 8.1, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2

Download and install the following package: https://www.microsoft.com/en-us/download/details.aspx?id=54616

b. *Powershell 7*:

- 32-bit package here.
- 64-bit package here.

2. If you decide to use PowerShell 7 on the Liz PCs instead of using the *winrm quickconfig* option mentioned in previous steps, from a PowerShell 7 terminal (both icon and background are black) run the following:

```
Enable-PSRemoting -Force
```

As this also uses WinRM as connection method it is mandatory that all the network connection profiles are set to private or domain.

Files

Version	config.json	Enable-WinRM	Add-ComputerConfiguration	Deploy-Liz	Full Package	Changes
1.3				Deploy-Liz-1.3.ps1	Deploy-Liz-1.3.zip	• Now -DataPath is no longer mandatory. If you don´t fill it the script will search for config.json in the same folder than the script. • Now -MsiPath is no longer mandatory. If you don´t fill it the script will open a FileDialog window to select and rename the msi.
1.2.0.1				Deploy-Liz-1.2.0.1.ps1	Deploy-Liz-1.2.0.1.zip	• Fixed a ConvertFrom-SecureString command that was missing a -Force switch.
1.2	config1.2.json			Deploy-Liz1.2.ps1	Deploy-Liz1.2.zip	• Better handling of Liz Service and Liz Operations service. (No more errors should appear if the service is not present or the process is stopped) • Testing: Logs the installation result (MSI code) • [**JTI**] Now the script can manually change the user running the Liz Service. This is ideal for cases in which the MSI installation fails with [Unable to grant log on as a service permissions] • Testing: Logs the result of changing the service running account.
1.1.1				Deploy-Liz-1.1.1.ps1		• Testing: Logs the installation result (MSI code) • Please rename the script to Deploy-Liz.ps1
1.1				Deploy-Liz.ps1		• Fixed an error with path handling
1.0	config.json	Enable-WinRM.ps1	Add-ComputerConfiguration.ps1			• Initial Stable Release

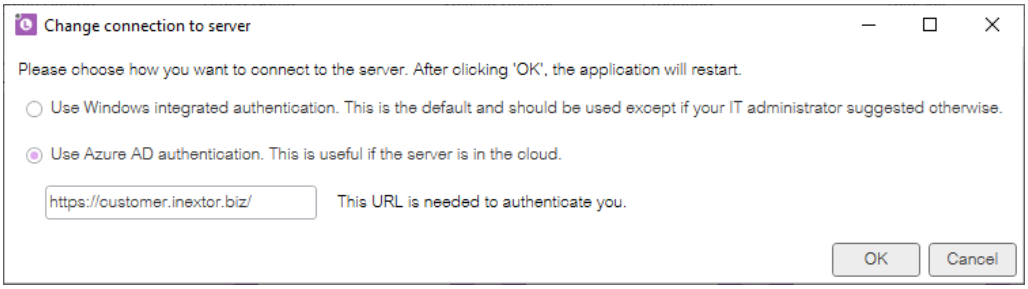
User Authentication on Liz Server

By default, users launching the Liz administration program connect to the Gate server (indirectly through the Liz server) using the Windows integrated authentication mechanism (i.e. their Windows credentials are transparently re-used).

The list of Active Directory groups they are a member of is used to manage their permissions on the Liz.

If the Gate server is running on a machine that is not part of a Windows domain, this authorization mechanism did not work. Starting from version 10.1.0.3350 of the Gate, this scenario is now supported.

The Liz administration program has a button called 'Change connection' in its information pane. Clicking it presents us with a configuration window:



Activating the first radio button is the default configuration, suitable for use in a Windows domain.

Activating the second radio button enables the use of Azure AD to authenticate the user (and find the groups s/he is a member of). It is necessary to specify the URL of the Vault Web site, in order to perform this authentication. Typically, single sign-on will be enabled and the user is not prompted for credentials.

When configuring groups for Liz security (see page Gate Security UI: Liz Security), the group name needs to be entered without any domain name.

When this authentication mechanism is used, the Liz server is usually unreachable from the Gate server (for security reasons, the network in the production environment is not accessible from the Internet).

In this case, the following restrictions apply on the Gate operations:

- The Gate server is not able to monitor the status of the Liz
- The Gate administration program can no longer be used to manage the Liz

General installation guide: IT infrastructure

- IT Infrastructure: On Premise vs. Cloud
- IT Infrastructure: Hardware and Software Requirements
- IT Infrastructure: Vault for IT

IT Infrastructure: On Premise vs. Cloud

On Premise vs. Cloud

On Premises: In an on-premises environment, the servers are deployed in-house within thr manufacturer's IT network. An enterprise is responsible for maintaining the solution and all its related processes, which can be either the manufacturer himself or a third-party IT company.

Cloud: While there are different forms of cloud computing (such as public cloud, private cloud, and a hybrid cloud), in a public cloud computing environment, resources are hosted on the premises of the service provider but enterprises are able to access those resources and use as much as they want at any given time.

Key role of the Gate Server

The Gate Server is a critical software component because it performs the following tasks:

- stores the master data required to initialize (start) a new production.
- cache external codes which have been downloaded from various id issuers sources.
- record production start/stop events.

If the factory loses connectivity to the Cloud (accident occuring in a Data Center, or internet network outage), the manufacturing lines will be able to continue printing codes for the Production Orders already in progress, using the codes already downloaded.

However, the following operations become impossible:

- manufacturing lines cannot download more codes from the Gate Server.
- production changes (new PO start) cannot be executed.

On-premises systems remain the most secure choice

We strongly recommend to favor the hosting of the Gate on-premises to ensure continuity of manucturing operations.

IT Infrastructure: Hardware and Software Requirements

In this chapter, you will find the **minimum** hardware and software requirements for the installation of each INEXTOR component.

> Generally speaking, it is recommended to install SQL server on a separate server, and purchase SQL licensing per core model.
>
> If the INEXTOR Services (Vault, Gate, Liz) and SQL Server are installed on the same server, the requirements must be added.
>
> If the servers are also used as workstations with end-users using RDP and running the Administration GUI applications, then some additional RAM should be added.

> (*1) SQL Server 2016 SP1 or later is recommended for external code verification because it supports DECOMPRESS and avoids the need to enable CLR execution. See also /wiki/spaces/S3RRG/pages/402340631

For all components hosted by the customer (e.g., Gate server, Gate database, Vault Server, Vault Database, Liz), the following best practices are strongly recommended:

- Keeping Windows operating system up-to-date, especially security patches.
- Use Windows authentication to connect to the database.
- Use network filtering rules to restrict access to the database (and to the server components, if possible).
- And of course, standard IT best practices.

For more details, refer to prerequisites page for each component:

- Vault Prerequisites and Vault Required Firewall Settings.
- Gate Prerequisites and Gate network: Required firewall settings.
- Liz Prerequisites and Liz network: Required firewall settings.

Hardware and Software Requirements: Vault Requirements

- Architecture is x64.
- Virtualization is supported on HyperV, VMWare, KVM or SQL Azure.
- Windows Operation System should be Server 2016 or higher.
- SQL Server should be **2016 SP1**, or higher (2017, 2019).
- SQL Server collation should be **SQL_Latin1_General_CP1_CI_AS**
- .NET Framework 4.8 is required.
- The Vault can be installed in a virtual environment.
- The database can be installed on remote SQL Servers.
- The Vault is a 3-tier application that can be installed on 1 to 7 servers, depending on the data center environment and redundancy strategy.

	Vault Service	**Vault database**	
Number of manufacturing lines *(please interpolate the requirements based on your number of lines)*	***Azure Virtual Machines*** *(Reference: https://learn.microsoft.com/en-us/azure/virtual-machines/sizes)*	***Azure SQL***	***Physical SQL Server*** *(CPU : 3rd Generation Intel® Xeon®)*
20 Lizs	Standard D2s v3(2 vcpus, 8 GiB memory) Storage = 100GB (for log files)	Standard S3: 100 DTUs	CPU: 4 cores RAM: 16Go Basic Storage (*) : 16 Gb
100 Lizs	Standard D4s v3 (4 vcpus, 16GiB memory) Storage = 100GB (for log files)	General Purpose: Standard-series (Gen5), 4 vCores	CPU: 4 cores RAM: 32Go Basic Storage (*) : 16 Gb
400 Lizs	Standard D8s v3 (8 vcpus, 32 GiB memory) Storage = 100GB (for log files)	General Purpose: Standard-series (Gen5), 12 vCores	CPU: 12 cores RAM: 64Go Basic Storage (*) : 16 Gb

(*) The Vault Database storage mostly dependents on the volume of external codes, and the data retention time :

	Raw Storage per billion codes	**Typical Retention time**	**Typical code volume in billion per year**	**Example: Storage needed for a volume of 10 billion codes per year, during the typical retention time**
External Code Storage (TPD, DLR, SAT)	10 Gb	5 years	10	10 x 10 x 5 = 500 Gb
SQL Code materialization (TPD, DLR, SAT)	60 Gb	5 years	10	60 x 10 x 5 = 3000 Gb
SQL Code materialization (SUPI12 Jordan)	150 Gb	5 years	1	150 x 1 x 5 = 750 Gb
Tpd Primary Code Check History	60 Gb	6 months	10	60 x 10 x 1/2 = 300 Gb
Code Check History (codes verified on Vault)	300 Gb	5 years	0.001	300 x 0.001 x 5 = 1.5 Gb

Hardware and Software Requirements: Gate Requirements

- Architecture is x64.
- Virtualization is supported on HyperV, VMWare, KVM.
- Windows Operation System should be Server 2016 or higher.
- SQL Server should be **2016 SP1**, or higher (2017, 2019).
- SQL Server collation should be **SQL_Latin1_General_CP1_CI_AS**
- .NET Framework 4.8 is required.
- The Gate can be installed in a virtual environment.
- The database can be installed on remote SQL Servers.

Number of manufacturing lines	**Gate Service** **Azure Virtual Machines**	**Gate database** **Azure SQL**	**Gate database** **Physical SQL Server**
10 Lizs	CPU: 2 cores RAM: 8 Gb RAM Storage = 100GB (for log files)	General Purpose: Standard-series (Gen5), 2 vCores	CPU: 4 cores RAM: 16 Go
20 Lizs	CPU: 2 cores RAM: 8 Gb RAM Storage = 100GB (for log files)	General Purpose: Standard-series (Gen5), 4 vCores	CPU: 8 cores RAM: 32 Go
30 Lizs	CPU: 2 cores RAM: 8 Gb RAM Storage = 100GB (for log files)	General Purpose: Standard-series (Gen5), 12 vCores	CPU: 12 cores RAM: 64 Go Note: Standard Edition supports the lesser of 4 sockets or 24 cores, allowing full benefits from 12 cores without needing SQL Server Enterprise (see "Editions and Supported Features – SQL Server 2016", section "Scale Limits").

(*) The Gate Database storage mostly dependents on the volume of external codes, usage of reporting features (Fortuna, CRPT) and the data retention time :

	Raw Storage(*) per billion codes	**Typical Retention time**	**Example: Storage needed for a volume of 1 billion codes per year, during the typical retention time**
SUPI12 Code Generation	2.5 Gb	1 year	2.5 Gb
External Code Storage (TPD, DLR, SAT)	10 Gb	1 year	10 Gb

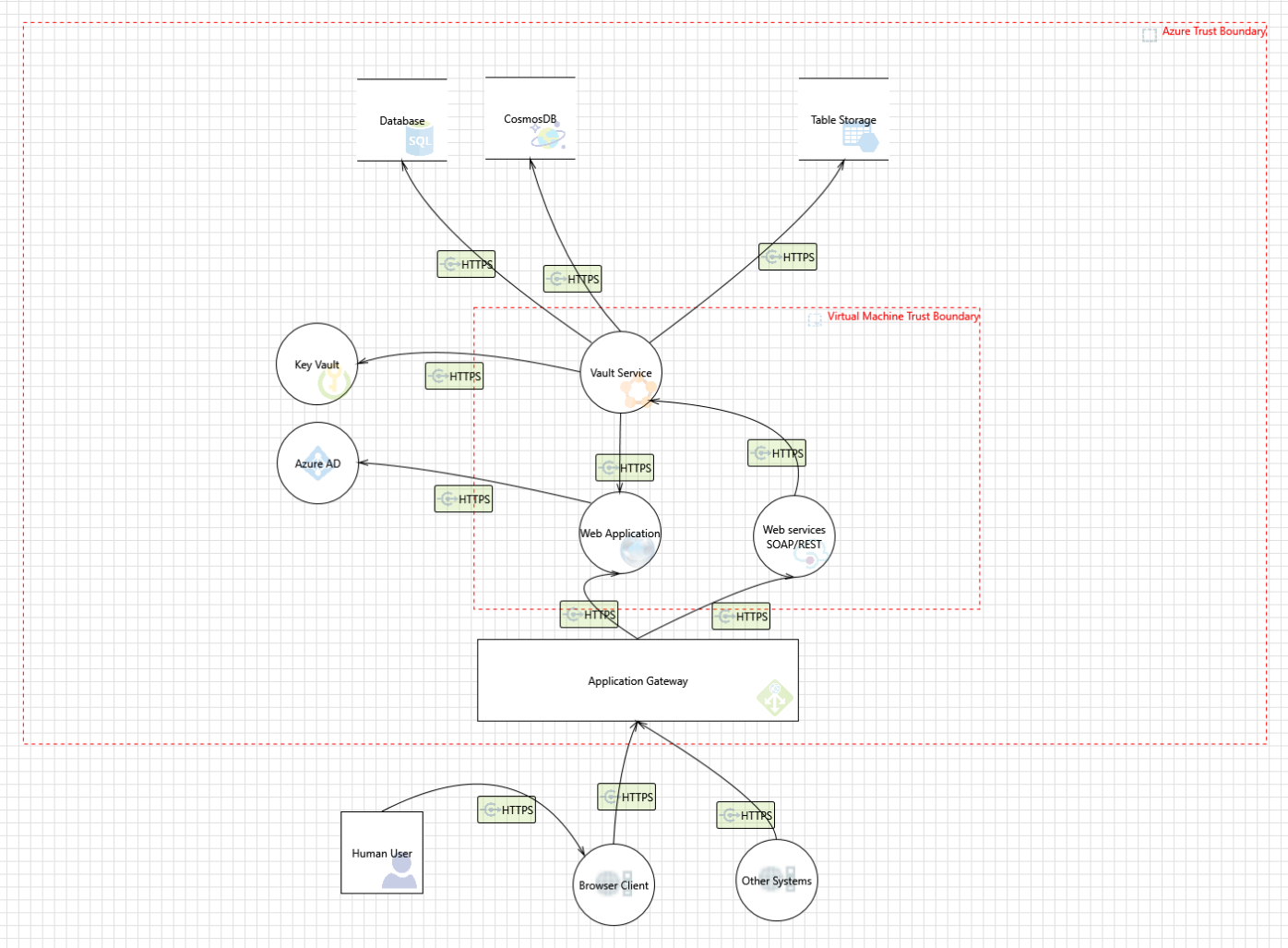
SQL Code materialization (TPD, DLR, SAT)	60 Gb	1 year	60 Gb

The **Recommended Storage** is 50% higher than the calculated **Raw Storage.**

Hardware and Software Requirements: Liz Requirements

- Architecture x64 is recommended, x32 is supported.
- Virtualization is supported on HyperV, VMWare, KVM.
- Both .NET Framework 4.8 and .NET Framework 3.5 are required.
- Built-in SQL Server Compact 4.0.0.1.
- Windows 10 is recommended.
- 2 Ethernet NICs (one for printer's LAN and the other for Company's LAN)
- 40GB of free disk space
- CPU: INTEL compatible CPU
- RAM: 4GB or higher

IT Infrastructure: Vault for IT



IT Infrastructure: Prerequisites for Implementation of the Vault

This document outlines the essential technical requirements that must be met before starting the installation of the Vault.

- Windows Domain
- Software Requirements
- Database Configuration
- Network Requirements
- Hardware Requirement
- Certificates and Licenses
- Service Accounts

Windows Domain

Deploying Vault servers within a Windows domain is recommended but not mandatory. The key advantage of this deployment strategy includes:

- Simplified maintenance of service accounts, particularly for the account running the Vault NT Service and accessing the Vault database on SQL Server.

When deploying in a Windows domain, it's important to:

- Ensure the domain service account, which will operate the Vault NT Service, is granted **dbo** rights to the Vault database.

In scenarios where deployment in a Windows domain is not feasible:

- Use an identical local service account on both the Vault Server and all associated SQL Servers, maintaining consistency in passwords across these servers.

> **Note**: Despite deploying Vault servers in a Windows domain, it's possible to utilize local accounts on these servers. This allows for a flexible mix of domain and local service/user accounts.

Software Requirements

Before commencing the Vault installation, ensure the server meets the following software prerequisites:

1. **Operating System**: Windows Server 2016 or later version.
2. **Framework**: Microsoft .NET Framework 4.8 must be installed.
3. **Internet Information Services (IIS)**: IIS should be installed and configured with these specific features:
    - Basic Authentication
    - Client Certificate Mapping Authentication
    - IIS Client Certificate Mapping Authentication
    - Request Filtering
    - Windows Authentication

Additionally, follow these steps to configure server roles:

1. Download the file named Roles.csv.
2. Copy 'Roles.csv' to the server's desktop.
3. Execute the following PowerShell script to add the required Windows features as listed in the CSV file.

```
Import-Csv $env:USERPROFILE\Desktop\roles.csv | foreach{Add-WindowsFeature $_.name }
```

Database Configuration

For the Vault service, the following database server requirements must be met:

- An operational database server running SQL Server 2016 Service Pack 1 (SP1) or a more recent version is required.
- This database server must be accessible to the Vault service.

For assistance with SQL Server installation, please consult the Database Administration section.

Network Requirements

Depending on the type of Vault being deployed, the network connectivity requirements are as follows:

- **Public Vaults**: An active internet connection is necessary.
- **Private Vaults**: An intranet connection is sufficient.

Hardware Requirement

To ensure optimal performance of the service, the following hardware specifications are required:

1. **For the Service PC**:

- Minimum of 100GB available space on the system.
- CPU with a speed of 1.5GHz or higher.

2. **For the Database**:
   - At least 20GB of database size is required, with a recommendation of 100GB or more for optimal performance.

Important Notes:

1. **Transaction Log Management**:
   - Aim to minimize the size of transaction logs, particularly when the recovery model is set to Full.
   - If database mirroring is in use and the Full recovery model is required, be aware that the transaction log will continue to grow unless:
     - Regular backups are performed and are functioning correctly.
     - In case of mirroring, ensure all partners are operational to keep the mirrored database synchronized with the principal SQL Server.
   - Refer to the document KB22 - Administration Performance Monitoring for details on backup and monitoring.
2. **Database Log Table Size**:
   - The log table in the database can significantly impact the overall database size.
   - Typically, the log table should not exceed a growth of 300 MByte per month.
   - Regular monitoring of the database size is crucial, as detailed in KB22 - Administration Performance Monitoring.
   - The size of the database also varies depending on the number of Gates connected to the Vault.

Certificates and Licenses

For a new Vault installation using the INEXTOR Certificate Tool, the following certificate requirements must be met:

- **Root Certificate**: This is a self-signed certificate.
- **Technical Provider (TP) Certificate**: This certificate must be signed by the root certificate.
- **Vault Certificate**: This certificate is signed by the technical provider certificate.

Additionally, INEXTOR uses other types of certificates, which are generated as follows:

- **Gate Certificate**: Generated on the Vault website when managing a Gate connection, signed by the Vault certificate.
- **Liz Certificate**: Generated on the Vault website when creating Liz licenses, signed by the TP certificate.
- **MSK Certificate**: Generated by the Vault when setting up the Liz, signed by the Vault certificate.
- **Actor Certificate**: Generated on the Vault website when managing entities, signed by the Vault certificate.

See Certificate Tool for step-by-step instructions.

Service Accounts

For optimal use of Windows Integrated security, the Vault service must be operated under a service account rather than a local system account. Key guidelines include:

Primary Service Account for Vault:

- The account should have local administrator privileges on the PC.
- Optionally, the account may be a member of the domain but is not required to be a domain administrator.

Important Notes:

- **Note 1**: If the account does not have local administrator rights, ensure it has sufficient privileges. This includes:
  - Membership in the Users group.
  - Rights for Bypass Traverse Checking.
  - Rights to Increase a Process Working Set.
  - Additional rights might be needed for network access and other functions.
- **Note 2**: The Vault configuration files specify two key permissions:
  - **Required Privileges**: should be set to:

```
"SeChangeNotifyPrivilege,SeIncreaseWorkingSetPrivilege,SeImpersonatePrivilege,SeCreateGlobalPrivilege"
```

  - **Denied Privileges**: should be set to:

```
"SeMachineAccountPrivilege,SeTrustedCredManAccessPrivilege,SeAssignPrimaryTokenPrivilege,SeAuditPrivilege,
SeBackupPrivilege,SeCreatePagefilePrivilege,SeCreatePermanentPrivilege,SeCreateSymbolicLinkPrivilege,SeCreateT
SeEnableDelegationPrivilege,SeIncreaseBasePriorityPrivilege,SeIncreaseQuotaPrivilege,SeLoadDriverPrivilege,SeL
SeProfileSingleProcessPrivilege,SeRelabelPrivilege,SeRemoteShutdownPrivilege,SeRestorePrivilege,SeSecurityPriv
SeSystemEnvironmentPrivilege,SeSystemProfilePrivilege,SeSystemtimePrivilege,SeTakeOwnershipPrivilege,SeTcbPriv
SeTrustedCredManAccessPrivilege,SeUndockPrivilege".
```

- The account must have Read/Write access to the log directory as specified in the VaultService.exe.config configuration file (see VaultService.exe.config Setup).
- A login for this account on the Vault database must be created during installation (see Database Setup)

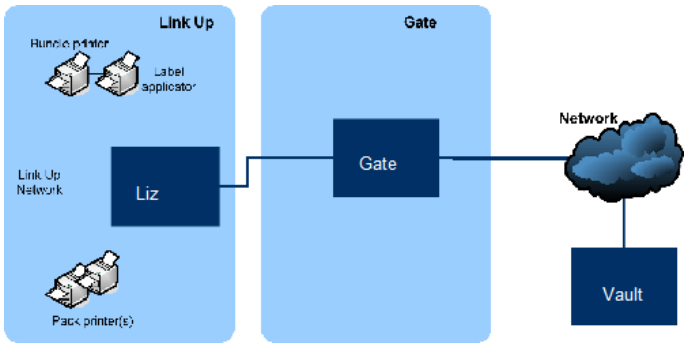
1. **Additional Service Accounts**:
   - **For IIS Application Pool**:
     - A dedicated service account to manage the IIS application pool hosting the Vault website and web services.
     - This account should have Read/Write access to both the installation folders of the Vault Website and web services (see Installation Media for the list of installation folders) and to the log directories as defined in the web.config files. (see Vault Web applications installation)
   - **For SQL Agent Service**:
     - A separate service account for running the SQL agent service on the database server.

General installation guide: Procedures

- General Installation Procedures: Technical Overview
- Manufacturers Checklist
- General Installation Procedures: Upgrade Rules
- General Installation Procedures: Vault Procedures
- General Installation Procedures: Gate Procedures
- General installation procedures: Liz procedures
- General installation procedures: Vault proxy procedures
- General Installation Procedures: Migrating from Codentify to Inextor

General Installation Procedures: Technical Overview

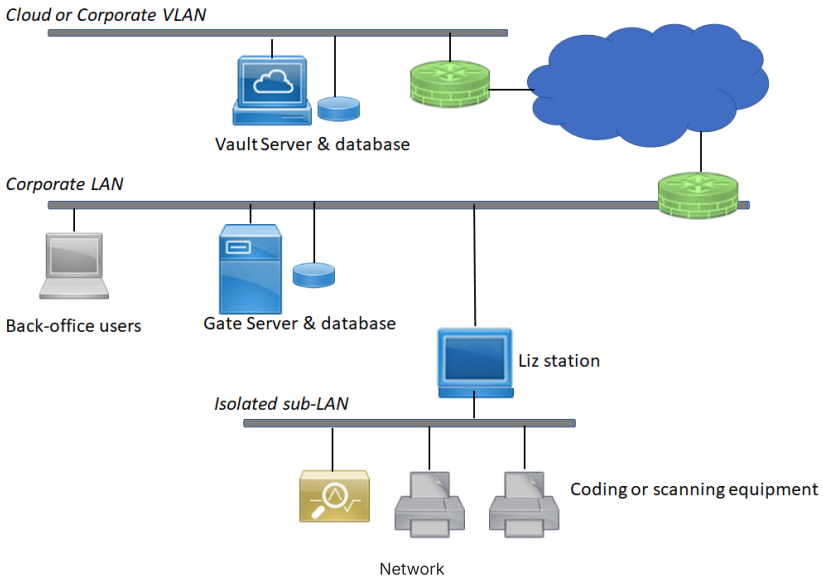
The following figure shows the different components of the solution.



System Overview

Components	Description	Documentation	Responsibility
Coding Equipment (Printers)	Coding equipment are used to print/mark the products with Inextor codes and/or with any other type of information.	Installation documentation is available from your Coding Equipment Supplier or Technical Provider.	Equipment Supplier Technician (or OEM) & Factory Engineering Services
Linkup Network	Ethernet network connecting the Liz PC to the printers	Specific for each Factory	Factory services / Network Specialist
Liz (Liz)	The software component that generates Inextor codes and sends them to the connected printers	Liz Installation Guide	Factory services
Gate (Gate)	Software installed at factory level that acts as a gateway between all the Lizs installed in a factory and the Vaults	Gate Installation Guide	Factory services
Vault	Central system that permits users to verify the validity of a Inextor code	Vault Installation Guide	Government or Central Organization for Manufacturer
Network	Ethernet network connecting Lizs to the Gate and the Gate to the Vaults	Specific for each Factory	Network Specialists

Network



Network

Gate Server and Client

Gate Server should be installed on a machine with at least Microsoft Windows Server 2008 R2, 2012 or later version.
The Gate Server must have Internet access.
The Gate Administration can be installed on the same machine as the server as well as on other machines on the same network.

SQL Server Database

The SQL Server Database can be installed on the same Microsoft Windows Server as the Gate Server.

Vault Client

The Vault client is any machine with internet access and a browser.

Liz Server, Client

The Liz Server should be installed on a machine with Windows 10. The Liz UI can be installed on the same machine as the Liz server as well as on other machines on the same network.

Printer LAN (Linkup network)

It is preferable and suggested to have a **separate** LAN to connect the Liz to the printers and to use fixed IP addresses.

Manufacturers Checklist

Checklist for the Manufacturers

Planning and Preparations

Identifying people

Inextor Team at factory

Parameter	Description	Value
Responsible	Responsible for the Inextor Factory implementation.	
Coordinator	Supervise the installation, Coordinate people in order to ensure installation.	
Gate IS Installation	Responsible for Gate Installation	
Liz IS Installation	Responsible for each Liz Installation/Upgrade	
Network	Responsible for linkup Network installation/configuration	
Coding Equipment Coordinator	Responsible for coding equipment installation and setup	
Machine Preparation	Responsible for installation on the machine, typically Factory Engineering Services	
Production	Responsible for the operation of the system	
Specification	Responsible for product definition and in case of automatic production change coordinate the transfer of information to the system	
Technical support	Responsible for the management of the coding equipments (layout, templates)	

Inextor Team within your company

Parameter	Description	Value
Responsible	Responsible for the Inextor implementation worldwide.	
Technical support	Responsible for the Inextor implementation worldwide from a technical point of view.	

Inextor Team suppliers and Vault contact

Parameter	Description	Value
Technical Provider for installation		
Coding Equipment Supplier		
Government Vault contact		

General preparations by factory

Description	Responsible
Send filled list of Inextor team to your technical provider and to other relevant parties.	
Send filled preparation list to your technical provider and to other relevant parties.	

Miscellaneous preparations by factory

Description	Responsible
Optional equipment for securing that readable Inextor codes are printed (e.g. OCR-system)	

Vault preparations

Description	Action or Document
Decide which type of roles to be used for the accounts on the Vault (MAN/IMP/LE).	Introduction (Vault Configuration) Vault Reference Administration Entity
Registration request on Vault	Contact Vault owner
Create additional users on Vault	Vault Reference Administration User Profiles and Users
Create Gate on Vault	Vault Reference Administration Factory Factories and Lines
Create SKUs on Vault	Vault Reference Administration Product Product Groups and Products
License SKU from LE to MAN/IMP	Vault Reference Administration Product Product Licensing
License SKU from MAN/IMP to Gate	Vault Reference Administration Product /wiki/spaces/INEXTOR/pages/395819955

Liz License file

Description	Action or Document
Get Liz License file(s) from Authorized source (E.g. GOV).	Contact Authorized source

Gate preparations by factory

Description	Action or Document
Define where to physically place the PC(s) for the Gate server(s), client and SQL Server(s).	
Decide whether using a single or redundant server.	Vault Procedures: Setup Vault
Decide if to use SQL server mirroring	Vault Procedures: Setup Vault
Prepare mechanics and cabinets for PC	
Prepare electrics for PC	
Prepare internet connection for the Gate. Proxy server settings and firewall settings.	
Purchase the PC for the Gate server(s) and SQL Server(s).	INEXTOR Requirements
Purchase SQL Server licenses if required.	Contact Microsoft
Define a domain user account that is local administrator on the PC for Gate Server and SQL Server installation	Overview of INEXTOR System Archived Gate Procedures - Gate Setup peFO
Define a domain user account for Gate service and optionally for the SQL Server.	Overview of INEXTOR System Setup the Gate
Install Operating System and .NET framework on the server PC:s.	Overview of INEXTOR System
Prepare server with security packs and user accounts	Overview of INEXTOR System Setup the Gate
Install SQL Server	SQL Server Installation
Define IP addresses for Gate	
Make Firewalls exceptions for the Gate server and optionally for the SQL Server	Gate Firewall Settings

Description	Action or Document
Define a two character manufacturing ID for Gate server	
Prepare Configuration files	

Liz preparations by factory

Description	Action or Document
Define where to physically place the PC for the Liz server and client. Preferable close to the production line.	
Purchase the PC for the Liz servers and clients	INEXTOR Requirements
Prepare mechanics and cabinets for PC	
Prepare electrics for PC	
Prepare network between Liz and printers	Overview of INEXTOR System
Prepare network between Liz and Gate	Overview of INEXTOR System
Decide whether to use the Liz automatic production change interface.	Overview of INEXTOR System
In case of automatic production change, prepare the client system (e.g. MES system) to be used for sending automatic production change.	
Define a domain user account that is local administrator on the PC for Liz Server/Client installation	
Define domain user account for Liz client	
Install Operating System and .NET framework on Liz server PC.	Overview of INEXTOR System
Prepare Liz server with security packs and user accounts	
Define IP addresses for Liz:s on the factory LAN	Overview of INEXTOR System
Define IP addresses for Liz-printer LAN	Overview of INEXTOR System
Make Firewalls exceptions for the Liz server	Liz network: Required firewall settings
Define a two-four character machine ID for Liz server	Overview of INEXTOR System
Prepare Configuration files	

Printer preparations by factory

Description	Action or Document
Define what to be printed (Inextor code + additional information)	
Is Inextor able to print the additional information required?	Printers, Configuration
Define where on the product to print the Inextor code + additional information	
Define which printer(s) to use and optionally purchase it.	Printers, Configuration
Define where to place the printer(s) and printer head in the production line	
Prepare mechanics for printer and printer head	
Prepare electrics for printer (power, trig signals etc.)	
Decide whether to manage print templates on the printers or on the Liz. If managing template on printer, check if Liz and Printer supports it.	

Installation and training

Installation

Description	Action or Document
Install Gate Server	Setup the Gate
Install Gate Clients	Setup the Gate
Configure Gate Server	Configure the Gate
Test Gate	
Register Gate on Vault	Configure the Gate

Description	Action or Document
Integration test Gate – Vault	
Install Liz Server	Setup the Liz
Install Liz Clients	Setup the Liz
Configure Liz Server	Setup the Liz
Test Liz	Setup the Liz
Integration test Liz – Gate	
Request and Register Liz on Vault.	Setup the Liz
Request and Register MSK on Vault	Configure the Vault
Install Printer	Printers, Configuration
Configure Printer	Printers, Configuration
Test Printer	
Integration test Printer – Liz	
System test Printer – Liz – Gate – Vault	

Training

Description	Responsible
Train trainer for Liz User	
Train trainer for Gate User	
Train trainer for Vault User	
Train trainer for Gate Expert	
Train trainer for Liz Expert	
Train trainer for Printer Expert	

Production

Description	Responsible
Template Editor(s) for the printer to create templates	
Gate Administration GUI (Gate Client) to manage Templates and Printer Drivers	
Gate Administration GUI (Gate Client) to manage products to produce	
Vault to manage SKU (Government Vault)	
Gate Administration GUI (Gate Client) to manage SKU (Private Vault)	
Use of manual or automatic production change on the Liz. Additional Production change data (e.g. order number)	
Regularly verify Gate connection to Vault	
Regularly verify that production volume data reaches the Vault	
Entering waste according to government specification	

Maintenance & Troubleshooting

Description	Responsible
Backup on Liz configured data	
Backup on printer data	
Backup on Gate configured data	
Backup on SQL Server Servers	
Backup on certificates, MSK:s and licenses	
Troubleshooting Liz	

Description	Responsible
Troubleshooting Gate	
Troubleshooting SQL Server	

General Installation Procedures: Upgrade Rules

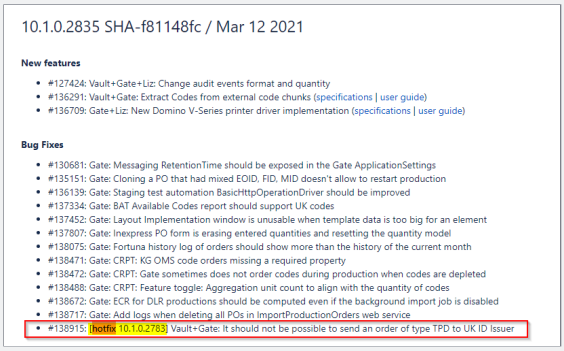
Before doing an upgrade of the Inextor Suite, please have in mind the following rules:

- The Vault keeps backward compatibility with older versions of the Gate and Liz.
- The Gate keeps backward compatibility with the older version of the Liz.
- The Vault is the component which is upgraded the most regularly (*once per month at PMI*).
- The Gate is upgraded in second, at a lower frequency because of factory constraints (*to limit downtime, mostly*).
- The Liz is upgraded usually after the Gate.
- **The recommendation is always to have Vault Version >= Gate Version >= Liz version.**



Exception to the rule
If a high or critical issue is found on the Gate or on the Liz, we usually deploy a hotfix on the impacted component. In such a case you could have a version of Gate > Vault or Liz > Gate. But this is easily identifiable as the version number would contain the word *hotfix*. Example: 10.1.0.3165-hotfix.1.

To see the information about the hotfixes, you can go directly to INEXTO SUITE Downloads > Inextor > Inextor Releases by Application > [Choosing the application]



10.1.0.2835 SHA-f81148fc / Mar 12 2021

New features

- #127424: Vault+Gate+Liz: Change audit events format and quantity
- #136291: Vault+Gate: Extract Codes from external code chunks (specifications | user guide)
- #136709: Gate+Liz: New Domino V-Series printer driver implementation (specifications | user guide)

Bug Fixes

- #130681: Gate: Messaging RetentionTime should be exposed in the Gate ApplicationSettings
- #135151: Gate: Cloning a PO that had mixed EOID, FID, MID doesn't allow to restart production
- #136139: Gate: Staging test automation BasicHttpOperationDriver should be improved
- #137334: Gate: BAT Available Codes report should support UK codes
- #137452: Gate: Layout Implementation window is unusable when template data is too big for an element
- #137807: Gate: Inexpress PO form is erasing entered quantities and resetting the quantity model
- #138075: Gate: Fortuna history log of orders should show more than the history of the current month
- #138471: Gate: CRPT: KG OMS code orders missing a required property
- #138472: Gate: CRPT: Gate sometimes does not order codes during production when codes are depleted
- #138488: Gate: CRPT: Feature toggle: Aggregation unit count to align with the quantity of codes
- #138672: Gate: ECR for DLR productions should be computed even if the background import job is disabled
- #138717: Gate: Add logs when deleting all POs in ImportProductionOrders web service
- #138915: [hotfix 10.1.0.2783] Vault+Gate: It should not be possible to send an order of type TPD to UK ID Issuer

General Installation Procedures: Vault Procedures

The following sections contain all the necessary information for the setup and configuration of a new Vault as well as instructions to upgrade the Vault to a more recent version.

Introduction

This section introduces the system, defines the target audience, and points the reader to appropriate sections for more information.

Prerequisites

This section details the technical prerequisites that must be in place prior to installation.

Firewall Settings

This section explains the required Firewall settings that must be configured.

Setup

These sections give step-by-step detailed instructions for installing the Vault in standard configurations.

Configuration Options

This section presents options and variants for configuring the Vault.

Vault Procedures: Disaster Recovery

This section details disaster recovery procedures in order to recover the Vault.

Contingency Procedures

This section details the contingency procedures of the Vault.

/wiki/spaces/INEXTOR/pages/395827701

This section gives instructions for troubleshooting the Vault.

Vault Procedures: Introduction

Overview

The purpose of this documentation is to describe the installation, upgrade and disaster recovery procedures for the Inextor v10 Vault. It provides IT personnel in charge of the installation with step-by-step instructions to complete the installation.

- For an overview of the technical prerequisites that must be fulfilled prior to installation, refer to the Prerequisites chapter.
- For step-by-step installation instructions, refer to the Setup chapter.
- For configuration options and variants, refer to the Configuration Options chapter.

Audience

This document is intended for the IT team in charge of installing or upgrading the INEXTOR v10 Vault.

More information

For further information on business concepts and scenarios, refer to the /wiki/spaces/INEXTOR/pages/395829924.

Vault Procedures: Migration Checker v4 to Vault v10

The migration from a checker v4 to a vault v10 is composed of two major steps:

1. **Initial migration (~8 hours)**: A backup of the checker database is created. This backup will be used to migrate all the data onto an empty Vault database. Once this database is fully migrated, it is optionally transferred onto the dedicated Azure server using the Data Migration Assistant.
2. **Continuous migration (~4 hours)**: A service is responsible of migrating the changes periodically from the running checker to the vault.

The Continuous migration has 2 advantages:

- Migration can be executed while the Checker remains online. Without continuous migration, we would need to complete the migration and keep the Checker offline during the whole time. In case of data error (inconsistent data unexpected by the migration tool), the migration process can be retried without any impact for client systems.
- All legacy systems connected to the Checker can continue to work against the Checker, including factories (MCG, Gate), Inextrack, Inextrace, etc... these systems can be switched to the Vault in a controlled fashion, and regression-tested.

Migration procedure

1. Ensure that you have the connection information to the Checker database.
2. Stop the Checker service as described in /wiki/spaces/CDFYMASTER/pages/399835945.
3. Create a backup of the Checker database.
4. Set up change tracking on the database as described below.
5. Turn the Checker service back on.
6. Install a vault database (see Vault Setup) , for example "Vaultv10ForMigration", without inserting data, on a SQL server (for example the checker database server). Follow only the steps for the database.
7. Restore a copy of the checker database to a physical (non-Azure) SQL Server. The migration must be performed on a non-Azure SQL Server.
8. Execute the migration tool providing the connection strings of the Vault database and the copy of the Checker as described below.
9. Migrate the database onto Azure, as described below.
10. Install the continuous migration service and start it, more details below.

Change tracking

Prerequisite

In order to ensure there is no service connecting to the database, it is necessary to execute the following script (just replace the string in the last statement with the name of the Checker database).

```
DECLARE @AllConnections TABLE(
    SPID INT,
    Status VARCHAR(MAX),
    LOGIN VARCHAR(MAX),
    HostName VARCHAR(MAX),
    BlkBy VARCHAR(MAX),
    DBName VARCHAR(MAX),
    Command VARCHAR(MAX),
    CPUTime INT,
    DiskIO INT,
    LastBatch VARCHAR(MAX),
    ProgramName VARCHAR(MAX),
    SPID_1 INT,
    REQUESTID INT
)

INSERT INTO @AllConnections EXEC sp_who2

SELECT * FROM @AllConnections WHERE DBName = '<Checker Database>'
```

Only the server that is currently connected to perform the operations to enable change tracking should be listed (or any mirroring service if mirroring is enabled).

Database setup

To setup the checker, the following script needs to be executed on the checker database (download script). Before executing it, go to to the "Query" menu, and select "SQLCMD mode".

All the parts of the script are explained below. The name of the Checker database needs to be modified **before running** the script. This is done in the first line, replace *<CheckerDatabase>* by the real name of the database.

```
:setvar DatabaseName "<CheckerDatabase>"
```

Change tracking is a functionality of SQL Server (official page).

First, the function needs to be enabled for the database. This is done by executing

```
ALTER DATABASE <databasename>
SET CHANGE_TRACKING = ON
(CHANGE_RETENTION = 180 DAYS, AUTO_CLEANUP = ON)
```

where *databasename* is the name of the checker database that is being migrated.

More details about the parameters of change tracking can be found in the official documentation.

To be sure that there won't be any problem during the migration, the snapshot isolation is used. To ensure that this functionality is enabled, run:

```
ALTER DATABASE <databasename>
SET ALLOW_SNAPSHOT_ISOLATION ON
```

where *databasename* is the name of the checker database that is being migrated.

Tables setup

Change tracking needs to be enabled on every table. To achieve that, run

```
exec sp_msforeachtable
'if object_id(''?'') not in
(select object_id from sys.change_tracking_tables
union select ISNULL(OBJECT_ID(''[dbo].[VERSION]''),0)
union select ISNULL(OBJECT_ID(''[dbo].[HostNames]''),0)
union select ISNULL(OBJECT_ID(''[dbo].[LogEntries]''),0))
```

```
alter table ?
enable change_tracking with(track_columns_updated = off)'
```

This script enabled change tracking on every table As change tracking uses the primary key to keep track of the modifications, the tables without primary keys are avoided (i.e. Version and LogEntries).

Synchronization setup

The continuous migration will use a new table called SynchronizationVersion to keep track of what has already been migrated. To create this table run:

```
create table SynchronizationVersion (version bigint primary key not null)
insert into SynchronizationVersion(version) values(0)
```

Verification of tracked tables

The following script will list the tables on which track changes is enabled. It can be used for verification.

```
SELECT s.name as Schema_name, t.name AS Table_name, tr.*
FROM sys.change_tracking_tables tr
INNER JOIN sys.tables t on t.object_id = tr.object_id
INNER JOIN sys.schemas s on s.schema_id = t.schema_id
order by Table_Name
```

Migration tool usage

Before calling the migration tool, the following SQL script should be applied to the destination database (Vault database, for example "Vaultv10ForMigration"):

```
delete from UserOperations
delete from Users
delete from Roles
delete from Actors
delete from ProductGroupDetails
delete from ProductGroups
```

Then, execute the following script: vault.database.sat.sql, still on Vault database (for example "Vaultv10ForMigration").

The migration tool is executed from a command line and is called "Inexto.S3r.Migrate". It has the following format :

`Inexto.S3r.Migrate.exe [MigrationAssembly] -SourceDb=[SourceDatabaseConnectionString] -DestinationDb=[DestinationDatabaseConnectionString] [-Variable:[Key]=[Value] -Variable:[Key]=[Value] ... ] -CheckerCode=[Value]`

where the parameters are

Parameters	Description
MigrationAssembly	The list of assemblies needed by the tool. In this case : *Inexto.S3r.Migrate.Library* and Inexto.*S3r.Migrate.Checker400ToVault1000*
SourceDb	Connection string of the Checker database copy (restored)
DestinationDb	Connection string of the Vault database for example "Vaultv10ForMigration", located normally on checker database server
Variables	list of variables, not used in the migration of the checker
CheckerCode	The identifier of the Checker to correctly migrate the production data

Example invocation:

```
Inexto.S3r.Migrate.exe "Inexto.S3r.Migrate.Checker400ToVault1000, Inexto.S3r.Migrate.Library" -SourceDb="Data Source=OLDSERVER\PRD;Initial Catalog=Checkerv4;Integrated
Security=True" -DestinationDb="Data Source=NEWSERVER\PRD;Initial Catalog=Vaultv10;Integrated Security=True" -CheckerCode=PRIV
```

Migrate to Azure

Vault migration

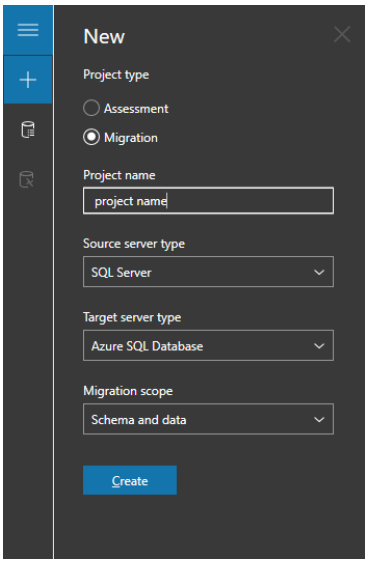
The Plugins table in the Vault database (for example "Vaultv10ForMigration") may be problematic due to its XML data being too big for the migration. To avoid the problem, cast it as nvarchar(max) :

```
alter table plugins
alter column Data nvarchar(max) null
```

And turn it back into XML once it has been migrated.

To migrate the data from SQL Server onto Azure, Microsoft Data Migration Assistant (DMA) is used (official page).

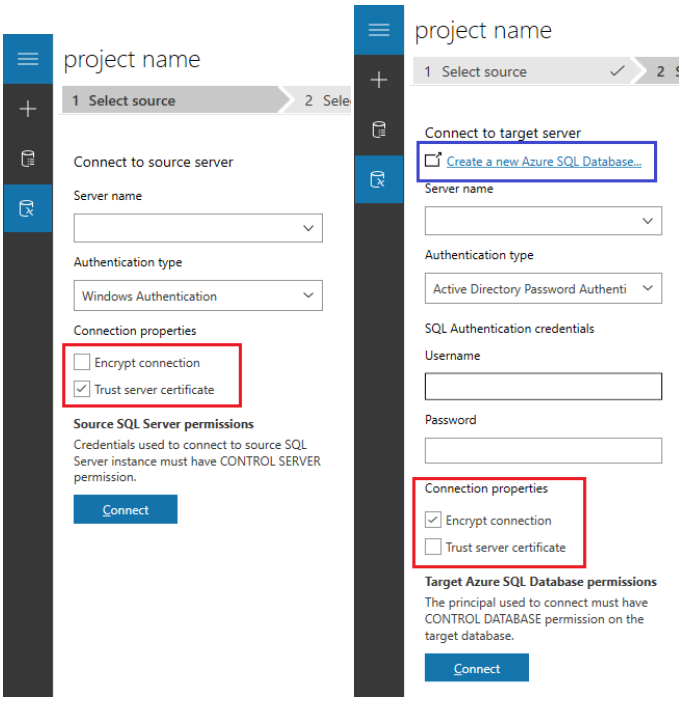
First create a new project by clicking on the "+" sign on the left side. Then, enter the following options (choose an appropriate project name):



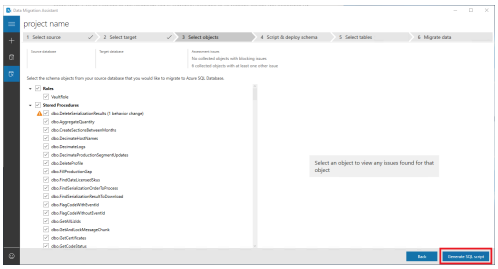
Once the options are set, click on "Create". This will bring you to the setup of the source database (i.e. the on-premise SQL Server) and the destination database (on Azure).

For the next two steps, connect to the two database servers, and choose the databases. On SQL Azure, create empty database, for example "VaultV10Db". For Azure, there is a link to the documentation.

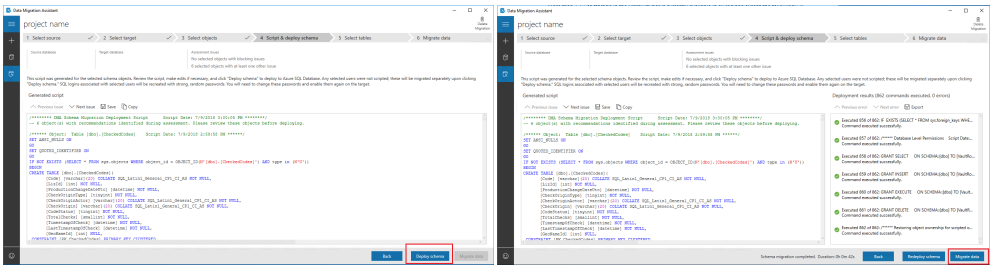
Please use the connection properties as shown on the next pictures.



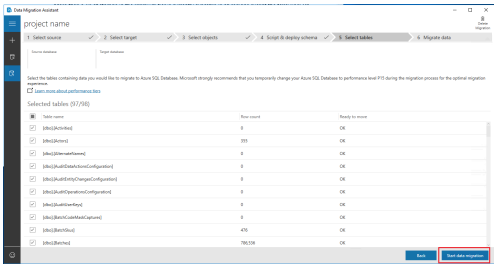
Once the two databases are set, DMA will collect all the objects to migrate. Select all the objects and click on "Generate SQL script".



This will lead you to the 4th step. Here you can see the SQL script. First click on "Deploy schema", once this is done, click on "Migrate data".



This will lead you to the 5th step. Select all the tables and click on "Start data migration".



This operation may take a while, but all your data will be migrated.

If you changed the Plugins table, remember to convert it back to XML:

```
alter table plugins
alter column Data xml null
```

User creation

The next thing needed to migrate on Azure is to register a user on the database.

If you have not created the user on the Azure server, run on master:

```
CREATE LOGIN <yourLogin> WITH PASSWORD = '<yourPassword>';

CREATE USER <yourUser> FROM LOGIN <yourLogin>;

ALTER ROLE dbmanager ADD MEMBER <yourUser>;
```

Once this has been made, add this user on your migrated database :

```
USE [<yourDB>]
CREATE USER <yourUser> FROM LOGIN <yourLogin>;
ALTER ROLE db_owner ADD MEMBER <yourUser>;
```

Connectivity

In order to grant access to the Vault database on Azure from the Checker database located in manufacturer's premises, it is necessary to define a firewall rule on the database to allow the range of IPs of the Manufacturer's to connect to the database.

This is performed using the following SQL statement:

```
USE [master]


EXECUTE sp_set_firewall_rule @name = N'ContinuousMigration', @start_ip_address = 'XXX.XXX.XXX.XXX', @end_ip_address = 'YYY.YYY.YYY.YYY'
```

It is possible to list the existing firewall rules using the following SQL statement:

```
USE [master]


SELECT * FROM sys.firewall_rules
```

Continuous migration service

Configuration

The configuration file of the service can be found under Inexto.S3r.ContinuousMigrationService and will look like this

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
    <configSections>
        <section name="log4net" type="log4net.Config.Log4NetConfigurationSectionHandler,log4net" requirePermission="false"/>
    </configSections>
    <startup>
        <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.5" />
    </startup>
    <appSettings>
        <add key="sourceConnectionString" value="" />
        <add key="destinationConnectionString" value="" />
        <add key="sleep" value="15" />
        <add key="changes" value="1000000" />
        <!--<add key="startingVersion" value="[value]" />-->
        <!--<add key="maxVersion" value="[value]" />-->
        <add key="doMigrateProdsUpdates" value="true" />
        <add key="doMigrateProds" value="true" />
```

```
        <add key="doMigrateAll" value="true" />
        <!--<add key="applyPatch" value="false" />-->
        <add key="checkerCode" value="CODE" />
    </appSettings>
    <log4net configSource="log4net.config"/>
</configuration>
```

A few values need to be configured under the *appSettings* section

key	description
sourceConnectionString	connection string of the Checker database on SQL Server.
destinationConnectionString	connection string of the Vault database on Azure. Be sure to use the user with the right privilege. Also, use **Trusted_Connection=False** and **Encrypt=True** to encrypt the data.
sleep	time [in min] between each delta update.
changes	number of changes treated in the same transaction. It is recommended to keep the default value.
`startingVersion`	Use only in the case of an error or restart after a long break. Migration will start using this value instead of the CHANGE_TRACKING_CURRENT_VERSION
`maxVersion`	Use only in the case of an error or restart after a long break. Migration will go until this version on the first run.
`doMigrateProdsUpdates`	Set if the PRODsUpdates will be migrated or not.
`doMigrateProds`	Set if the PRODs will be migrated or not.
`doMigrateAll`	Set if the other tables (i.e. not PRODs nor PRODsUpdates) will be migrated or not
`applyPatch`	Use only if a patch needs to be applied
`checkerCode`	The code of the Checker.

Installation

The migration service can be installed from the command line and is called "Inexto.S3r.ContinuousMigrationService". To install it, run

Inexto.S3r.ContinuousMigrationService.exe install --[option]

where the options are

option	description
--autostart	service will start automatically
--manual	service will have to be started manually
-username:	User name of the account running the service
-password:	Password of the account running the service

Other option can be found here: http://docs.topshelf-project.com/en/latest/overview/commandline.html

Example invocation:

```
Inexto.S3r.ContinuousMigrationService.exe install -username:DOMAIN\ServiceAccount -password:itsASecret --manual
```

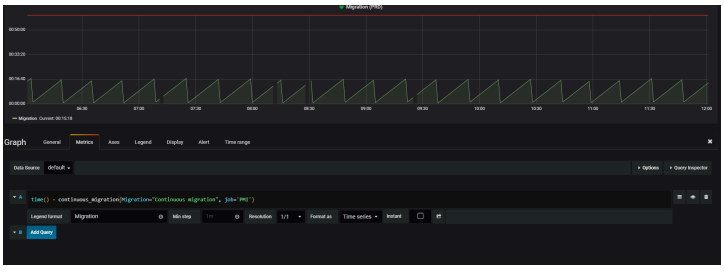
Once the service has been installed, you can access it through the service UI (service name : Continuous migration).

Alert

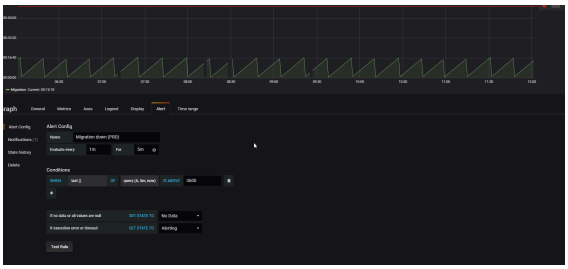
Every time the continuous migration runs, an entry is updated in the table CheckerMigrations of the Vault. This table is then monitored as to find out when a problem occurred in the migration. In such a case an alert is created and an email is sent to a predefined mailing list.

Set up monitoring

On Grafana, the setup for the migration monitoring looks like this



We then setup an alert system that will send an email when the migration service has not updated the table for one hour. An email is also sent when the migration is working again.



Restart after a long break

The PRODsUpdate problem

We have notice a weird behavior in the way that SQL Server Change Tracking works :

From the Microsoft documentation, the value of SYS_CHANGE_CREATION_VERSION should be the value (change number) of the creation of the element. But when we insert a ProdsUpdate, the corresponding PROD element is updated and the SYS_CHANGE_CREATION_VERSION is also modified to the current change number (and is equal to SYS_CHANGE_VERSION). This does not impact the migration as all the changes are taken in one single run. In the case of an important service disruption we have to split in different runs, and we are in a situation where a PRODsUpdate is inserted before its corresponding PROD, leading into an error. In those cases, we may have to follow the procedure below.

To know if it is the case, compare the value of the synchronization version (from the SynchronizationVersion table) to the CHANGE_TRACKING_CURRENT_VERSION. If the current version is greater by 300'000 or more, then the procedure should be applied.

Procedure

1 . Starting version

Before going through all this procedure, mark down the current Synchronization version, by executing

```
Select version from SynchronizationVersion
```

2 . Migrate all but PRODsUpdates

Change the config file keys to :

```
<add key="doMigrateProdsUpdates" value="false" />
<add key="doMigrateProds" value="true" />
<add key="doMigrateAll" value="true" />
```

and start the migration tool. It has been noticed that a change of more that 1 million rows can lead to a timeout error.

Make sure that the tool goes through more than one migration run. (You can change the sleep value of the config to avoid having to wait between two runs.)

3. Migrate only PRODsUpdates

Change the config file keys to :

```
<add key="doMigrateProdsUpdates" value="true" />
<add key="doMigrateProds" value="false" />
<add key="doMigrateAll" value="false" />
<add key="startingVersion" value="[value]" />
<add key="maxVersion" value="[value]" />
```

The value of the startingVersion is the one that you have written down in step 1.

The value of the maxVersion is the current Synchronization version. The parameter "change" should be set to 300'000. More than that would produce a timeout error.

4. Restart migration normally

Make sure that you take back the original migration keys :

```
<add key="doMigrateProdsUpdates" value="true" />
<add key="doMigrateProds" value="true" />
<add key="doMigrateAll" value="true" />
<!--<add key="startingVersion" value="[value]" />-->
<!--<add key="maxVersion" value="[value]" />-->
```

Known errors

In case of errors, note that there is a transaction in all the steps of the migration. So you can modify the startingVersion to match the maxVersion of the last run (the information is in the log file.)

Timeout error

This happens if the database was not able to treat all the operations. To reduce the number of operations, you can simply reduce the change key in the config file.

Foreign key error

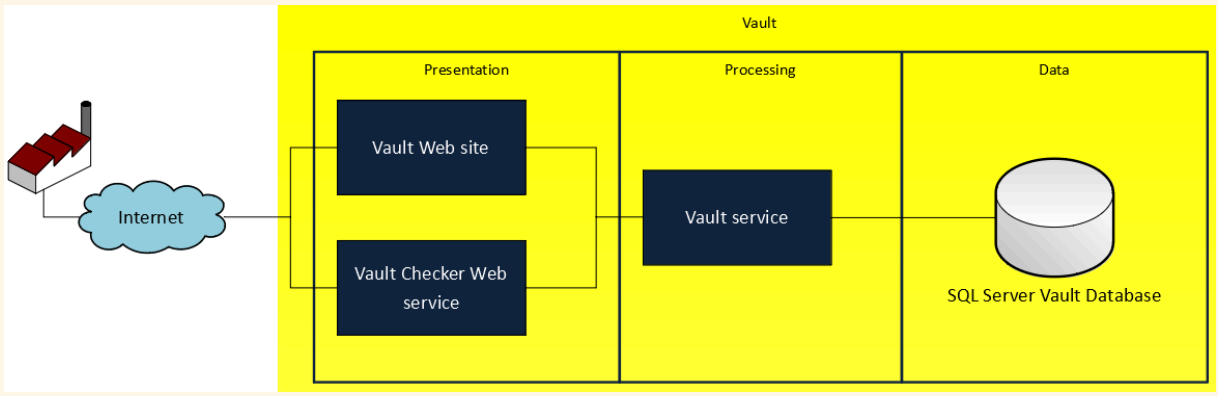
It has been noticed once that a PROD was not imported even though it should have been. In those cases, you may reimport only the PRODs.
To do that, change the config file keys to

```
<add key="doMigrateProdsUpdates" value="false" />
<add key="doMigrateProds" value="true" />
<add key="doMigrateAll" value="false" />
```

and you can set the change setting to 10'000'000.

You need to keep the startingVersion and maxVersion. There is a test to prevent any duplication, so this will not impact the migration.

Vault Procedures: Contingency Procedure



Vault Components

The four components could be deployed on one physical or on four separate servers or any combination (i.e. on 1, 2, 3 or 4 servers).

Note: the Processing component in the figure is referenced as 'B' (business) in the table below.

The presentation layer supports using load balancing techniques. On the data layer, SQL Server mirroring is supported. The following options are described in this section:

Deployment option	Components	Description
1 server	1 x (P + B + D)	1 x (Presentation, business and data deployed in one server)
2 server	2 x (P + B + D)	2 x (Presentation, business and data deployed in one server)
3 server	2 x (P + B + D) W	2 x (Presentation, business and data deployed in one server) 1 x (SQL Server Witness)
5 server	2 x (P + B) 2 x D W	2 x (Presentation and business deployed in one server) 2 x (Data are deployed in one server) 1 x (SQL Server Witness)
7 server	2 x P 2 x B 2 x D W	2 x (Presentation deployed in one server) 2 x (Business deployed in one server) 2 x (Data are deployed in one server) 1 x (SQL Server Witness)

Note: Component P always has a dedicated Component B, hence they work in a pair.

If you have a two site solution, and the disaster has the effect that the disaster site becomes active, you should consider switching back to the normal data site after recovery. You should then execute the following operation to switch between the sites:

- Perform failover between sites, see below.

In this section only option "2 servers" is described for failover between sites.

Hardware failure but hard disk is intact

- If the Vault Server has a hardware failure but the hard disk is intact, then try to repair the hardware.
- If the hardware cannot be repaired, then do one of the following two things:
  - In case of Single Server solution:
    - Follow description in section 2 One-Server solution disaster of Vault Procedures: Disaster Recovery
    - Copy the folder C:\VaultVXXXx from the hard disk on the old Server to the new Server, where xxxx is the version number.
    - If the Configuration Protection Tool has been used to encrypt data, e.g. the database connection string, then this must be done again to work on the new server (see Configuration Protection Tool - Encryption Data).
    - Restart the Vault Service and/or IIS.
  - In case of Multiple Server solution:
    - Follow the guidance in section 3 Multiple-Server solution disaster of Vault Procedures: Disaster Recovery
    - Copy the folder C:\VaultVXXXx from the hard disk on the old Server to the new Server, where xxxx is the version number.
    - If the Configuration Protection Tool has been used to encrypt data, e.g. the database connection string, then this must be done again to work on the new server (see InextorToolsGuide section Configuration Protection Tool - Encryption Data).
    - Restart the Vault Service and/or IIS.

Hard disk failure

See page Vault Procedures: Disaster Recovery.

Maintenance

Failover for maintenance on server

If one-server solution:

- Stop the Vault Service (see below).
- // Execute maintenance
- Start the Vault Service (see below).

If multiple-server solution:

- If maintenance on primary Vault:
  - Failover to secondary Vault (see below)
  - // Execute maintenance
  - Failover to the original principal Vault (see below)
- If maintenance on secondary Vault:
  - // Execute maintenance

Starting/Stopping Vault Service

- For Windows start button, select Control Panel, Administrative tools, Services
  - Double click on Inextor Vault 10.1.0.xxxx, where xxxx is the version number
  - Select General tab and:
    - Click Stop button for stopping the service
    - Click Start button for stopping the service



Vault Starting Stopping Web service

Failover to Secondary Vault

Common section

Make sure to perform the following on the Secondary Business Vault Server:

- From the "Control Panel", start "Administrative Tools" and "Services".

Start the Vault Service, and check that it is set to automatic start.

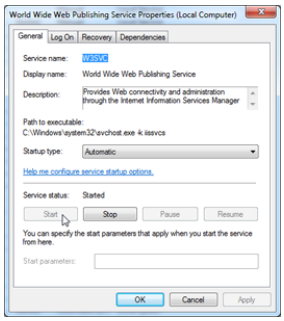


Automatic start

Make sure to perform the following on the secondary Presentation Vault:

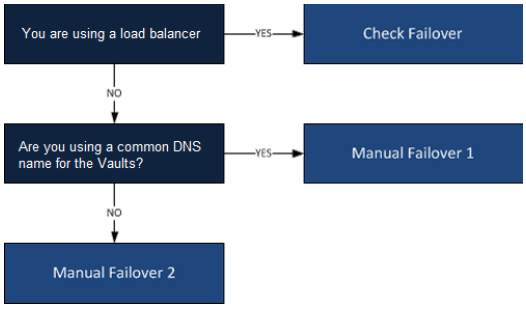
- From the "control panel", start "administrative tools" and "services".

Start "World Wide Web Publishing Service" (and set it to automatic start)



Automatic start

Follow the sequence diagram for your setup:



Sequence Diagram

Check Failover

Ping the secondary Presentation (P) Vault:

- Start the command prompt (cmd.exe)
- Issue the command "ping [SecondaryVaultAddress]"
  - Take notice of the Reply from: XXX.XXX.XXX.XXX
- Issue the command "ping [CommonVaultAddress]"
  - Take notice of the Reply from: XXX.XXX.XXX.XXX
- Check that both replies are equal

Manual Failover 1

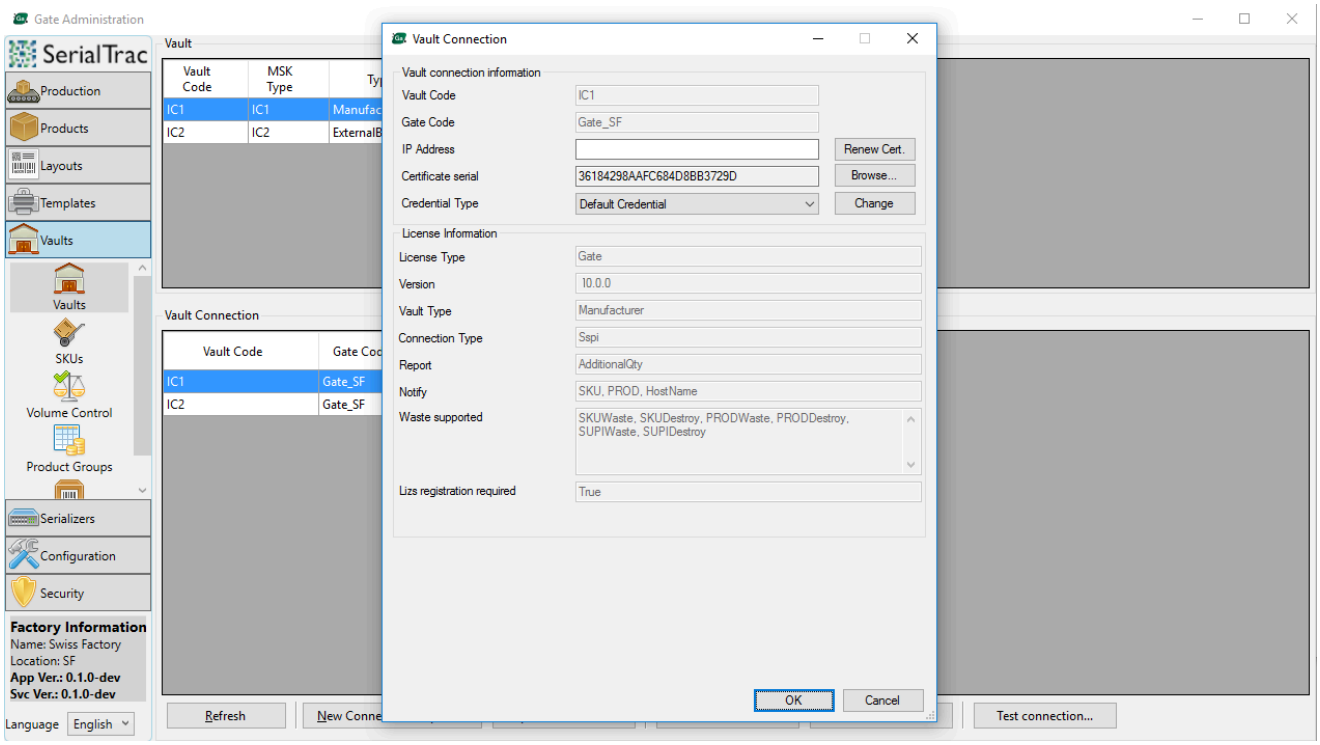
On the DNS-server, set the common address to point on the secondary server.

Check the failover, see above.

Manual Failover 2

- The Vault connection(s) of the Gate(s) has/have to be changed to reflect the new Vault address:

- On every Gate connected to the Vault, open Gate Administration GUI and select the Vault tab and click the Edit Connection button



Edit Connection

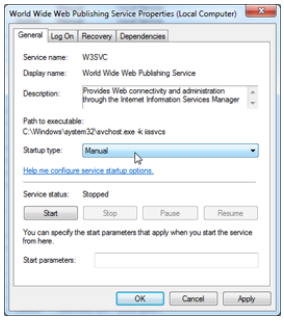
  - Enter the correct host address for the secondary Vault (and add /WS).
  - Verify that Status changes to Authorized
- Connections made to the Vault web site will have to use the new Vault address.
  - For instance if the Primary Vault has the address vault1.xxx.local/web and the other one vault2.xxx.local/web , you should enter the following in Internet Explorer browser: https:// vault2.xxx.local/web

Failing over between sites

If the Vaults are deployed on two sites and you want to failover from one site to the other, and put one site in standby, execute the following tasks

- On the Presentation (P) Vault that is to be put into standby:
  - From the "control panel", start "administrative tools" and "services".

Stop "World Wide Web Publishing Service" (and set it to manual start)



Stop World Wide Web Publishing Service

- On the Business (B) Vault that is to be put into standby:
  - From the "control panel", start "administrative tools" and "services".

Stop Vault Service, and check that it is set to manual start.



Manual Start

- On the Data (D) Server that is to be put into standby:
  - Execute a controlled failover to the other SQL Servers, See section 5.2.2

Execute the failover described above.

Vault Procedures: Azure Managed Identities

Context

In order to avoid using SQL user and password, we would like that our system deployed in Azure use Azure Managed Identity.

Documentation about Managed Identities in VM can be found here : How managed identities for Azure resources work with with Azure virtual machines | Microsoft Docs

A tutorial about the implementation can be fond here : Tutorial: Use a managed identity to access Azure SQL Database - Windows - Azure AD | Microsoft Docs

Implementation

There is a setting to enable the functionality *AzureManagedIdentities*. If set to false, then the connection works as before

Upon opening a connection to the DB, there is a call to the a new class *AzureConnectionManager*. The goal of the *AzureConnectionManager* is to provide the token to the connection.

The manager will store the token for duration of 23 hours (over a validity of 24 hours) and provide it to all the connections that opens. At each connection it checks whether the token needs to be refreshed (i.e. 23 hours passed since the token arrived) and make a request for a new one. The token is stored in memory. In case of a restart, then a new request is sent, as there is no token available anymore.

A request typically looks like this

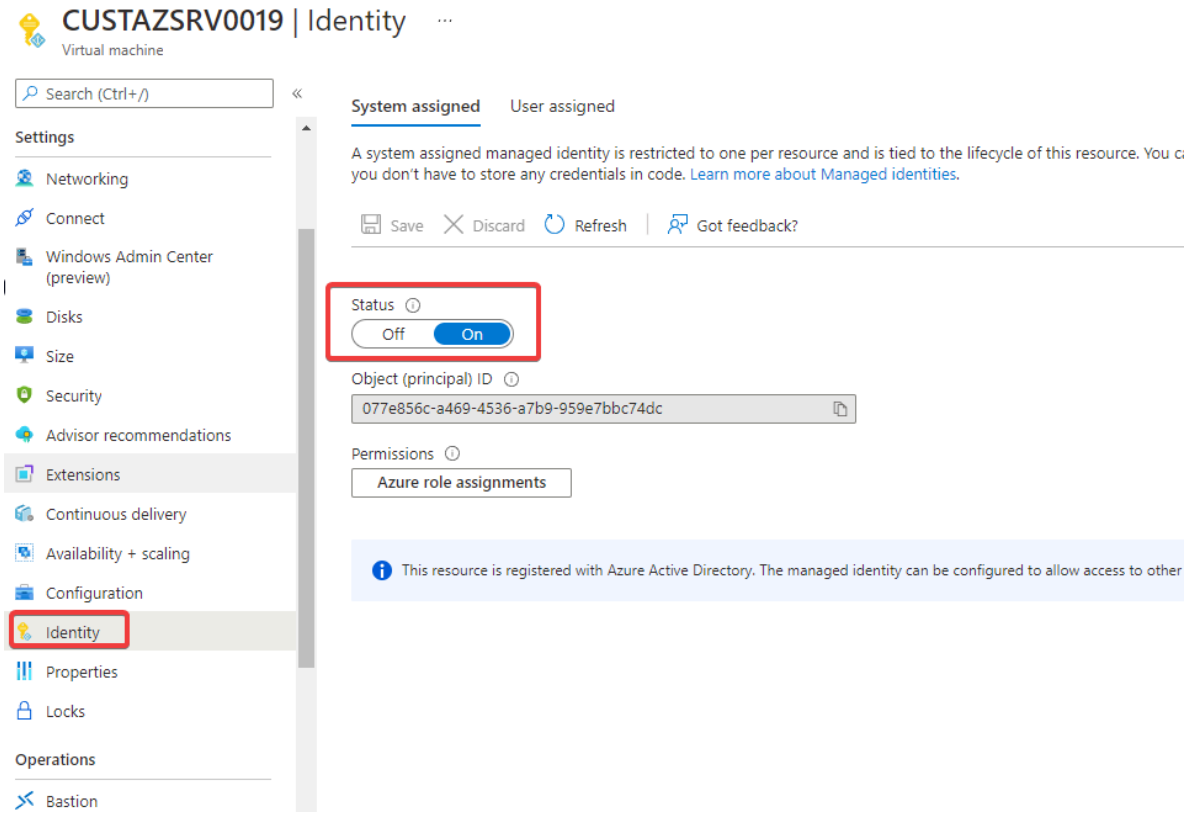
```
GET http://169.254.169.254/metadata/identity/oauth2/token?api-version=2018-02-01&amp;resource=https://database.windows.net/
```

While a response is usually of the following form

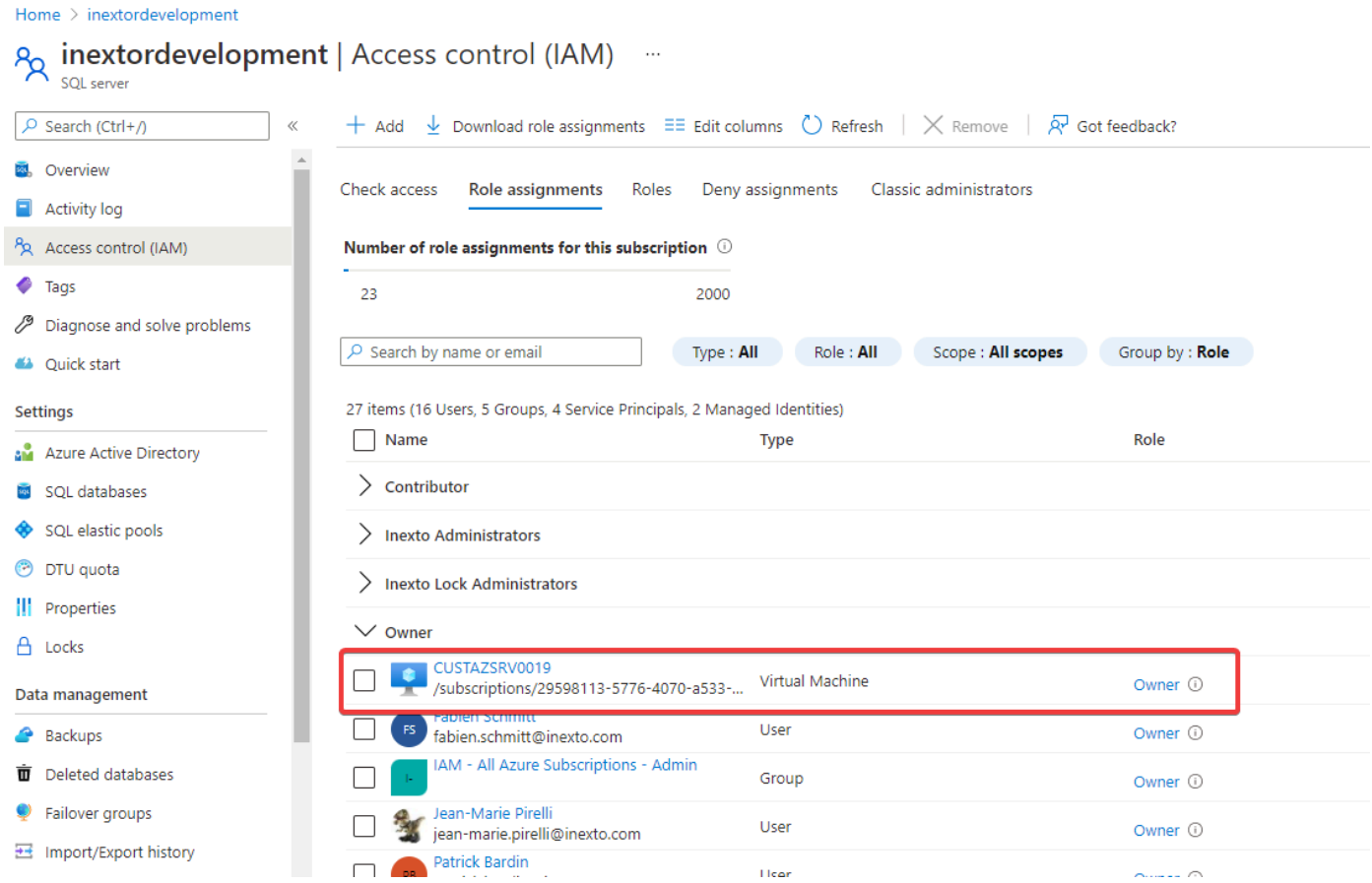
```
{
    "access_token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsIng1dCI6Imwzc1EtNTBjQ0g0eEJWWkxIVEd3blNSNzY4MCIsImtpZCI6Imwzc1EtNTBjQ0g
    "client_id": "d761cdfa-be97-4ac4-ab2e-4a07ae110900",
    "expires_in": "85805",
    "expires_on": "1635436455",
    "ext_expires_in": "86399",
    "not_before": "1635349755",
    "resource": "https://database.windows.net/",
    "token_type": "Bearer"
}
```

Setup Step 1 (for Azure Portal Administrators)

Make sure that the VM has the Managed Identities enabled.



On the SQL server page, you want to check that the VM does have the ownership of the server



If not, you want to add it, for that, you may follow the steps defined here. Make sure to user the owner role as well as the Managed Identity option.

Setup Step 2 (for Database Administrators)

In order to make the VM be able to to access the db, we need to add it to the Users using the following command

```
CREATE USER [VMName] FROM EXTERNAL PROVIDER
```

Then we need to define the role of that user, such as

```
ALTER ROLE db_owner ADD MEMBER [VMName]
```

Setup Step 3 (for Windows Administrators)

Changes vault.exe.config file:

- Add the configuration parameter *AzureManagedIdentities* and set it to true. It may not be present in the config file by default, you will need to add it if it is the case.
- Change the connection string to a string like "Data Source=inextordevelopment.database.windows.net;Initial Catalog=VaultDev;MultipleActiveResultSets=True;Persist Security Info=True"

In the end, the app settings will contain the following values

```
<appSettings file="AppSettings.config">
...
    <add key="AzureManagedIdentities" value="True" />
    <add key="DBConnectionString" value="Data Source=inextordevelopment.database.windows.net;Initial Catalog=VaultDev;MultipleAc
...
</appSettings>
```

Vault Procedures: Configuration

This section gives configuration information for various installation options.

Authentication Methods

The page Authentication Methods describes the various authentication modes used to connect to the Vault website and web services.

Security

The page Security (Configuration) gives additional information on security aspects, including data protection.

Migration

The page Migration (Configuration) gives information related to the Vault configuration settings in case of a migration to a newer version of the Gate.

Configuration Files

The page Configuration Files (Vault Configuration) lists and describes the content of all configuration files related to the Vault service, website and web services.

Configure Vault to use NetNamedPipeBinding

The page Configure Vault to use NetNamedPipeBinding (Configuration) gives instructions to configure the Vault to use NetNamedPipeBinding rather than NetTcpBinding for performance reasons.

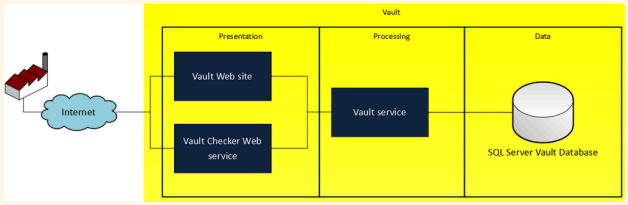
Exposing Vault Web Services

The page Exposing Vault Web Services describes the procedure to follow in order to expose web applications to external systems or users from outside the domain on a Vault that uses Windows authentication.

Configuring Web Clients with Limiting Access

The page Configuring Web Clients with Limiting Access describes the procedure to follow to configure the web clients to be able to connect to the Vault service when the Vault service is set up for limiting access.

Vault Procedures: Disaster Recovery



Vault Components

The four components could be deployed on one physical or on four separate servers or any combination (i.e. on 1, 2, 3 or 4 servers).

Note that the Processing component in the figure is referenced as 'B' (business) in the table below.

The presentation layer supports using load balancing techniques. On the data layer, SQL Server mirroring is supported. The following options are described in this section:

Deployment option	Components	Description
1 server	1 x (P + B + D)	1 x (Presentation, business and data deployed in one server)
2 server	2 x (P + B + D)	2 x (Presentation, business and data deployed in one server)
3 server	2 x (P + B + D) W	2 x (Presentation, business and data deployed in one server) 1 x (SQL Server Witness)
5 server	2 x (P + B) 2 x D W	2 x (Presentation and business deployed in one server) 2 x (Data are deployed in one server) 1 x (SQL Server Witness)
7 server	2 x P 2 x B 2 x D W	2 x (Presentation deployed in one server) 2 x (Business deployed in one server) 2 x (Data are deployed in one server) 1 x (SQL Server Witness)

Component P always has a dedicated Component B, hence they work in a pair.

If you have a two-site solution, and the disaster has the effect that the disaster site becomes active, you should consider switching back to the normal data site after recovery. You should then execute the following operation to switch between the sites:

- Perform failover between sites, see section 5 of Contingency Procedures.

In this section only option “2 servers” is described for failover between sites.

There should exist backup on items required for performing disaster recovery. The document Administration Performance Monitoring Guide describes what’s needed to keep back up on.

”Lost” in this section refers to the fact that the server is no longer functional and that a spare server must replace the lost server and that the database cannot be recovered from the hard disk.

When it is stated in this document that an action might result in data loss, the consequences of vital data loss must be managed, that is:

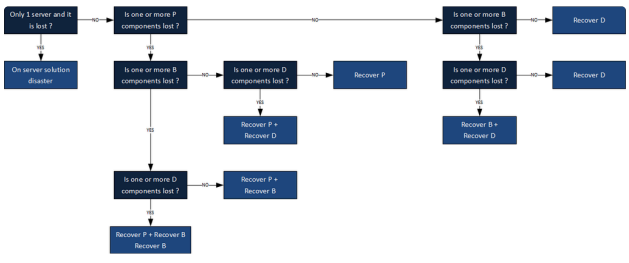
- Production batch information:
  - There exists no mechanism that will force manufacturers to resend lost production batch information.
- Waste and Export:
  - Waste and Export volumes declared that have been lost must be declared once more on the Gate.

Note: When using data protection (e.g. for securing certificates in the database or for encrypting the connection string to the SQL server in the Vault configuration file), you might have to create new data protection (however not creating new global or specific keys), see Certificate Management. This is only valid if you have to replace an existing server with a new one.

Disaster, overview

In this section the letters P, B and D represent the three layers “Presentation”, “Business” and “Data” respectively.

In all cases the Vault certificates should be reused.



Disaster Recovery

One-Server solution disaster

- Prepare Server, see Setup the Vault, section Prepare Server.
- Recover database from backup, see Restore.
- Prepare and install according to Setup the Vault (see section Installation sequence) for the following components.
  - Presentation Component
  - Business (Process) Component

Multiple-Server solution disaster

Recover P

- If there exists a running P Server, and it is not primary, then fail over to this server (see section 4 of Contingency Procedures)
- Prepare and install the P Server(s) that are lost according to Inextor Vault Installation Guide (see section Installation sequence) for the following components:
  - Presentation Component
- Failback to original P Server (If the current active server is on disaster site). See section 4 Failover to Secondary Vault of Contingency Procedures.

Recover B

- If there exists a running B Server, and it is not primary, then fail over to this server (see section 4 of Contingency Procedures)
- Prepare and install the B Server(s) that are lost according to Inextor Vault Installation Guide (see section Installation sequence) for the following components:
-
  - Business (Process) Component
- Failback to original server (If the current active server is on disaster site). See section 4 Failover to Secondary Vault of Contingency Procedures.

Recover D

- If there exists a running D Server, then check that the database here is Principal (see Mirroring States) If not:
  - Disable the witness for mirroring, see Database mirroring (if having a W server)
  - Perform a Forced SQL failover for the database, see Database mirroring: Forced manual failover
- Prepare and install according to Setup the Vault (see section Installation sequence) for the following components:
  - Data Component (but do not install the Vault database)
- If both D Servers were lost, then recover database from backup, see page Restore.
- Reestablish mirroring, see Database mirroring
- Enable witness; see Database mirroring (if having a W server).

Recover W

- Disable the witness for mirroring, see Database mirroring
- Prepare according to Setup the Vault for the component:
  - Data Component, mirroring
- Reestablish witness, see Database mirroring.
- Enable witness; see Database mirroring.

Recovering when the RSA keys are lost

In case you have lost the global public key and the specific key pair for the Vault, you have the following options:

- Use the ConfigurationProtectionTool to decrypt the RSA keys found in the VaultService.exe.config file. **This is only possible if the same user and the same installation of the operating system are used**. See the section about Decrypting Data in Certificate Management.
- If one of the servers in a multi-server configuration has crashed, the keys could be retrieved from one of the other servers that use the same Vault database. See the section about Decrypting Data in Certificate Management.
- If none of this works the certificate must be encrypted again using new keys. Normally, the only certificate in the database that includes a private key is the Vault certificate. Hence, if you have a backup copy of the Vault certificate, execute the following steps:

- Remove the current Vault certificate from the database. See the section about Deleting certificates from the database in Certificate Management.
- Create a new specific key pair (you also need the original global public key. If the global key is lost and cannot be obtained from the central organization, all Vaults will need to be updated with a new set of global keys). See the section about Create Specific key pair in Certificate Management.
- Import the Vault certificate into the database. See the section about Import certificates into Vault database in Certificate Management.
- Create protected configuration for the Vault service. See the section about Create keys for Vault services (Protect configuration) in Certificate Management.
- Restart the Vault Windows Service

- If you do not have a copy of the Vault certificate, follow these steps:
  - Create a new specific key pair. See the section about Create specific key pair in Certificate Management.
  - Create a protected configuration for the Vault service. See the section about Create keys for Vault services (Protect configuration) in Certificate Management.
  - Deploy the global private key file (GlobaRSAKeyWithPrivateKey.txt) in the same folder as the Vault NT service is deployed in.
  - Rename the file to “globalkey.xml”
  - Restart the Vault Windows Service. The service will decrypt the certificate using the private key of the global key and then encrypt it with the new keys.
  - Remove “globalkey.xml”.

Vault Procedures: Setup Vault

Vault components and installation sequence

An overview of the Vault components and the Vault installation for each scenario (one-server vs multi-server installation; new installation, upgrade or disaster recovery). Start here to find the sequence of steps to follow for each scenario.

Table of Contents

- General
- Vault Components
- Installation Sequence
- - One Server Option
  - Multi-Server Option

General

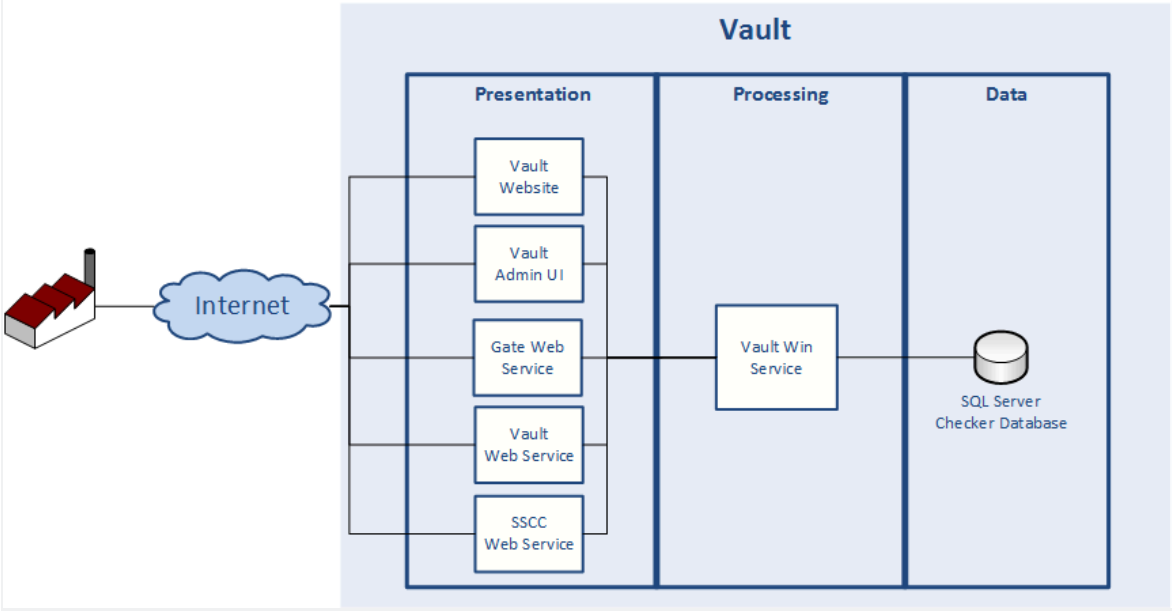
It is recommended that you note every step and every setting made on the servers during the installation, and that you use these notes to create a document that defines all settings made. This document should be at hand when later executing an upgrade and/or disaster recovery.

Important: Before starting one of the installation procedures listed below, refer to the Prerequisites Chapter, to the Technical Overview, to the Manufacturers Checklist and to the Hardware and Software Requirements.

Vault Components

A Vault consists of seven components:

- A Web Site, which runs under IIS in ASP.NET framework responding to human user interaction.
- An Administration Application to monitor messages sent by Gates and their processing.
- A Web Service, which runs under IIS in ASP.NET framework responding to Factory requests.
- A Web Service, which runs under IIS in ASP.NET framework responding to external systems requests for serialization and deserialization of codes.
- A Web Service, which runs under IIS in ASP.NET framework responding to external systems requests for allocation of SSCC ranges.
- An executable program, which runs as a Windows service responding to requests from the Web site and Web Services.
- A SQL Server database which stores data coming from the Windows service and provides data on request from the Windows service.



Vault Components

Note: the Processing part in the figure above as referenced as 'B' (business) below.

The components could be deployed on one server or on four separate servers or any combination (i.e., on 1, 2, 3 or 4 servers). Before installation, the deployment option that is to be used should be known.

This installation guide covers the following deployment options:

Deployment option	Components	Link
1 server	1 x (P + B + D)	1 x (Presentation, business and data deployed in one server)
2 server	2 x (P + B + D)	2 x (Presentation, business and data deployed in one server)
3 server	2 x (P + B + D) W	2 x (Presentation, business and data deployed in one server) 1 x (SQL Server Witness)
5 server	2 x (P + B) 2 x D W	2 x (Presentation and business deployed in one server) 2 x (Data are deployed in one server) 1 x (SQL Server Witness)
7 server	2 x P 2 x B 2 x D W	2 x (Presentation deployed in one server) 2 x (Business deployed in one server) 2 x (Data are deployed in one server) 1 x (SQL Server Witness)

ⓘ Note: Component P always has a dedicated Component B, hence they work in a pair. For gaining high available solution, NLB is supported for the presentation layer and SQL Server mirroring is supported in the data layer.

Installation Sequence

It is very important that the person performing the installation understands:

- Which deployment option to use
- Which type of installation to perform:
  - New Installation
  - Upgrade
  - Disaster recovery

This section describes all required **activities** in each case and in which **order** the activities should be performed. Only the sequence listed in the particular table that is relevant for you should be used.

One Server Option

New Installation

Action	Link
Prepare Server	Prerequisites and Required Firewall Settings (Vault)
Vault database installation	Database Setup
Vault service installation	Vault NT Service installation
Web applications installation	Vault Web applications installation
Enable SSL	Enable SSL
IIS Configuration	IIS configuration
Vault Admin UI	Vault Administration UI installation
Finalize installation	Verify Installation
Verify installation	Verify Installation and Vault Installation Problems

Upgrade

Action	Link
Vault database upgrade	Database Setup - Upgrade scripts
Vault service installation	Vault NT Service installation
Web applications installation	Vault Web applications installation
Vault Admin UI	Vault Administration UI installation
Verify installation	Verify Installation and Vault Installation Problems

Disaster Recovery

Action	Link
-	(Instructions in Disaster Recovery)
Prepare server	Prerequisites and Required Firewall Settings (Vault)
Vault service installation	Vault NT Service installation
Web applications installation	Vault Web applications installation
Import Certificate from backup	Import Certificate from Backup
IIS Configuration	IIS configuration
Vault Admin UI	Vault Administration UI installation
Verify installation	Verify Installation and Vault Installation Problems
-	(Instructions in Disaster Recovery)

Multi-Server option

New Installation

Action	Link
Prepare servers (all)	Prerequisites and Required Firewall Settings (Vault)
Vault Database Installation	Database Setup (primary server)
Setup Mirroring and Witness	Database Setup - SQL Mirroring
Vault service installation	Vault NT Service installation (both)
Web applications installation	Vault Web applications installation (both)
Enable SSL (primary server)	Enable SSL
Enable SSL on secondary server (export and import IIS certificate)	Enable SSL on Multiple Servers
IIS Configuration	IIS configuration (both)
Vault Admin UI	Vault Administration UI installation
Finalize installation	Verify Installation
Verify installation	Verify Installation and Vault Installation Problems

Upgrade

Action	Link
Vault database upgrade (primary server)	Database Setup - Upgrade scripts
Vault service installation	Vault NT Service installation (both)
Web applications installation	Vault Web applications installation (both)
Vault Admin UI	Vault Administration UI installation
Verify installation	Verify Installation and Vault Installation Problems (both)

Disaster Recovery (Vaults)

Action	Link
-	(Instructions in Disaster Recovery)
Prepare server	Prerequisites and Required Firewall Settings (Vault)
Vault service installation	Vault NT Service installation (if recovering this component)
Web applications installation	Vault Web applications installation (if recovering this component)
Import Certificate (if recovering Web application)	Import Certificate
IIS Configuration	IIS configuration (if recovering Web applications)
Vault Admin UI	Vault Administration UI installation
-	(Instructions in Disaster Recovery)

Disaster Recovery (witness server)

Action	Link
-	(Instructions in Disaster Recovery)
Prepare server (witness)	Prerequisites and Required Firewall Settings (Vault)
-	(Instructions in Disaster Recovery)

General Installation Procedures: Gate Procedures

The following sections contain all necessary information for the setup and configuration of a new Gate.

Introduction

This section introduces the system, defines the target audience, and points the reader to appropriate sections for more information.

Installation Prerequisites

This section gives the list of all technical prerequisites that must be fulfilled before installing the Gate.

Gate Setup

This section describes how to install a new Gate.

Gate Configuration

This section gives information on how to configure a Gate.

Disaster Recovery

This section details disaster recovery procedures in order to recover the Gate.

Contingency Procedures

This section details contingency procedures of the Gate.

Troubleshooting

This section gives instructions for troubleshooting the Gate.

Gate Installation Procedures: Introduction

Overview

The purpose of this document is to describe the installation, upgrade and disaster recovery procedures for the Inextor v10 Gate (Gate). It provides IT personnel in charge of the installation with step-by-step instructions to complete the installation.

- For an overview of the technical prerequisites that must be fulfilled prior to installation, refer to the Prerequisites chapter.
- For step-by-step installation instructions, refer to the Setup chapter.
- For additional configuration information, refer to the Configuration chapter.

Audience

This document is intended for the IT team in charge of installing or upgrading the Inextor v10 Gate.

More information

For further information on business concepts and scenarios, refer to the INEXTOR Operational Guide.

Gate Installation Procedures: Configuration

Finalize the Installation

At the end of the installation, the Inextor Gate is installed and the Windows Service is running, the access rights have been granted according to your environment. In order to finalize the installation and have a Gate fully operational, the following procedures must be executed:

- Register the Gate you have just installed against a Vault; for this, follow the procedure described in INEXTOR Operational Guide - Register Gate.
- Create a Linkup Setup using the files with the extension.luc that have been provided for your factory; for this, follow the procedure described in INEXTOR Operational Guide - Create LinkUp setup.

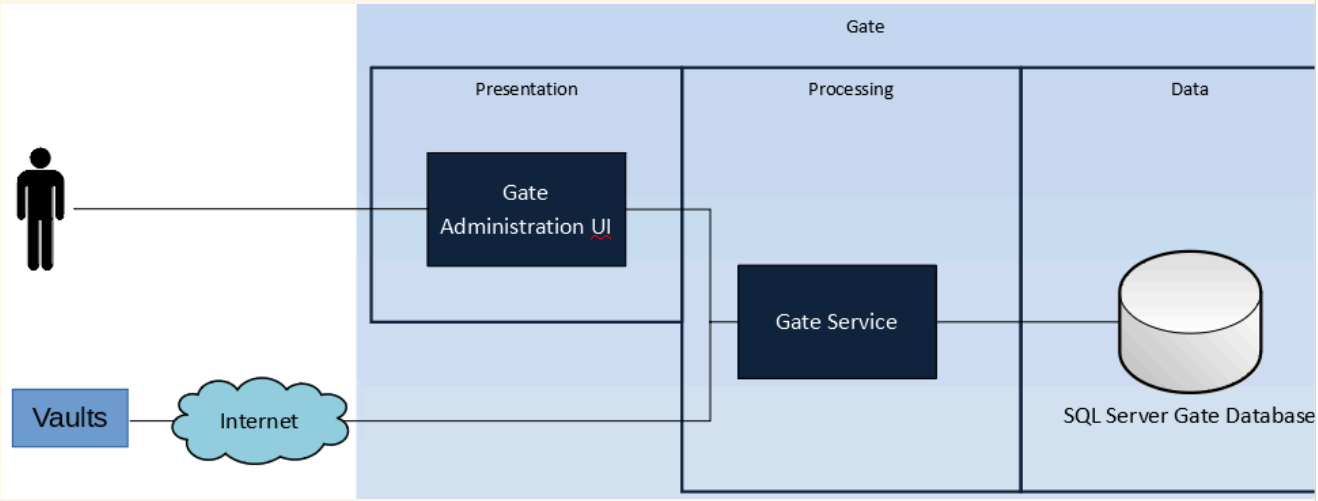
Configure the Gate Service

After the Gate installation, you can configure the Gate Service by editing the associated configuration files.

Configure the Operation Drivers

To view and modify the configuration settings for the External Service and Automation Service operation drivers, see WCF Interface and Operation Drivers.

Gate Installation Procedures: Disaster Recovery Gate



Gate Components

The processing and data components could be deployed on one or two separate servers. The processing layer can be deployed with two equally configured servers gaining high availability for the data layer, SQL Server mirroring is supported.

The following deployment options are described in this section:

Deployment option	Components	Description
1 server	1 x (P + B + D)	1 x (Presentation, business and data deployed in one server)
3 server	2 x (P + B + D) W	2 x (Presentation, business and data deployed in one server) 1 x (SQL Server Witness)

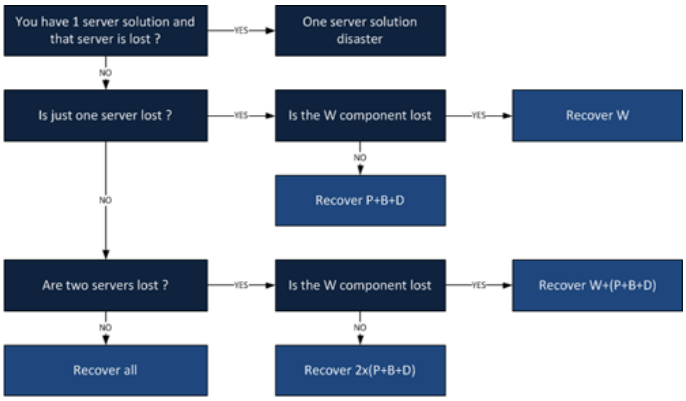
There should exist backup on items required for performing disaster recovery. The document Administration Performance Monitoring Guide describes what needs to be backed up and how often.

”Lost” in this section means that the server is no longer functional and that a spare server must replace the lost server and that the database cannot be recovered from the hard disk.

Wherever stated in this section that an action might result in data loss, and that this might result in data loss on the Vaults, this must be managed, that is:

- Production batch information:
  - There exists a synchronization mechanism between the Lizs and the Vault, which should take care of production batch information that has been lost. You should check that data that has been recovered actually is sent to Vaults. Depending on rules stipulated by the Vault owner, it might be required to report any data loss.
- Waste and Export:
  - Waste and Export volumes declared that have been lost must be declared once more on the Gate since otherwise it will not be reported on the Vault.

In all the following cases the Gate license can be reused. It will not be lost since it is in the database, and the database can be recovered from backup.



Recovering Procedure

One server solution disaster

- Prepare Gate Server, see Setup the Gate
- Recover database from backup, see the SQL Server Summary: Restore page.
- Prepare and install according to Setup the Gate (see section Installation sequence).
  - Note 1: Since database is recovered from backup, you should not create a new database.
  - Note 2: You should use the manifest file (*.uni) that does not include the scripts for populating the database with tables and default data since you already have the database recovered from backup.

Recover W

- Disable the witness for mirroring, see Database Contigency Plans: Database mirroring
- Prepare the witness server according to Setup the Gate section Prepare Gate Server.
- Reestablish witness, see Database Contigency Plans: Database mirroring.
- Enable witness; see Database Contigency Plans: Database mirroring.

Recover P+B+D

- If the primary Gate has failed:
  - If using the Gate WCF Interface, make sure that the WCF clients are redirected to the secondary server, see Checking that Gate WCF Clients are using the correct Gate WCF Interface.
- Check that the database that is functional is Principal (see Mirroring States). If not:
  - Disable the witness for mirroring, see Database Contigency Plans: Database mirroring
  - Perform a Forced SQL failover for the database, see Database Contigency Plans: Database mirroring.
- Prepare and install according to Setup the Gate (see section Installation sequence).
  - Note 1: You should not create a new database.
  - Note 2: You should use the manifest file (*.uni) that does not include the scripts for populating the database with tables and default data.
- If the primary Gate has been recovered:
  - If using the Gate WCF Interface, make sure that the WCF clients are redirected to the primary server, see Checking that Gate WCF Clients are using the correct Gate WCF Interface.
- Reestablish mirroring, see Database Contigency Plans: Database mirroring.
- Enable witness; see Database Contigency Plans: Database mirroring.

Recover W+(P+B+D)

- Follow Recover W.
- Follow Recover P+B+D.

Recover 2x(P+B+D)

- On the primary Gate:
  - Prepare Gate Server, see Setup the Gate
  - Recover database from backup, see SQL Server Summary: Restore.
  - Prepare and install according to Setup the Gate
    - Note 1: You should not create a new database.
    - Note 2: You should use the manifest file (*.uni) that does not include the scripts for populating the database with tables and default data.
- On the secondary Gate:
  - Prepare Gate Server, see Setup the Gate
  - Prepare and install according to Setup the Gate (see section Installation sequence).
    - Note 1: You should not create a new database.
    - Note 2: You should use the manifest file (*.uni) that does not include the scripts for populating the database with tables and default data.
- Setup mirroring, see Mirroring States.

Recover All

- Follow Recover 2x(P+B+D).
- Follow Recover W.

Checking that Gate WCF Clients are using the correct Gate WCF Interface

For every client, check that they are redirected either to:

- [PrimaryGateServer].[DomainName]

or

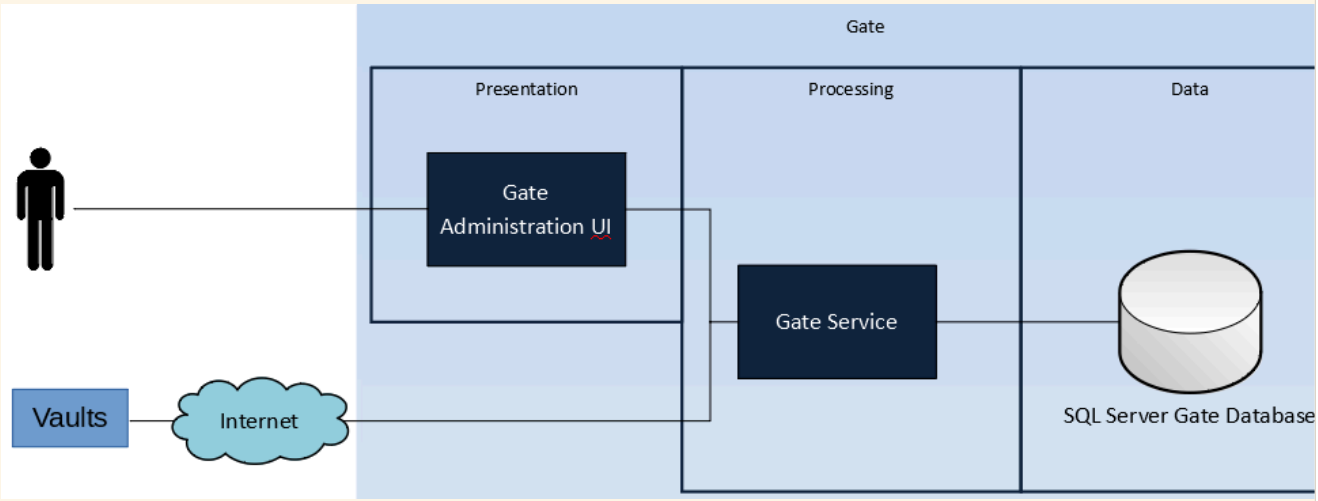
- [SecondaryGateServer].[DomainName]

If you are using a common name for the servers, then set the common address to point on the Gate server at interest in your DNS server.

Then you could test by pinging the common name and the Gate you want to be redirected to:

- Start the command prompt (cmd.exe)
- Issue the command “ping [GateXXXXAddress]”
  - Take notice of the Reply from: XXX.XXX.XXX.XXX
- Issue the command “ping [CommonGateAddress]”
  - Take notice of the Reply from: XXX.XXX.XXX.XXX
- Check that both replies are equal

Gate Installation Procedures: Contingency Procedures



Gate Components

The processing and data components could be deployed on one or two separate servers. The processing layer can be deployed with two equally configured servers gaining high availability for the data layer, SQL Server mirroring is supported. The following deployment options are described in this section:

Deployment option	Components	Description
1 server	1 x (P + B + D)	1 x (Presentation, business and data deployed in one server)
3 server	2 x (P + B + D) W	2 x (Presentation, business and data deployed in one server) 1 x (SQL Server Witness)

Hardware failure but hard disk is intact

- If the Gate Server has a hardware failure but the hard disk is intact, then try to repair the hardware.
- If the hardware cannot be repaired, then do one of the following:
  - In case of Single Server:
    - Prepare a new Gate server and install a new Gate (same version as the old) according to Setup the Gate (see section Installation Sequence).
      - Note 1: You should not create a new database.
      - Note 2: You should use the manifest file (*.uni) that does NOT include the scripts for populating the database with tables and default data, see Setup the Gate, section Using correct Manifest file (*.uni)
    - Copy the folder C:\Program Files (x86)\Inextor\Gate 10.X from the hard disk on the old Server to the new Server.
  - In case of high availability configuration:
    - Follow the guidance Recover P+B+D of Gate Disaster Recovery
    - Copy the folder C:\Program Files (x86)\Inextor\Gate 10.X from the hard disk on the old Server to the new Server.
- Restart the Gate Service.

Hard disk failure

See Gate Disaster Recovery.

Maintenance

Failover for maintenance on server

- If maintenance on the primary Gate:
  - If using the Gate WCF Interface, make sure that the WCF clients are redirected to the secondary server, see Gate Disaster Recovery.
- Stop the Gate Service on the server to be maintained.
- Execute maintenance
- Start the Gate Service on the server to be maintained.
- If the primary Gate has been maintained:

- If using the Gate WCF Interface, make sure that the WCF clients are redirected to the primary server, see of Gate Disaster Recovery.

Starting/Stopping Gate Service

- For Windows start button, select Control Panel, Administrative tools, Services
  - Double click on Inextor Gate 10.X
  - Select General tab and:
    - Click Stop button for stopping the service
    - Click Start button for stopping the service

Gate Installation Procedures: Prerequisites

Table of Contents

1. Hardware and Software Specifications
2. Licenses
3. Software Setup
4. Account Requirements
5. Installation Media

Hardware and Software Specifications

Please refer to the INEXTOR Requirements.

To install the DACPAC that contains the Gate database objects, it is necessary to use SQL Server Management Studio (SSMS) 2012 or higher.

Licenses

For each Vault with which the Gate is registered, one license file (.licx file) should be provided by the Vault owner.

Software Setup

The components will be installed using an MSI package.

Account Requirements

In order to run the Gate Service, it is advised to use a Domain Service Account. This Service Account should be granted access to the SQL database. Running the Gate Administration does not required special access rights; expect those to access the Gate Service.

When connecting Gate to Vault through a proxy installed in the factory, it is necessary for the proxy to use an account to connect to the Vault using basic http authentication. This account is created on Vault and the credentials needs to be known by the proxy to connect.

Installation Media

The following items are provided in the installation media:

Item	Type	Description
Gate Database	DAC Package	DAC Package to be deployed in order to create the Inextor Gate database.
Gate Installer	.msi file	Installation software, for new installation or upgrade
Vault license file(s)	.licx file	One license file per Vault, provided by the Vault owner.

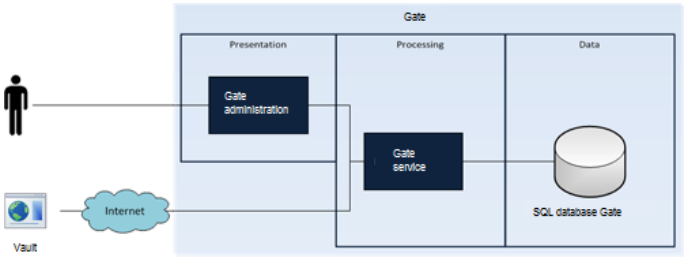
LinkUp setup	.lus file	LinkUp setup files that have been provided.

Gate Setup: Installation Sequence

Gate Components

A Gate consists of three components:

- A Gate administration GUI, which is a Windows program responding to human users interactions.
- A Service, which runs as a NT service that:
  - Responding on requests from the Gate Administration GUI(s).
  - Responding on requests from the Lizs
  - Send data and collect data from Vault(s)
- A SQL Server database which stores data coming from the NT service and provides data on request to the NT service.



Gate Components

Deployment options

The processing and data components could be deployed on one physical or virtual server or on two separate servers.

The Gate Administration GUI could be deployed on the same server as the processing server or on any other PC in the same domain.

High availability

The processing layer can be deployed with two equally configured servers gaining high availability for Lizs. On the data layer, SQL Server mirroring is supported.

Enabling NLB for Win2003/2008 Server have never been tested on Gates. SQL server mirroring is described in Database setup guide.

For Failover Clustering, please refer to Microsoft Failover Clustering

> For production environment it is recommended to use high availability (i.e. having two Gates equally configured with SQL Server mirroring using a witness for automatic failover).
>
> When having many Lizs, it could be advisable to also consider having a separate disaster site with adequate recovery plans. There are several ways to accomplish but this is out of scope for this documentation.

Interfaces

The Gate exposes several WCF Services for:

- Automating production order preparation and operation on Lizs
- Obtaining information about generated codes and productions used by the Ranger (INEXPRESS V1).

These services can be configured on the Configuration tab on the Gate Administration.

Installation Sequence

The table below gives the steps to follow in each of the following installation scenarios:

- A new installation (Gate Service and Gate Administration) on a single server
- A new installation in a client-server mode: the Gate Service and potentially the Gate Administration on the server, and the Gate Administration on any number of client machines
- An upgrade of an existing installation.

Action	New Installation - Single Server	New Installation - Client-server Mode	Upgrade
Preparation	Check the General Installation Procedures: Technical Overview, the Manufacturers Checklist, the IT Infrastructure: Hardware and Software Requirements, the Gate Prerequisites and the Required Firewall Settings		-

Database	Database Setup	Database Upgrade
Gate Installation / Upgrade	Gate Installation	
Configuration	Configuration	

General installation procedures: Liz procedures

- Liz Procedures: Introduction
- Liz Procedures: Configuration
- Liz Procedures: Disaster Recovery Liz
- Liz Procedures: Contingency Procedures Liz
- Liz Procedures: Prerequisites for Implementation of the LIZ

Liz Procedures: Introduction

Overview

The purpose of this document is to describe the installation, upgrade and disaster recovery procedures for the Inextor v10 Liz. It provides IT personnel in charge of the installation with step-by-step instructions to complete the installation.

- For an overview of the technical prerequisites that must be fulfilled prior to installation, refer to the Prerequisites chapter.
- For step-by-step installation instructions, refer to the Setup chapter.
- For additional configuration information, refer to the Configuration chapter.

Audience

This document is intended for the IT team in charge of installing or upgrading the Inextor v10 Liz.

More information

For further information on business concepts and scenarios, refer to the INEXTOR General Operational Guide.

Liz Procedures: Configuration

The following sections contain resources related to the configuration of the Liz.

Serializer Migration (From Cog v4.x to Liz v10)

This page shows how to use the migration tool provided with Inextor v10 to migrate COGs v.4.x.

Configure the Liz on the First Run

This page describes the necessary steps in order to configure the Liz on the first run prior to operation.

Backup & Restore

This page describes how to configure the backup and restore the Liz database.

Configuration Files

This page lists and describes the content of the configuration files for the Liz service and Liz operation.

Liz Procedures: Disaster Recovery Liz

If a Liz computer is in a state so that production no longer can continue, it has to be replaced by a preconfigured Liz (section 1) or reinstalled (section 2) or you need a complete new Liz installation (see the Setup the Liz page).

There should exist backup on items required for performing disaster recovery. The document Administration Performance Monitoring Guide describes what is needed to keep back up on.

If a Liz that currently is in production has a disaster, some data loss will occur. At most that could be the volume information for the current running batch. Depending on rules stated by the Vault owner, the data loss most likely has to be reported to him. Furthermore, the batch will be left as not terminated on the Vault, though it is possible to manually terminate the batch on the Vault web site, see Operation Guide.

> Note: Depending on security settings for the Vault Web site, it might be that only the Vault owner (having GOV-role) is able to force a production batch into terminated state.
>
> Note: The Liz license can only be reused when a backup can be used for recovery.

Connect a spare preconfigured Liz

By having a spare Liz PC that is preconfigured and preregistered (also the MSK is registered on the Vault(s)) then it could almost immediately be taken into production.

If you have to register the Liz and/or MSK registration then you are relying on that;

- The Liz must have connection to the Vault(s).
- The registration might require an acceptance by the Vault owner (on Vault Web site), hence in such cases could take hours/days before it is accepted.

Only Liz software is installed but Liz is not configured

If on the spare Liz PC only the Liz software is installed but not configured you then have to configure the Liz according to section Configure the Liz.

For the Linkup Setup step, you should follow the procedure in the next section in order to recover the Linkup Setup from the broken Liz.

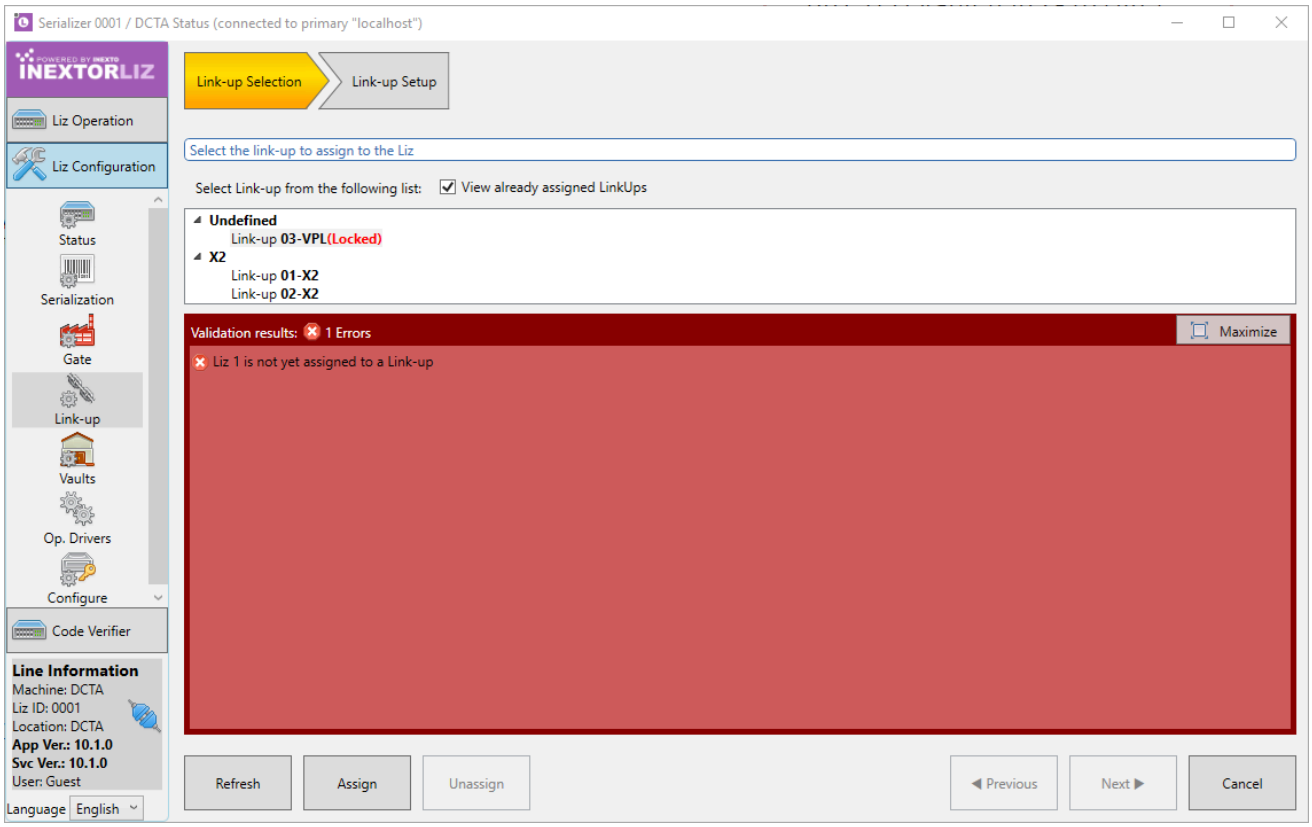
Liz software is installed and is registered (Liz and MSK) on Vault(s)

If you have a Liz preconfigured with:

- Liz ID (Liz license)
- Serialization
- Manufacturing Center ID
- MSK registered (all Vault(s))
- Configure tab (backup/restore)

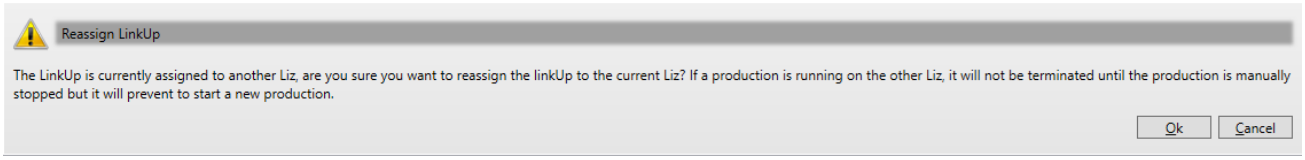
You can then follow the following instructions:

- Assign the Linkup Setup to the Spare Liz
  - On the Spare Liz PC, start the Liz UI
  - Go to Liz Configuration→Link-up
  - If the Spare Liz has a Linkup Setup already assigned, click Unassign.
  - Check the checkbox 'View already assigned LinkUps' → The assigned ones will show a red label with the text locked.
  - Select the Link-up having the Machine ID previously noted.



Liz Configuration - Linkup Setup

- Click Assign
- The following warning will be shown if the Linkup is locked:



- Click Ok

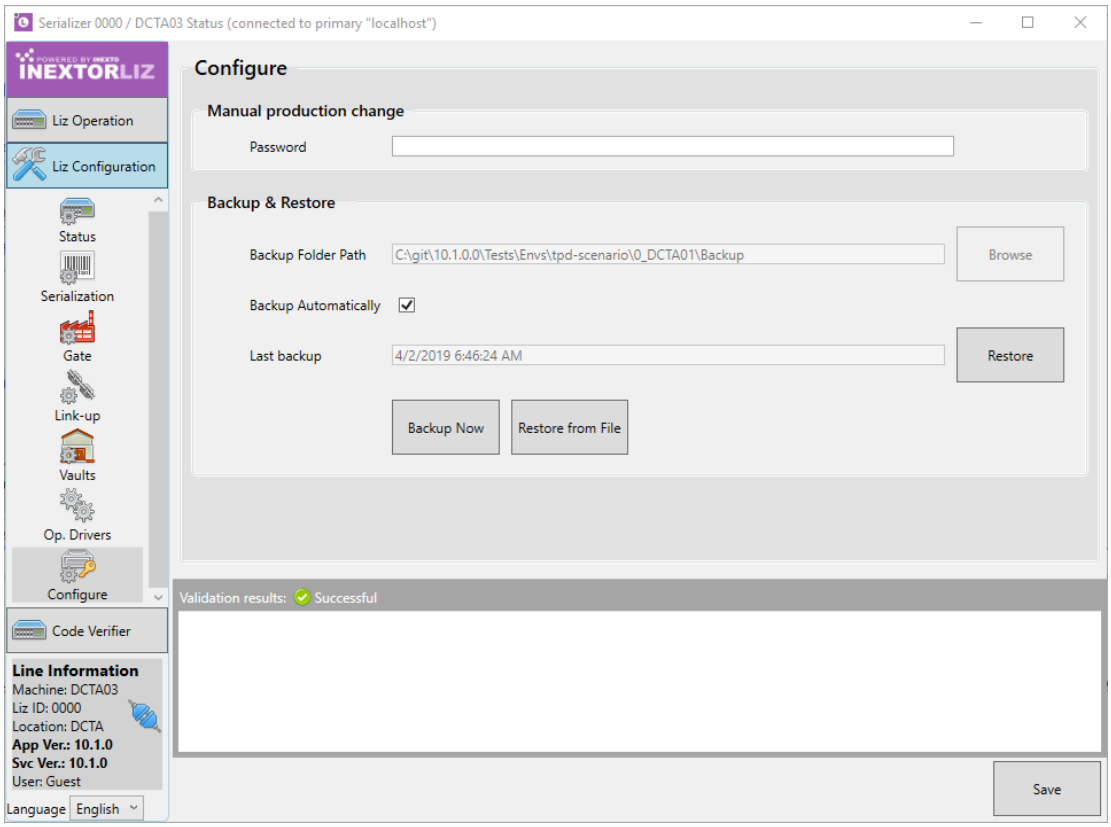
> There is currently a bug in all versions of the Liz until v10.1.0.2783 where the 2 buttons are inverted. So you need to click Cancel to assigned a locked Linkup.
>
> This issue is fixed in later versions.

Restore Liz from backup

Reinstall and recover backup

Use the following procedure to reinstall the Liz on a new computer:

- Setup the Liz PC according to Setup the Liz.
- Install the Liz software according to Setup the Liz. Depending on your setup, that could be:
  - Liz Service
  - Liz Operation
  - Language packs
- Load the latest backup file containing the database and the MSK file/files by going to the Serializer Operation GUI, choose the Configure tab and press Restore from File... and select the latest backup. It is vital that the backup being restored includes the latest configuration change registered on the Gate.

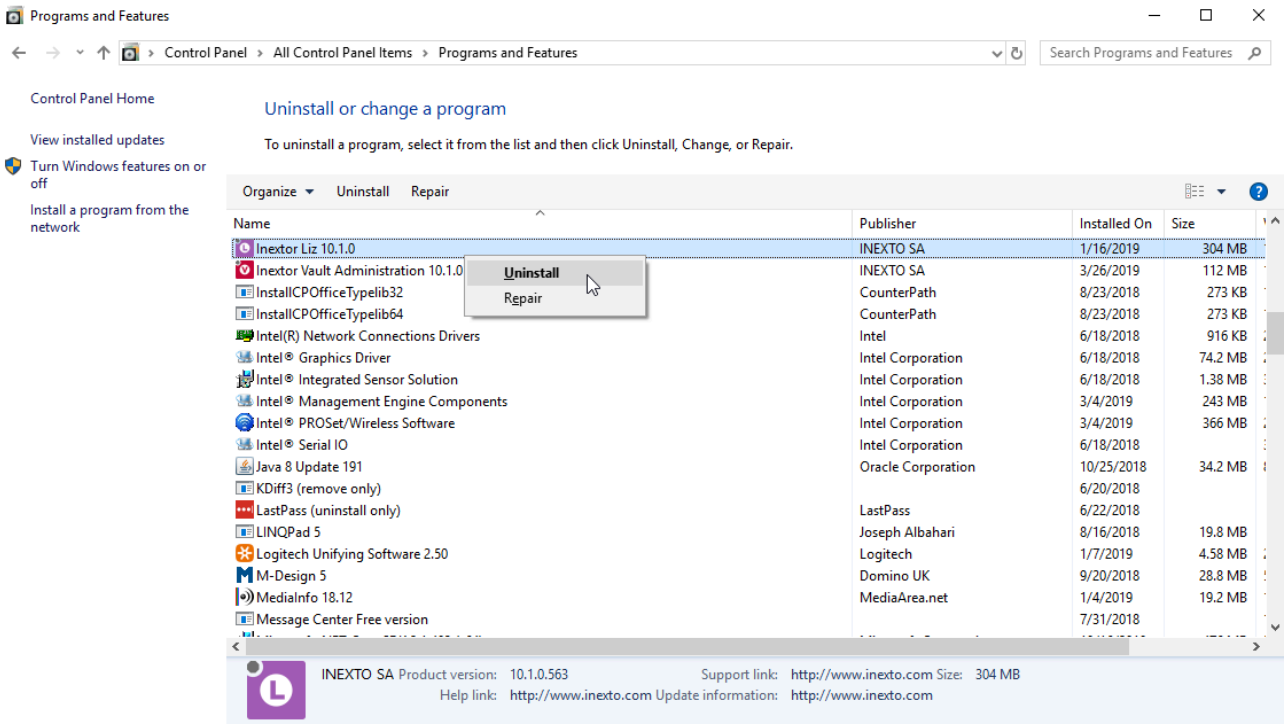


Reinstall Recover Backup

- If it did not work, the backup cannot be used. You then have to uninstall the Liz, see **Uninstall Liz** below and then install a new Liz with a new License, see Setup the Liz. Before installing a new Liz, keep trace of the following from the current Liz:
  - In Liz Operation GUI - Configuration Tab:
    - Serialization
      - Export the Scheme Configuration (Click Export)
    - Gate
      - Gate Address Primary address
      - Gate Secondary Address
      - Manufacturing Center Id (click Next)
    - Linkup
      - Machine ID
    - Operation Drivers tab
      - Operation drivers used
  - In Liz Operation GUI - Liz Operation Tab
    - For each printer icon in the dashboard click Settings
      - Keep trace of the value of each setting
- If the restore is successful, it is necessary to restart the Liz Service

Uninstall Serializer

- Uninstall the currently installed Liz.
  - From the Windows Start menu, run the Control Panel and select "Programs and Features"
  - In the list of programs, locate the program "Liz 10.x.x", right click and select "Uninstall"



Uninstall Liz

- The Liz Installer starts and uninstalls the Liz from the computer

- Install a complete new Liz according to Setup the Liz with a new Liz license file.
- When configuring the new Liz, you can restore the scheme configuration previously exported from the backup and all printer settings.

Liz Procedures: Contingency Procedures Liz

Hardware failure but hard disk is intact

- If the Liz Server has a hardware failure but the hard disk is intact, then try to repair the hardware.
- If the hardware cannot be repaired, then do one of the following two things:
    1. If a spare Liz Server is available with the Liz software installed, then can use it:
        - Verify that the version number on the spare Liz is correct (this is written in the lower right corner of the GUIs).
        - If the spare Liz is already configured, then follow the instructions in page Liz Disaster Recovery section 1. under "Liz software is installed and is registered (Liz and MSK) on Vault(s)"
        - If the spare Liz is not configured yet (only installed), then copy the folder C:\ProgramData\Inextor\Liz 10.X from the hard disk from the old Liz machine to the new spare machine.
        - Restart the Liz Service.
    2. If a spare Liz is not available, then you must prepare a new machine to do new installation:
        - Install a new Inextor Liz on the new machine according to Setup the Liz (only installation - do not configure the new Liz).
        - Copy the folder C:\ProgramData\Inextor\Liz 10.X from the old Server hard disk to the new Server.
        - Restart the Liz Service.

Hard disk failure

See section Liz Disaster Recovery.

Liz Procedures: Prerequisites for Implementation of the LIZ

Hardware and Software Specifications

Please refer to the INEXTOR Requirements.

License

The license file ".licx" containing the Liz license is required at this step. The Liz license is provided by your Service Provider.

Software Setup

The components will be installed using the MSI package.

> The installation and the configuration have to be done with administrator privileges on the local system. If the Service is intended to log on using a user account, it is preferable for the account to be local administrator of the system.
>
> If the user is not local administrator on the PC, then you need to verify that the user has adequate rights, e.g. the user must have rights for the file system and being able to logon as a service. It is recommended that the service logs on to a user account on the domain. The user does not have to be administrator on the domain.

> Make sure before setup that the clock of the system on which the Liz is to be installed is synchronized to the clock of the Gate server.

Required Firewall Settings

See page Required Firewall Settings Liz

Installation Media

The following items are provided in the installation media:

Item	Type	Description
Liz Installer	.msi file	Installation software, for new installation or upgrade
License file	.licx file	Liz license, provided by your Service Provider
Item management configuration	.imc file	Configuration for the serialization installation

General installation procedures: Vault proxy procedures

New Installation Steps for Vault Proxy Setup

New Installation

Order	Action	Link
1	Prepare Server	Vault Proxy: Prerequisites
2	Vault Proxy database installation	Database Setup
3	Deploy Vault Proxy web application to IIS	InextoClientWS Setup
4	Configure IIS	Vault Proxy: IIS Configuration
5	How to configure MCG/Gate to communicate with Vault via Vault Proxy	Vault Proxy Integration

General Installation Procedures: Migrating from Codentify to Inextor

- This procedure applies to manufacturers who already have INEXTOR installed, but still have some factories operating with Codentify.
- The Checker is continuously migrated to the Vault, which means there is nothing to do at the Checker and Vault level.
- Migration from MCG to Gate is explained in page Migration (MCGv4 to Gate v10).
- Migration from COG to Liz is explained in page Migration (COG v4 to Liz v10).
- The new Gate can stay connected to the Checker at first, and can then be connected to the Vault.
- There is no need to migrate data history, since everything in the COG and MCG is uploaded continuously to the Checker, which in turn is continuously migrated to the Vault.
- The most time-consuming part of the migration process is the migration of Master Data, including printing data (templates etc.).
- Everything must be tested by the INEXTO Support Team first, using backups from a COG and from an MCG provided by the customer.

Here is a list of steps to be followed:

#	Tasks	Expected duration	Documentation link	Comments
1. Prepare Infrastructure				
1.1	Define which Infrastructure is needed to host INEXTOR	NA	Requirements	Dedicated servers vs Private Cloud (SDI) vs AWS For TPMs, it is envisaged to go for a SAAS solution for INEXTOR + INEXPRESS (possible PoC candidate: Macedonian factory)
1.2	Review factory legal obligations (regulations)	NA	NA	The goal is to determine any potential impact on INEXTOR/INEXPRESS Possible new regulations to consider: Norway?, GB (Brexit)?
1.3	Choose a pilot linkup	NA	NA	
2. Gate Installation				
2.1	Verify that the Gate prerequisites are fulfilled	2-4h	Gate installation prerequisites	
2.2	Install the Gate database and MSI		Gate installation	
2.3	Migrate the MCG to the Gate		MCG to Gate Migration	
2.4	Integrate the Gate with SMDL		PMI Product Integration	

2.5	Enable the Gate-MCG side by side installation		Gate-MCG side by side	
2.6	Start the Gate and verify the installation		Gate Installation	
2.7	Configure Interfaces Driver V1 for Ranger and forwarding to MCG		Interfaces Driver V1	
2.8	Configure Product Integration		PMI Product Integration	
2.9	Configure INEXPRESS V2 operation driver	1h	Configuration for INEXPRESS V2	
3. Regulations				
3.1	Optional: Configuration for TPD	0.5h	TPD Gate Configuration	
3.2	Optional: Configuration for DLR (UAE/KSA)	0.5h	DLR Gate Configuration	
3.3	Optional: Configuration for CRPT	1h	.GateService CRPT Configuration	Specific configuration for Kazakhstan
3.4	Optional: Configuration for the specific implementation		To be determined at implementation time	
3.5	Optional: Configuration for Jordan		INEXTOR Jordan Operational Guide	
4. Liz Installation				
4.1	Verify that the Liz prerequisites are fulfilled	1h	Liz installation prerequisites	
4.2	Install the Liz MSI and configure the Liz		Liz Installation Configure Liz on First Run	
4.3	Customize the Linkup Setup on the Liz according to the line setup	2h	Create Linkup Setup	
4.4	Configure the Printers drivers		Printer Drivers	
4.5	Configure PMI MES Integration (including MES script for INEXPRESS property conversions)	0.5h	PMI MES Driver	MES or iMel integration (should be similar)
4.6	Configure INEXPRESS V2 operation driver	0,5h	Liz Inexpress V2 operation	

			driver	
5. Prepare Master Data				
5.1	Prepare an inventory of: - the templates which need to be migrated - the list of properties which are required for printing		NA	We highly recommend to perform this step. This is also the opportunity to clean all templates up.
5.2	Migrate the templates from the MCG to the Gate	0.5h	MCG to Gate Migration (section manual migration)	
5.3	Recreate the templates for printers which cannot be migrated (e.g. Dseries)		Printer Drivers	Expected duration depends on the number of templates
6. Production Test				
6.1	Production test on a pilot linkup with a pilot Product		NA	
6.2	Test with the first packs including these verifications: - test of proper print of HR + MR codes at each packaging level - code verifier check for each packaging level - Inexto support should also verify chunk production data synchronization with Gate and Vault		NA	

Migration (COG v4 to Liz v10)

A migration tool has been created to migrate the master data from an Cog v4.x to a new database for the Liz v10.

The master data includes:

- Certificates
- CogConfig
- Msk
- PrinterDrivers
- PrinterConfigs

Prerequisites

Before running the migration tool, a side-by-side installation of a Liz v10 must be performed on the machine where the COG v4 is running.

Note: After the execution of the Migration Tool it will not be possible to run the COG v4 anymore.

Migration process

Start the Migration Tool

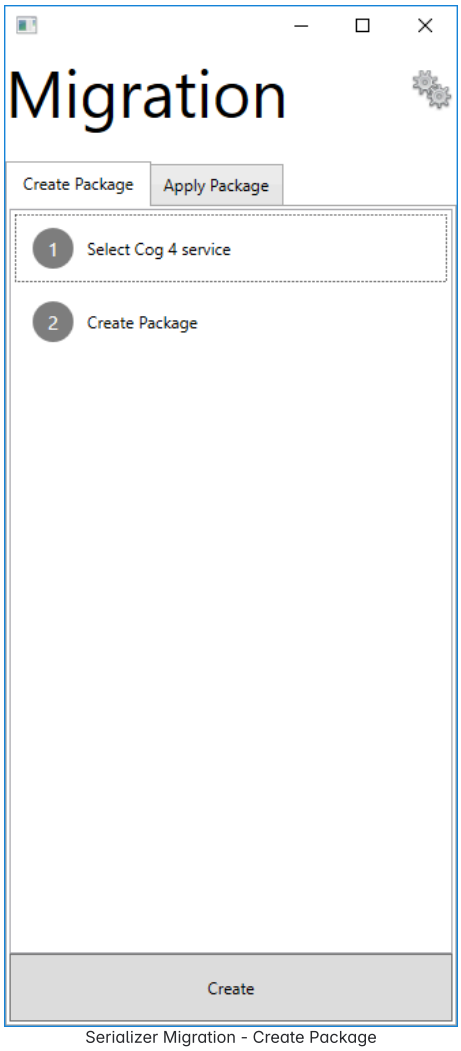
Launch the Migration tool provided with Inextor v10.

Create Package

The first operation is to Create a Package. By clicking on the Create button, the tool extracts the configuration from an installation of COG v4.x.

The tool will try to locate and stop the COG Service v4. In case of failure, it will ask for the location of the service executable and will also require to stop the service manually.

The result of this operation will be a package file (.pkg).



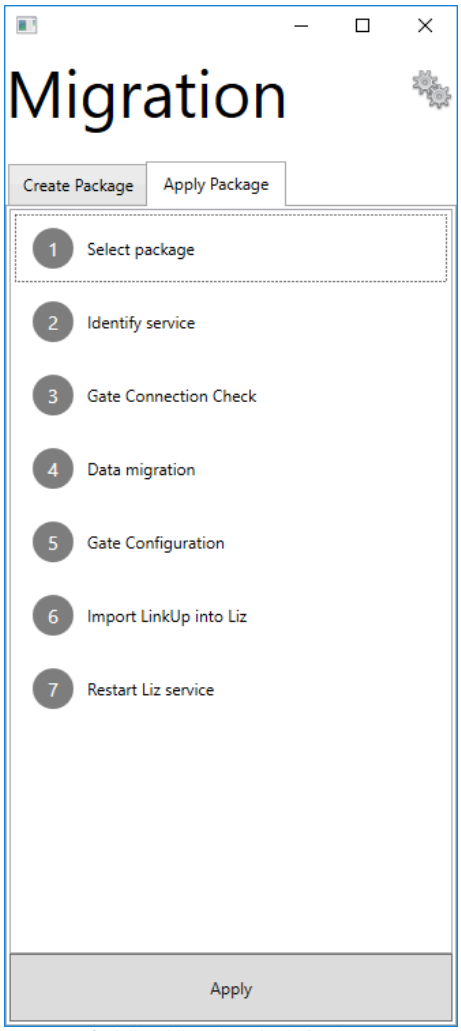
Serializer Migration - Create Package

Apply Package

By clicking on Apply Package the tool configures the Liz v10 with the information extracted from the COG v4.

The tool will ask for the .pkg file generated in the previous step. It will also try to locate and stop Liz v10 service, asking the user to provide the actual location in case of failure.

Note : if the Liz port is not the default one (i.e. 8087), the configuration file of the migration tool needs to be updated. Open the configuration file and find the *appSettings* section, modify the *TargetPort* value to the value that is used. Then restart the tool so that the modification is taken into account.

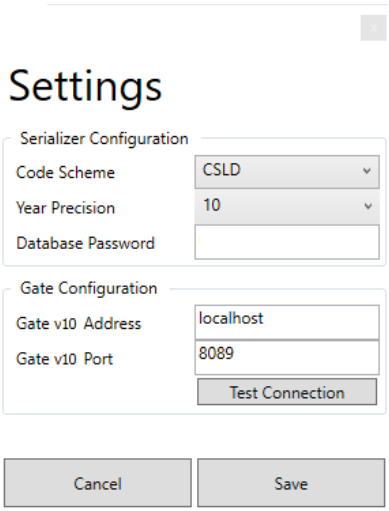


Serializer Migration - Apply Package

Settings

The following settings can be edited by clicking on the Settings Icon  located in the top-right corner of the main window:

- Code Scheme: CSLD or CSLB (choose **CSLD**, CSLB is deprecated)
- Year Precision: 10 or 20 (default: 10)
- Database Password: (default: mcgenerator)
- Gate v10 Address: (default: localhost)
- Gate v10 Port: 8089



Serializer Migration - Settings

Compare Report

Database comparisson report between **COG 4.4.3** and Liz **10.1.0.3554** Cog_Liz_DB_report.html. Note: system objects and properties are not included.

Codentify Inextor Migration: Migration (MCGv4 to Gate v10)

A migration tool has been created to migrate the master data from an MCG v4.4.4 to a new database for the Gate v10.

The master data includes:

- Certificates
- Checker & connections
- Gate identity
- Tax levels
- Calculated properties
- Printer drivers*
- Layouts
- Templates*

*The original template of the following printer drivers is modified during the migration:

- DSeries (Markzone does not exist on Codentify template but is required on Inextor's ones, it is thus added during the migration)

* The following printer drivers / templates cannot be migrated automatically due to limitation of the printer :

- -

The migration does not migrate the production data. The old database needs to be kept for archiving purpose and a side-by-side installation need to be envisioned for a smooth transition in regards of the external systems.

Migration Procedure

In order to migrate the MCG v4, the following procedure should be executed:

1. Ensure you have the connection information to MCG database v4.4.
2. Execute a blank installation (side-by-side) of the Gate v10 (see Gate Setup). (Do **NOT** start the Gate service)
3. Backup the new Gate database.
4. Execute the migration tool (see Migration tool usage) providing the connection strings to the source v4 database and target v10 database

   Example: Inexto.S3r.Migrate.exe *"Inexto.S3r.Migrate.Mcg440ToGate1000, Inexto.S3r.Migrate.Library" -SourceDb="Data Source=OLDSERVER\PRD;Initial Catalog=MCG43;Integrated Security=True" -DestinationDb="Data Source=NEWSERVER\PRD;Initial Catalog=GATE10;Integrated Security=True" -Variable:"ManufacturingCenterId"="AA" -Variable:"Gate Name"="AA factory" -Variable:"Gate Description"="Aaaaaa factory"*

   a. If the migration has been succesful, at this stage, layouts and templates that are mapped to a layout have been migrated on the new Gate v10.
   It is recommended to verify and compare the two template lists to identify which templates have not been migrated to Gate v10.
   Possible causes are:
      i. the template is not mapped to any layout (not used in any production).
      ii. the migration process was not able to parse the template (malformed template).

b. If the migration has failed, check the data existing in the MCG v4 and fix it if necessary. Once done, restore the backup performed on step 3 and execute the migration tool again.

5. Migrate the COGs v4.4.
   After the migration from MCG v4 to Gate v10 is completed it is possible to migrate some COGs in the factory to Lizs v10. These new Lizs will connect to the new Gate v10 (see Migration (COG v4 to Liz v10)).
6. Migrate layouts and templates afterwards
   At that stage, it is very likely that operators continue to add new layouts and templates on the MCG v4 for the productions on the COGs that have not been migrated to Lizs v10 yet.
   The Manual migration tool provides the ability to migrate those information at anytime after the first migration (see section below).

Migration tool usage

The migration is executed by a generic command line tool called "Inexto.S3r.Migrate.exe" (download link: Inexto.S3r.Migrate_10.1.0.4592.zip)

The tool is called by invoking it in a command line with the following parameters:

Inexto.S3r.Migrate.exe [MigrationAssembly] -SourceDb=[SourceDatabaseConnectionString] -DestinationDb=[DestinationDatabaseConnectionString] [-Variable:[Key]=[Value] -Variable:[Key]=[Value] ...]

The mapping between layout Types and Unit Types configured in the file *Inexto.S3r.Migrate.exe.config* can be customized.By default, the configuration looks like the following:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <configSections>
    <section name="custom" type="Inexto.S3r.Migrate.Configuration.CustomConfigSection, Inexto.S3r.Migrate.Library" />
  </configSections>
  <custom>
    <unitTypeMappings>
      <add layoutTypeName="BUNDLE" mapsOnUnitType="Bundles" />
      <add layoutTypeName="CASE" mapsOnUnitType="Cases" />
      <add layoutTypeName="PACK" mapsOnUnitType="Packs" />
      <add layoutTypeName="BUNDLES" mapsOnUnitType="Bundles" />
      <add layoutTypeName="CASES" mapsOnUnitType="Cases" />
      <add layoutTypeName="PACKS" mapsOnUnitType="Packs" />
      <add layoutTypeName="BUNDLETEMPLATE" mapsOnUnitType="Bundles" />
      <add layoutTypeName="CASETEMPLATE" mapsOnUnitType="Cases" />
      <add layoutTypeName="PACKTEMPLATE" mapsOnUnitType="Packs" />
      <add layoutTypeName="BUNDLETEMPLATES" mapsOnUnitType="Bundles" />
      <add layoutTypeName="CASETEMPLATES" mapsOnUnitType="Cases" />
      <add layoutTypeName="PACKTEMPLATES" mapsOnUnitType="Packs" />
    </unitTypeMappings>
  </custom>
```

Migration Assembly

The migration assembly "*Inexto.S3r.Migrate.Mcg440ToGate1000, Inexto.S3r.Migrate.Library*" handles the migration from MCG v4.4.4 to Gate v10.

It has the following variables:

Variable	Description
ManufacturingCenterId	The Manufacturing Center ID of the migrated Gate (also called MCID).
Gate Name	A short name to describe the Gate.
Gate Description	A description of the Gate.

Example invocation:

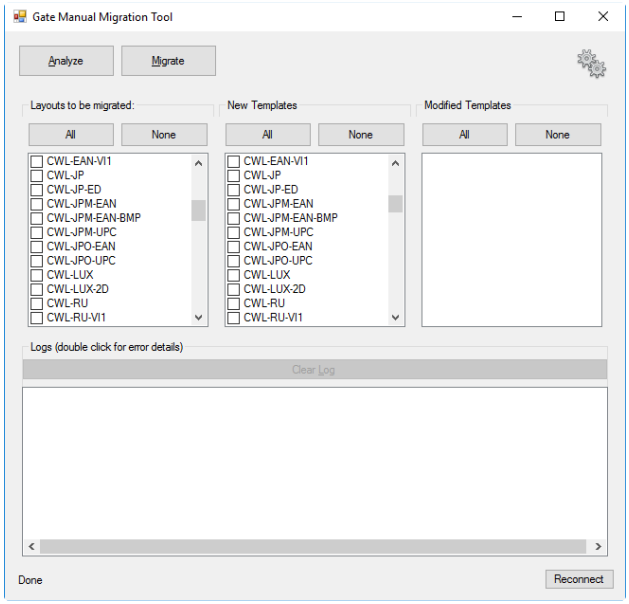
```
Inexto.S3r.Migrate.exe "Inexto.S3r.Migrate.Mcg440ToGate1000, Inexto.S3r.Migrate.Library" -SourceDb="Data Source=OLDSERVER\PRD;Initial Catalog=MCG4;Integrated Security=True" -DestinationDb="Data Source=NEWSERVER\PRD;Initial Catalog=GATE10;Integrated Security=True" -Variable:"ManufacturingCenterId"="AA" -Variable:"Gate Name"="AA factory" -Variable:"Gate Description"="Aaaaaa factory"
```

Manual migration

After the first migration, typically both MCG v4 and Gate v10 coexist, to ensure productivity of the lines that have not been migrated yet. During this transition phase, new templates and layouts must be created on both systems.

To reduce the workload related to templates and layouts definition, we provide the **Manual Migration Tool**, which provides the ability to migrate layouts and templates anytime.

Download link : https://inexto.atlassian.net/wiki/download/attachments/394266757/Inexto.S3r.ManualMigrationTool.exe?api=v2

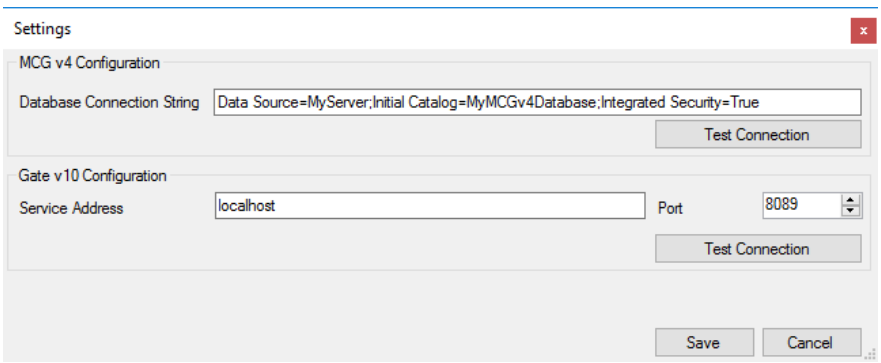


Manual migration tool window

This tool analyzes both the MCG v4 and the Gate v10 to identify the differences. Among them, users can select what they would like to migrate from the old installation to the new one.

Configuration of the tool

The configuration mapping between Layout Types and Unit Types can be configured in the file app.*config* file, as explained above for *Inexto.S3r.Migrate.exe*.
All the other settings are editable from the Settings Editor, which can be opened by clicking on the double cog icon at the top-right corner of the application window.



Manual migration configuration file

The editable settings are:

- MCG 4 Database Connection String: the connection string to connect to the database of MCG v.4. The *Test Connection* button can help to verify that the connection string is correct.

- Gate 10 Service Addres and Port: the IP endpoint to connect to the Gate UI service. Use the *Test Connection* button to check that the tool can connect to the server.