

Izytroniq Enterpriseupdate

Gültig ab Version 4.8.

Vorbereitung: Als erstes sollte sichergestellt werden, dass das Dienstkonto der Izytroniq bekannt ist und man darauf Zugriff hat. Das Konto ist der DB_Owner der Datenbanken Izytron.Iq und SyncService. Man kann es auch herausfinden, indem man prüft welches Konto in der services.msc den Dienst Izytroniq.Backend.Syncservice ausführt.

Für das Update werden die gleichen Installationsdateien wie für eine Installation verwendet. Diese können hier heruntergeladen werden:

<https://izytron.com/downloads.php>

Alle Pakete sollten mit dem oben genannten Benutzer ausgeführt werden oder als Alternative mit einem sysadmin des SQL Servers.

Izytroniq Enterprise benötigt Net Framework 4.8 für den Betrieb.

Windows Server 2016 und 2019 müssen das Update nachinstallieren:

<https://support.microsoft.com/de-de/topic/microsoft-net-framework-4-8-offline-installer-f%C3%BCr-windows-9d23f658-3b97-68ab-d013-aa3c3e7495e0>

Wichtig: Sollten Sie Ultimate-Versionen der Izytroniq Enterprise im Einsatz haben müssen diese *alle* ihre ausgecheckten Daten zum Zeitpunkt des Updates wieder eingecheckt haben. Ansonsten kann es zu Datenverlust oder Datenkompromittierung führen.

Es darf also niemand draußen im Einsatz mit Offlinedaten unterwegs sein, während ein Update durchgeführt wird.

Wenn diese Voraussetzungen erfüllt sind und das Update durchgeführt wird, kann der Dienst Izytroniq Backend SyncService gestoppt werden.

1. Update der Datenbank

Wird die Datenbank mit einem SQL User (statt eines AD oder Windows-Users) betrieben sollten die Registryeinträge der DB vorher gesichert werden. Der Pfad dieser Einträge ist

HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\Gossen-Metrawatt\IZYTRON.IQ\BackEndDatabaseConnectionString

HKEY_LOCAL_MACHINE\SOFTWARE\Gossen-Metrawatt\IZYTRON.IQ\BackSyncServiceDatabaseConnectionString

Sollte die Anwendung seit der Installation nicht verändert worden sein (Anderer SQL Server z.B.) kann das Update der Datenbank via Doppelklick ausgeführt werden, da die Informationen bei der Installation in einem Cache abgelegt wurden.

Sollte dies nicht der Fall sein muss der Setup cache gelöscht werden:

Name	Type	Data
(Default)	REG_SZ	(value not set)
ApplicationInstallDir	REG_SZ	C:\Program Files\Gossen-Metrawatt\IZYTRON.IQ
ApplicationServerBaseAddress	REG_SZ	https://mwizytestnue02.gmc-instruments.com/izytron.iq
ApplicationVersion	REG_SZ	4.6.2.1
BackSyncServiceDatabaseConnectionString	REG_SZ	Data Source=(local)\SQLEXPRESS;Initial Catalog=SyncService;Integrated Security=True;
BackSyncServiceVersion	REG_SZ	4.6.2.1
DatabaseInstallDir	REG_SZ	C:\ProgramData\Gossen-Metrawatt\IZYTRON.IQ
DatabaseVersion	REG_SZ	4.6.2.1
DependentComponents	REG_SZ	IZYTRON.IQ BackEnd ApplicationServer=4.6.2.1,IZYTRON.IQ BackEnd Database=4.6.2.1
InstallationStatus	REG_SZ	Updated
InstallationType	REG_SZ	Client
LauncherInstallPath	REG_SZ	C:\Program Files\Gossen-Metrawatt\IZYTRON.IQ Launcher\IZYTRON.IQ.Launcher.exe
LauncherVersion	REG_SZ	4.6.2.1
SetupCache	REG_SZ	PABTAGUAdAB1AHAAQwBhAGMAaABIAD4ADQAKACAAIAA8AFAAcgBvAGQAdQBjAHQAIABOAGEAbQBIAID
UpdateBaseUrl	REG_SZ	https://update.gossenmetrawatt.com/IZYTRON.IQ.Update_v4.1/

In diesem Fall ist dann das Database.msi mit dem Parameter BACKEND_SERVERNAME=(name der SQL Instanz) auszuführen

2. Update des Application Servers / SyncService

Der Zertifikatseintrag aus folgender Configdatei kann gesichert werden:

C:\Program Files\Gossen-Metrawatt\IZYTRON.IQ
BackEnd SyncService\ IZYTRON.IQ.SyncSvc.exe.config

Der String hat folgendes Format.

```
<serviceCertificate findValue="1234567890ABCDEF"
storeLocation="LocalMachine" storeName="My"
x509FindType="FindByThumbprint" />
```

Es wird nur dieser eine String gesichert, da der Rest der Configdateien sich zwischen den Versionen ändern können.

Die beiden Pakete werden ausgeführt. Danach müssen in folgenden Dateien der oben gesicherte String des Zertifikats wieder hinterlegt werden.

C:\inetpub\wwwroot\IZYTRON.IQ\web.config

C:\Program Files\Gossen-Metrawatt\IZYTRON.IQ
BackEnd SyncService\ IZYTRON.IQ.SyncSvc.exe.config

Beispiel des finalen Ergebnisses aus der web.config:

```
<behavior name="IZYTRON.IQ_Behavior">
  <callInspectorEndpointBehavior />
  <!-- To avoid disclosing metadata information, set the values below to false before deployment -->
  <serviceMetadata httpGetEnabled="true" httpsGetEnabled="true" />
  <!-- To receive exception details in faults for debugging purposes, set the value below to true. Set to false b
  <serviceDebug includeExceptionDetailInFaults="true" />
  <serviceCredentials>
    <userNameAuthentication userNamePasswordValidationMode="Custom" customUserNamePasswordValidatorType="EtcPro.Bar
    <serviceCertificate findValue="40AA97B03982579D6F4675F7AB287E4217990552" storeLocation="LocalMachine" storeName
  </serviceCredentials>
</behavior>
</serviceBehaviors>
```

3. Benutzerberechtigungen.

Jetzt müssen im IIS Applicationpool izytroniq.Backendpool der Benutzer wieder eingetragen werden:

Rechtsklick – Advanced Settings / erweiterte Einstellung

Runterscrollen zu Process Model / Prozessmodell

Identity / Identität bearbeiten (Default ApplicationPoolIdentity)

Application Pools

This page lets you view and manage the list of application pools on the server. Application pools are associated with worker processes, contain one or more applications, and provide different applications.

Name	Status	.NET CLR V...	Managed Pipel...	Identity	Applications
.NET v4.5	Started	v4.0	Integrated	ApplicationPoolId...	0
.NET v4.5 Classic	Started	v4.0	Classic	ApplicationPoolId...	0
DefaultAppPool	Started	v4.0	Integrated	ApplicationPoolId...	1
IZYTRON.IQ.BackEnd.Pool	Started	v4.0	Integrated	.\izyadmin	1

Advanced Settings

Start Mode	OnDemand
▼ CPU	
Limit (percent)	0
Limit Action	NoAction
Limit Interval (minutes)	5
NUMA Node Affinity Mode	Soft
NUMA Node Assignment	MostAvailableMemory
Processor Affinity Enabled	False
Processor Affinity Mask	4294967295
Processor Affinity Mask (64-bit c	4294967295
Processor Group	0
▼ Process Model	
Generate Process Model Event L	
Identity	.\izyadmin
Idle Time-out (minutes)	20

Application Pool Identity

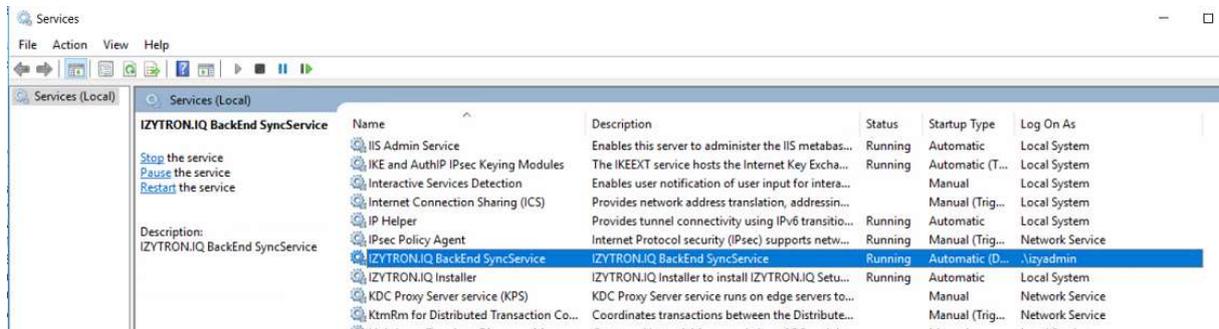
Built-in account:

Custom account:

.\izyadmin Set...

OK Cancel

Sowie in der Services.msc der Benutzer im Dienst wieder hinterlegt werden:



Danach wird der Dienst neu gestartet.
Die Benutzer gehen verloren, weil die exe-dateien dahinter ausgetauscht werden und Windows damit nicht zurechtkommt.

4. Update LicensingTool

Das Update des Lizenztools ist optional.

Mit folgenden Prüfungen kann man dann testen, ob das Update erfolgreich war:

Der Dienst SyncService darf beim Starten über Services.msc im Eventviewer keinen Error werfen. Lediglich der Hinweis, dass der Dienst gestartet wurde sollte dort stehen.

Der SQL Query

```
select value from database_info
```

gibt die neue Version zurück.



Die Testseiten können ohne Fehler aufgerufen werden. Diese lauten

- <https://<servername>/syncservice>
- <https://<servername>/bits>
- <https://<servername>/izytron.iq/handshakeremoteservice.svc>

wenn alle Tests bestanden sind sollte die izytroniq Enterprise wieder laufen und man kann jetzt durch starten der Clientanwendungen das automatische Clientupdate starten.

Troubleshooting:

Beim Überschreiben älterer Versionen des ApplicationServers kann es vorkommen, dass der ApplicationPool verschwindet. In diesem Fall einfach das Paket ApplicationServer komplett deinstallieren und neu installieren. Es müssen mindestens die vier Einträge aus dem screenshot oben zu sehen sein.

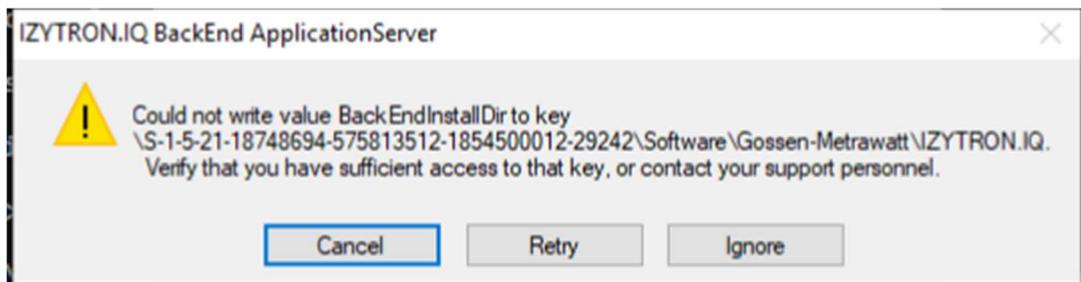
Das gleiche gilt für den Eintrag BackSyncServiceDatabaseConnectionString in der Registry. Sollte dieser nicht erstellt worden sein das Paket SyncService deinstallieren und erneut installieren.

In dem Fall muss der SQL Server dann auch in der Registry nachgetragen werden:

HKEY_LOCAL_MACHINE\SOFTWARE\Gossen-Metrawatt\IZYTRON.IQ\BackSyncServiceDatabaseConnectionString

Data Source=**SQLSERVER**;Initial Catalog=SyncService;Integrated Security=True;MultipleActiveResultSets=True

Manchmal kann es bei der Durchführung des Updates von ApplicationServer oder SyncService zu folgender Nachricht kommen:



Ursache hierfür ist häufig, dass die Berechtigungen mit der die Izytroniq erstmals installiert wurde von Windows nicht mehr nachvollzogen werden können. In diesem Fall cancel, dann das entsprechende Paket komplett deinstallieren und dann neu installieren.

https://.../IZYTRON.IQ/HandshakeRemoteService.svc

Login failed. The login is fr...

Server Error in '/IZYTRON.IQ' Application.

Login failed. The login is from an untrusted domain and cannot be used with Integrated authentication.

Description: An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it originated in the code.

Exception Details: System.Data.SqlClient.SqlException: Login failed. The login is from an untrusted domain and cannot be used with Integrated authentication.

Source Error:

An unhandled exception was generated during the execution of the current web request. Information regarding the origin and location of the exception is provided below.

Stack Trace:

```
[SqlException (0x80131904): Login failed. The login is from an untrusted domain and cannot be used with Integrated authentication]
System.Data.ProviderBase.DbConnectionPool.TryGetConnection(DbConnection owningObject, UInt32 waitForMultipleObjectsTimeout,
System.Data.ProviderBase.DbConnectionPool.TryGetConnection(DbConnection owningObject, TaskCompletionSource`1 retry, DbConnection
System.Data.ProviderBase.DbConnectionFactory.TryGetConnection(DbConnection owningConnection, TaskCompletionSource`1 retry,
System.Data.ProviderBase.DbConnectionInternal.TryOpenConnectionInternal(DbConnection outerConnection, DbConnectionFactory
System.Data.SqlClient.SqlConnection.TryOpenInner(TaskCompletionSource`1 retry) +202
System.Data.SqlClient.SqlConnection.TryOpen(TaskCompletionSource`1 retry) +413
System.Data.SqlClient.SqlConnection.Open() +128
System.Data.Entity.Infrastructure.Interception.InternalDispatcher`1.Dispatch(TTarget target, Action`2 operation, TIntercepte
System.Data.Entity.Infrastructure.Interception.DbConnectionDispatcher.Open(DbConnection connection, DbInterceptionContext
System.Data.Entity.SqlServer.<>c__DisplayClass2_0.<Execute>b__0() +18
System.Data.Entity.SqlServer.DefaultSqlExecutionStrategy.Execute(Func`1 operation) +234
System.Data.Entity.Core.EntityClient.EntityConnection.Open() +321
```

In diesem Fall ist der Benutzer nicht berechtigt, auf den SQL Server zuzugreifen. Hier müssen die Registryeinträge wiederhergestellt werden (Siehe Punkt 1)