Windows Server 2012-2016 Install Guide

March 20, 2025

Contents

This document provides a Windows Server 2012/2016 install guide. The guide can be followed for Windows Server 2012/2016 installation or serve as a starting point for installing on later Windows Server OS. You should read the Deployment documentation beforehand, in order to understand the components and their roles.

The Sirenia solutions are capable of being installed and operated on a Windows Server 2012/2016 platform as selfcontained windows services. Further more, the deployment is dependant on a postgres instance, which should also be installed.

System requirements

The minimum system requirements for a minimal installation on Windows Server 2012/2016. For more then a minimal installation referer to the sizing guidelines of the Sirenia software components.

- OS Versions: Server 2012, Server 2016
- RAM: 8GB
- Disk space: 64GB minimum recommendation for Windows

Prepare Install

The first step is to prepare an administrator PowerShell

- Log in to Windows Server.
- Click the Search icon in the taskbar and type powershell in the search box.
- Right-click Windows PowerShell in the search results and select Run as administrator from the menu.

- Enter administrator credentials as prompted.

Install Postgres 10

At the powershell command prompt, use this command

```
cd ~
Import-Module BitsTransfer
Start-BitsTransfer -source "https://oscg-downloads.s3.amazonaws.com/
    packages/PostgreSQL-10.7-1-win64-bigsql.exe"
.\PostgreSQL-10.7-1-win64-bigsql.exe
```

Follow onscreen install. Set username and password to `postgres` for testing. For production you should use a harder password.

Find `PostgreSQL 10.5-1 Server` in the service panel. Open `Properties`. Select `Recovery` tab. Select `Restart the Service` on all three drop downs `Firstfailure, Second failure, and Subsequent failures`. Press `Apply` and `OK`

Obtain and unpack the Sirenia Software

Get Kwanza and Cuesta from your proviede software repository. In powershell

```
mkdir "C:\Program Files (x86)\Sirenia\Deploy"
cd "C:\Program Files (x86)\Sirenia\Deploy"
```

- Extract Kwanza in `C:\Program Files (x86)\Sirenia\Deploy`
- Extract Cuesta in `C:\Program Files (x86)\Sirenia\Deploy`

Install Kwanza as a Service

Kwanza is the backend application used for configuration and communication. First Kwanza must be installed.

Create `key.pem` and `cert.pem` using kwanza. Replace `localhost` with your needed Fully Qualified Domain Name(s). You MUST use all small letters in the fqdn. eg. some.sirenia.io

```
cd "C:\Program Files (x86)\Sirenia\Deploy"
./kwanza_windows_amd64.exe cert --subjects localhost
```

Create `kwanza conf`

```
notepad .kwanza.conf
```

Add lines like these

```
users:
  admin:
     d224cfd091471383708424f3e494f8029b456b0e559fe82ee9adb5b66a7f1e55
  john:
     d224cfd091471383708424f3e494f8029b456b0e559fe82ee9adb5b66a7f1e55
  jonathan:
     d224cfd091471383708424f3e494f8029b456b0e559fe82ee9adb5b66a7f1e55
```

Test Kwanza configuration. The format of the database string is `pg://user:pass@localhost/database` change to fit your installation. Also note that you can adjust TLS version and supported cipher suites.

```
cmd /c kwanza_windows_amd64.exe serve --database pg://postgres:
    postgres@localhost/kwanza --minTlsVersion 1.2 --cipherSuites "
    TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256,
    TLS_RSA_WITH_AES_128_GCM_SHA256,TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
    ,TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256" --preferServerCipherSuites
     "true" --cert "$(pwd)\cert.pem" --key "$(pwd)\key.pem" --
    strictTransportSecurity "true" --auth jwt --config "$(pwd)\.kwanza.
    conf" --logFile "/kwanza.log" --port 8000
```

If no errors are displayed, stop again with `ctrl^c` and install Kwanza service

```
cmd /c kwanza_windows_amd64.exe serve --database pg://postgres:
    postgres@localhost/kwanza --minTlsVersion 1.2 --cipherSuites "
    TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256,
    TLS_RSA_WITH_AES_128_GCM_SHA256,TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
    ,TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256" --preferServerCipherSuites
     "true" --cert "$(pwd)\cert.pem" --key "$(pwd)\key.pem" --
    strictTransportSecurity "true" --auth jwt --config "$(pwd)\.kwanza.
    conf" --logFile "/kwanza.log" --port 8000 --install
cmd /c sc config eu.sirenia.kwanza depend="PostgreSQL 10.5-1 Server"
```

Find `Sirenia Context Management Registry` in the service panel. Open `Properties`. Select `Recovery` tab. Select `Restart the Service` on all three drop downs `Firstfailure`, `Second failure`, `and Subsequent failures`. Press `Apply` and `OK`

Start Kwanza service in services. Find “Sirenia Context Management Registry” in the service panel. Push start. Check the status from powershell. Or start the service from powershell.

```
net start eu.sirenia.kwanza
service eu.sirenia.kwanza
```

Should display

```
Status   Name               DisplayName
------   ----               -----------
Running  eu.sirenia.kwanza  Sirenia Context Management Registry
```

Kwanza will use the self-signed cert created earlier. Alternatively copy valid cert for prod here `C:\Program Files (x86)\Sirenia\Deploy` It must be a valid x.509 certificate with a full trust chain to a CA in PEM format.

Install Cuesta as a Service

Cuesta is the graphical web application used for configuration on top of the Kwanza application. Cuesta must now be installed. Change settings.js for cuesta

```
cd "C:\Program Files (x86)\Sirenia\Deploy"
notepad settings.js
```

Add lines like these (change localhost with your FQDN

```
window.env = window.env || {};
window.env['KWANZA_URL'] = 'https://localhost:8000/v1';
window.env['KWANZA_STREAMURL'] = 'wss://localhost:8000/v1/stream';
```

Test Cuesta configuration

```
cmd /c cuesta_windows_amd64.exe serve --cert "$(pwd)\cert.pem" --key "$
    (pwd)\key.pem" --strictTransportSecurity "true" --extras /settings.
    js="$(pwd)\settings.js"
```

If no errors are displayed, stop again with `ctrl^c` and install Cuesta service

```
cmd /c cuesta_windows_amd64.exe serve --cert "$(pwd)\cert.pem" --key "$
    (pwd)\key.pem" --strictTransportSecurity "true" --extras /settings.
    js="$(pwd)\settings.js" --install
```

Find `Cuesta_Windows` in the service panel. Open `Properties`. Select `Recovery` tab. Select `Restart the Service` on all three drop downs `Firstfailure`, `Second failure`, and `Subsequent failures`. Press `Apply` and `OK`

Start Cuesta service in services. Find `Cuesta_Windows` in the service panel. Push start. Check the status from powershell. Or start the service from powershell.

```
net start eu.sirenia.static.cuesta_windows_amd64.exe
service eu.sirenia.*
```

Should display two services running

```
Status   Name               DisplayName
------   ----               -----------
Running  eu.sirenia.kwanza  Sirenia Context Management Registry
Running  eu.sirenia.stat... cuesta_windows_amd64.exe is a serve...
```

Open Firewall

If your server is deployig a firewall, you must grant open access to the following ports

- TCP/80 - For Cuesta
- TCP/443 - For Cuesta
- TCP/8000 - For Kwanza
- TCP/8001 - For Kwanza

Test

Ok, we are ready to test the complete setup

Login to Cuesta

- `https://localhost/`
- `user:john pass:1234`

If no errors show up, we are ready to go.

Upgrade Procedure

If you need to install a newer version of the Cuesta og Kwanza service, follow these steps. At the powershell command prompt:

```
service eu.sirenia.*
```

Should display two services running

```
Status   Name               DisplayName
------   ----               -----------
Running  eu.sirenia.kwanza  Sirenia Context Management Registry
Running  eu.sirenia.stat... cuesta_windows_amd64.exe is a serve...
```

Stop Kwanza and Cuesta service in the service panel. Push stop for both services.

```
service eu.sirenia.*
```

Should display two services stopped.

```
Status   Name               DisplayName
------   ----               -----------
Stopped  eu.sirenia.kwanza  Sirenia Context Management Registry
Stopped  eu.sirenia.stat... cuesta_windows_amd64.exe is a serve...
```

Uninstall one or both services.

```
cd "C:\Program Files (x86)\Sirenia\Deploy"
cmd /c cuesta_windows_amd64.exe serve --uninstall
cmd /c kwanza_windows_amd64.exe serve --config "$(pwd)\.kwanza.conf" --
    uninstall
service eu.sirenia.*
```

Should now display no services.

Now follow the install procedure as explained earlier in this document.

Sirenia Analytics

If you have acquired a license to the Data Driven Operational Intelligence solution Sirenia Analytics, follow the instalation guide here. You can deploy this on the same server as Cuesta and Kwanza (assuming it is sized coorectly), or on is's own. If you install on a new server.

Install Fluentd as a Service

Install td-agent. Download the .msi file and install the software. At the powershell command prompt, use this command

```
cd ~
Import-Module BitsTransfer
```

```
Start-BitsTransfer -source "http://packages.treasuredata.com.s3.
    amazonaws.com/3/windows/td-agent-3.4.1-0-x64.msi"
.\td-agent-3.4.1-0-x64.msi
```

To configure Fluentd do the following. First, please prepare your config file

```
cd /opt/td-agent/etc/td-agent/
move td-agent.conf td-agent.conf-orig
notepad td-agent.conf
```

Add this to the file

```
#UDP input
<source>
  @type udp
  #8081 is stats (info-log)
  tag manatee.8081 # required
  format none
  port 8081 # optional. 5160 by default
  bind 0.0.0.0 # optional. 0.0.0.0 by default
  message_length_limit 1MB
</source>

<source>
  @type udp
  #8082 is everything (debug-log)
  tag manatee.8082 # required
  format none
  port 8082 # optional. 5160 by default
  bind 0.0.0.0 # optional. 0.0.0.0 by default
  message_length_limit 1MB
</source>

#Filters. Everything to stdout
<filter **>
  @type stdout
</filter>

#Output
<match manatee.8081>
  @type file
  format single_value
  path          /fluentd/log/stats.manatee
```

```
  buffer_type memory
  flush_interval 0s
  append       true
</match>

<match manatee.8082>
  @type file
  format single_value
  path           /fluentd/log/all.manatee
  buffer_type memory
  flush_interval 0s
  append       true
</match>
```

After you've installed .msi package, you'll see the program called `Td-agent Command Prompt` installed. Please execute `Td-agent Command Prompt` with administrative privilege, and type the two commands below.

```
fluentd --reg-winsvc i
fluentd --reg-winsvc-fluentdopt '-c C:/opt/td-agent/etc/td-agent/td-agent.conf -o C:/opt/td-agent/td-agent.log'
```

Start Fluentd service in services. Find "Fluentd" in the service panel. Push start. Check the status from powershell. Or start the service from powershell.

```
cmd /c sc config fluentdwinsvc start="auto"
net start fluentdwinsvc
service fluentd*
```

Should display the fluentd service as running.

```
Status   Name               DisplayName
------   ----               -----------
Running  fluentdwinsvc      Fluentd Windows Service
```

Logfiles collected from the Manatee clients will be in `C:\fluentd\log\`

Restart Server

You should always finish an install procedure with a complete servere restart, to test that all services starts after a complete host restart

```
1 Restart-Computer
```