

Test system Build Diary

“Big Build” Part 3 – X3 Services (ADC) installation

Disclaimer

This document is provided "as is" and is for your guidance and educational purposes only. It does not replace the Online documentation, nor is any warranty expressed nor implied for the steps described herein.

Document Information

Author: Mike Shaw, Sage UK X3 Support Team

Contents

Test system Build Diary.....	1
Introduction	3
2023 R1 – Multi-Node install X3 services (ADC) build diary.....	4
Objective	4
Current Architecture.....	4
Documentation to use for planning and execution of this task.....	6
Summary of steps.....	7
Installation steps	7
Sage X3 Services installation steps.....	8
Check pre-requisites	8
Install and configure X3 Services.....	8
Configure Syracuse nodelocal.js and restart Syracuse.....	10
Configure SQL Server and X3 solution in X3 Administration screens	10
Republish Web services	11
Create/configure SOAP Pool.....	11
Testing.....	13
Conclusion.....	17

Introduction

What is a "Build Diary"

A Build Diary simply describes the steps taken by Sage Support to perform a task or tasks on our internal test systems. Build diaries could be created for major multi-node installations, but may also just be describing the steps taken when installing a small hotfix, or anything in-between.

Why is this being shared

It may be useful for you to see the steps we have taken to create or implement some feature or installation, as this may highlight "gotcha's", issues encountered or just give you some guidance if you are planning something similar yourself.

You could potentially use these documents as the base for your own "Workplan document" (Described in "Overview of patching X3 and supporting technologies" (<https://www.sagecity.com/gb/sage-x3-uk/b/sage-x3-uk-support-insights/posts/sage-x3-technical-support-tips-and-tricks---march-2021-index>) when you are planning your own activities

Target Audience

This document is aimed at Sage X3 Certified Technical consultants. Sage prescribe that X3 system installation, maintenance, migrations, etc. should be performed by suitably qualified Sage X3 consultants. The prerequisite consideration would be for them to have the latest "Sage X3 Certified Technical Consultant" certification. You can read more about the Sage X3 qualifications and requirements in Sage University (<https://sageu.csod.com/catalog/CustomPage.aspx?id=20000242#tc>)

Additional things to note

- This document does NOT purport to illustrate "best practice" for the task being described
- The steps described will not necessarily be for a "perfect" task, as there may have been issues that needed to be overcome, worked around, or ignored
- The Sage internal test system has network and hardware configuration specific to Sage
- The Sage internal test system does not necessarily include a Windows Domain and has Sage sandbox specific Windows security setup, so operating system permissions are generally not discussed
- If you intend to use these notes as a guide for your own activities, use with caution and perform your own testing to ensure the described steps are suitable and identify any additional considerations that apply to your own situation
- Ensure you only install and use software you are licensed for

What does this Build Diary describe?

This build diary primarily describes installing X3 Services (ADC) into my previously created multi-node 2023 R1 instance.

2023 R1 – Multi-Node install X3 services (ADC) build diary

Objective

I want to install X3 services (henceforth known as ADC) into my existing 2023 R1 system.

Current Architecture

Six Windows Server 2022 servers (Server names mzAD, mzDB, mzPRINT, mzWEB1, mzWEB2, mzWEB3)

Software already loaded:

Windows Server 2022

OpenJDK 1.8.0_282

Edge, Firefox and Chrome browsers

7-Zip 19.00

Sage X3 2023 R1 (Patch 33)

Windows users setup (Local users)

“x3admin” for installation and management

“X3run” for service runtime

mzWEB1

- Syracuse (12.18.3.1-0)
- MongoDB (4.4.12.9)
- X3 Runtime (95.2.97)
- AdxAdmin (95.2.97)
- Powershell 7.3.1

mzWEB2

- Syracuse (12.18.3.1-0)
- MongoDB (4.4.12.9)
- X3 Runtime (95.2.97)
- AdxAdmin (95.2.97)
- Powershell 7.3.1

mzWEB3

- Syracuse (12.18.3.1-0)
- MongoDB (4.4.12.9)
- X3 Runtime (95.2.97)
- AdxAdmin (95.2.97)
- Powershell 7.3.1

mzPRINT

- Elastic Search (7.16.3)
- Print Server (2.28.0.10)
- AdxAdmin (95.2.97)

mzAD

- MS AD
- Apache Load Balancer (Apache 2.4)

- X3 Console (2.57.0.11)

mzDB

- SQL Server Database (2019)
- SSMS (18.12.1)

Firewall setup

All servers are in the same MS AD domain. Firewall rules between these servers allow all traffic. Only external traffic needs to be added to the firewall rules, which should only be port 443 into the MZAD server (see notes later in this document).

Documentation to use for planning and execution of this task

Overall V12 documentation

<http://online-help.sageerp3.com/erp/12/public/index.html>

Pre-requisites

[http://online-help.sageerp3.com/erp/12/public/Prerequisites-\(Last-version\).html](http://online-help.sageerp3.com/erp/12/public/Prerequisites-(Last-version).html)

http://online-help.sageerp3.com/erp/12/public/prerequisites_overview.html

X3 Services installation

https://online-help.sageerp3.com/erp/12/public/getting-started_Sage-X3-Services-installation.html

Additional notes when installing or upgrading X3 services (New ADC)

<https://support.na.sage.com/selfservice/viewdocument.do?externalId=114447>

Summary of steps

Installation steps

- Check pre-requisites, in particular you need a license with the ADC badges and confirm compatibility with hand-held devices.
- Install and configure X3 Services
- Configure Syracuse nodelocal.js and restart Syracuse
- Configure SQL Server and X3 solution in X3 Administration screens
- Create/configure SOAP Pool
- Do initial testing using browser to confirm functionality is working as expected
- Complete testing with all makes/models of handheld devices that will be used

Sage X3 Services installation steps

Check pre-requisites

Checked and all OK

Setup new user for testing. User “adctest” in my case.

- User must have at least one ADC badge assigned
- Security profile for the user must include “Read” permission for “statusAndUsage”
- User must have “code” and “login” set the same in the user record for the folder.

Install and configure X3 Services

I am installing the X3 Services service onto my mzPRINT server.

Unzip “x3-services-29.0.42-win.zip” into “D:\Sage\x3services\” directory

Edit xtrem-security.yml

```

1  loginUrl: https://mzad:443/
2
3  # The following clientId and secret must be set with the same values in the syracuse nodelocal.js in the section
4  # Both this file and nodelocal.js must be kept safe with restricted access to admin only.
5  # exports.config = {
6  #   [...]
7  #   etna: {
8  #     security: {
9  #       clientId: "create-your-own-client-id-uuid",
10 #       secret: "change-to-use-a-strong-secret-for-your-client-id"
11 #     }
12 #   }
13 #   [...]
14 # };
15 syracuse:
16   clientId: ClientIDuuid
17   secret: VerySafeAndSecureSecretKey
18

```

NOTE: the URL used to connect to X3 by the user in the browser is passed to X3 Services to use in order to connect to the SOAP pool, rather than the “loginUrl” defined here.

As I am using HTTPS url, I must add the CA certificate to the service.js file in the ADC setup, if it is not an already known CA. In my case I need to add the following to “service.js”

```

1  const { Service } = require('node-windows');
2  const path = require('path');
3  const os = require('os');
4
5  const platform = `${os.platform()}-${os.arch()}`;
6
7  const errorCodes = {
8    badOperation: 1,
9    windowsOnly: 2,
10   cannotSetServiceFailure: 3,
11 };
12 const serviceConfig = {
13   name: 'Sage X3 Services',
14   description: 'Sage X3 Services',
15   script: path.join(__dirname, 'node_modules@sage/xtrem-cli/bin/xtrem'),
16   nodeOptions: ['--max_old_space_size=4096'],
17   workingDirectory: __dirname,
18   execPath: path.join(__dirname, 'nodejs', `${platform}`, 'node.exe'),
19   env: {
20     name: 'NODE_EXTRA_CA_CERTS',
21     value: 'D:\\SageSupport\\ADC_certs\\ADCCA.pem',
22   },
23 };
24

```

I then need to copy the CA certificate to the location specified.

These steps are described in KB article 112644 “ADC: Cannot read properties of undefined when creating records from ADC connected to HTTPS connection”

(Optional) Edit xtrem-config.yml to provide detailed logging

```

1 storage:
2   managedExternal: true
3 logs:
4   disabledForTestsx: false
5   domains:
6     sage/xtrem-x3-gateway/storage:
7       level: verbose
8     sage/xtrem-x3-gateway/web-service:
9       level: verbose
10    sage/xtrem-x3-sql-manager/sql:
11      level: verbose
12    sage/xtrem-service/service:
13      level: verbose
14    sage/xtrem-service/http:
15      level: verbose
16

```

Launch Powershell prompt using “Run as administrator”

Run “.\install.ps1”

```

PS D:\sage\x3services> .\install.ps1
Install complete.
Start complete.
PS D:\sage\x3services>

```

Check “Sage X3 Services” service is running

Name	Description	Status	Startup Type	Log On As
Remote Desktop Configuration	Remote Desktop Configuration service (RDCS) is responsible for all Remote Desk...	Running	Manual	Local System
Remote Desktop Services	Allows users to connect interactively to a remote computer. Remote Desktop and...	Running	Manual	Network Service
Remote Desktop Services UserMode Port Redirec...	Allows the redirection of Printers/Drives/Ports for RDP connections	Running	Manual	Local System
Remote Procedure Call (RPC)	The RPCSS service is the Service Control Manager for COM and DCOM servers. It ...	Running	Automatic	Network Service
Remote Procedure Call (RPC) Locator	In Windows 2003 and earlier versions of Windows, the Remote Procedure Call (RP...	Running	Manual	Network Service
Remote Registry	Enables remote users to modify registry settings on this computer. If this service i...	Running	Automatic (Trigg...	Local Service
Resultant Set of Policy Provider	Provides a network service that processes requests to simulate application of Gro...	Running	Manual	Local System
Routing and Remote Access	Offers routing services to businesses in local area and wide area network environ...	Running	Disabled	Local System
RPC Endpoint Mapper	Resolves RPC interfaces identifiers to transport endpoints. If this service is stoppe...	Running	Automatic	Network Service
Safe_X3_SrvEdt_V2_EDT_DEFAULT	Sage X3 PrintServer	Running	Automatic	EXAMPLE\x3run
Sage Safe X3 AdxAdmin		Running	Automatic	Local System
Sage X3 Services	Sage X3 Services	Running	Automatic	Local System
Secondary Log-on	Enables starting processes under alternate credentials. If this service is stopped, th...	Running	Manual	Local System
Secure Socket Tunneling Protocol Service	Provides support for the Secure Socket Tunneling Protocol (SSTP) to connect to r...	Running	Manual	Local Service
Security Accounts Manager	The status of this service signals other services that the Security Accounts Mana...	Running	Automatic	Local System

In Powershell run the command “invoke-webrequest <http://localhost:8240/ping>” which should return “StatusCode : 200”

```

PS D:\sage\x3services> invoke-webrequest http://localhost:8240/ping

StatusCode      : 200
StatusDescription : OK
Content         : {}
RawContent     : HTTP/1.1 200 OK
Content-Security-Policy: script-src 'unsafe-eval' 'unsafe-inline' 'self' data: https://cdn.pendo.io
https://pendo-io-static.storage.googleapis.com https://pendo-static-5069687185997824...
Forms          : {}
Headers       : {[Content-Security-Policy, script-src 'unsafe-eval' 'unsafe-inline' 'self' data: https://cdn.pendo.io
https://pendo-io-static.storage.googleapis.com https://pendo-static-5069687185997824.storage.googleapis.com https://app.pendo.io
https://pendo-static-5069687185997824.storage.googleapis.com https://data.pendo.io https://id-shadow.sage.com https://id.sage.com
https://bam.nr-data.net https://js-agent.newrelic.com https://eu.jsagent.tcell.insight.rapid7.com;img-src 'self' data: https://cdn.pendo.io
https://pendo-io-static.storage.googleapis.com https://pendo-static-5069687185997824.storage.googleapis.com https://app.pendo.io
https://pendo-static-5069687185997824.storage.googleapis.com https://data.pendo.io;style-src 'unsafe-inline' 'self' data: https://cdn.pendo.io
https://pendo-io-static.storage.googleapis.com https://pendo-static-5069687185997824.storage.googleapis.com https://app.pendo.io
https://fonts.sage.com;object-src 'self' blob;frame-src 'self' data: https://cdn.pendo.io https://pendo-io-static.storage.googleapis.com
https://pendo-static-5069687185997824.storage.googleapis.com https://app.pendo.io https://eu.browser.tcell.insight.rapid7.com https://mzad
blob;connect-src 'self' https://app.pendo.io https://data.pendo.io https://pendo-static-5069687185997824.storage.googleapis.com
https://bam.nr-data.net https://eu.browser.tcell.insight.rapid7.com https://mzad;child-src 'self' https://app.pendo.io;default-src
'self';base-uri 'self';block-all-mixed-content;font-src 'self' https://data;frame-ancestors 'self';script-src-attr 'none'],
[X-DNS-Prefetch-Control, off], [Expect-CT, max-age=0], [X-Frame-Options, SAMEORIGIN]...}
Images        : {}
InputFields   : {}
Links         : {}
ParsedHtml    : System.__ComObject
RawContentLength : 2

PS D:\sage\x3services>

```

Also confirm it works OK with the hostname in the URL “[invoke-webrequest http://mzPRINT:8240/ping](http://mzPRINT:8240/ping)”

Configure Syracuse nodelocal.js and restart Syracuse

On all three Syracuse servers, make the following change to nodelocal.js then restart Syracuse service

```

13  x3key: true,
14  collaboration: {
15    driver: "mongodb",
16    database: "syracuse",
17    hostname: "mzweb1",
18    port: 27017,
19    logpath: "d:\\Sage\\Syracuse\\syracuse\\logs",
20    certdir: "d:\\Sage\\Syracuse\\syracuse\\certs",
21    cacheDir: "d:\\Sage\\Syracuse\\syracuse\\cache"
22  },
23  mongodb: {
24    options: {
25      writeConcern: { w: 1 },
26      ssl: true,
27      sslCA: require('fs').readFileSync("d:\\Sage\\Syracuse\\syracuse\\certs\\mongodb\\ca.cacrt"),
28      sslValidate: true,
29      sslKey: require('fs').readFileSync("d:\\Sage\\Syracuse\\syracuse\\certs\\mongodb\\client.pem"),
30      sslCert: require('fs').readFileSync("d:\\Sage\\Syracuse\\syracuse\\certs\\mongodb\\client.pem")
31    }
32  },
33  session: {
34    timeout: 30, // minutes
35    checkInterval: 60, // seconds
36    auth: ["basic","basic"]
37  },
38  hosting: {
39    localBalancer: true,
40  },
41  searchEngine: {
42    hostname: "mzPrint",
43    port: 9200,
44  },
45  stna: {
46    security: {
47      clientId: "clientDaudd",
48      secret: "VerySafeAndSecureSecretKey"
49    },
50  },
51  x3fusion: {
52    records: {
53      "dumpPath": "d:\\Sage\\Syracuse\\syracuse\\cache\\_cvg__USERNAME_"
54    }
55  }
56  }
57  }

```

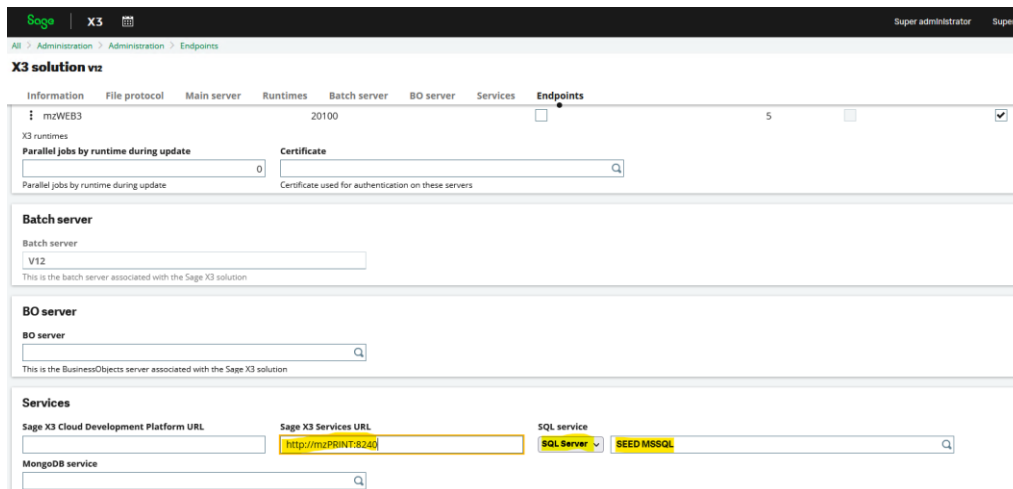
Configure SQL Server and X3 solution in X3 Administration screens

Navigate to Administration. Administration, Endpoints, MS SQL service

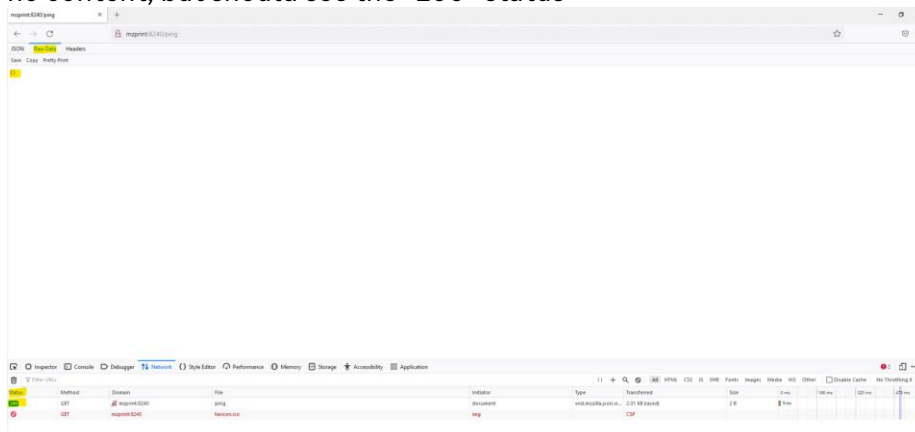
Actions: Create msSqlService



Navigate to Administration, Administration, Endpoints, X3 solution and edit the solution for the Sage X3 services URL and SQL Service



From each Syracuse server, check can access the Sage X3 Services URL from a browser. There is no content, but should see the “200” status



Republish Web services

Navigate to Development, Script dictionary, Scripts, Web services
Re-publish the web services AOWSIMPORT, CWSSCS, CWSSIS and CWSSST

Create/configure SOAP Pool

ADC will automatically use the first SOAP Pool in alphabetic order for the folder concerned.
Make sure your naming convention allows for this.

In my case, I do not have any web pools setup yet, so will call mine “ADC_SEED”

Navigate to Administration, Administration, Web services, Classic SOAP pools configuration

Actions: New SOAP pool

Restart Syracuse on MZWEB1, MZWEB2 and MZWEB3 to ensure they all pick up the changes and startup the Web Pool correctly.

Use SoapUI, Postman or the inbuilt SOAP tester to confirm you can execute a SOAP request using the SOAP pool

For example, use the following as a quick test that the SOAP pool is working OK.

URL:

<https://mzad/soap-generic/syracuse/collaboration/syracuse/CAdxWebServiceXmlCC>

Payload:

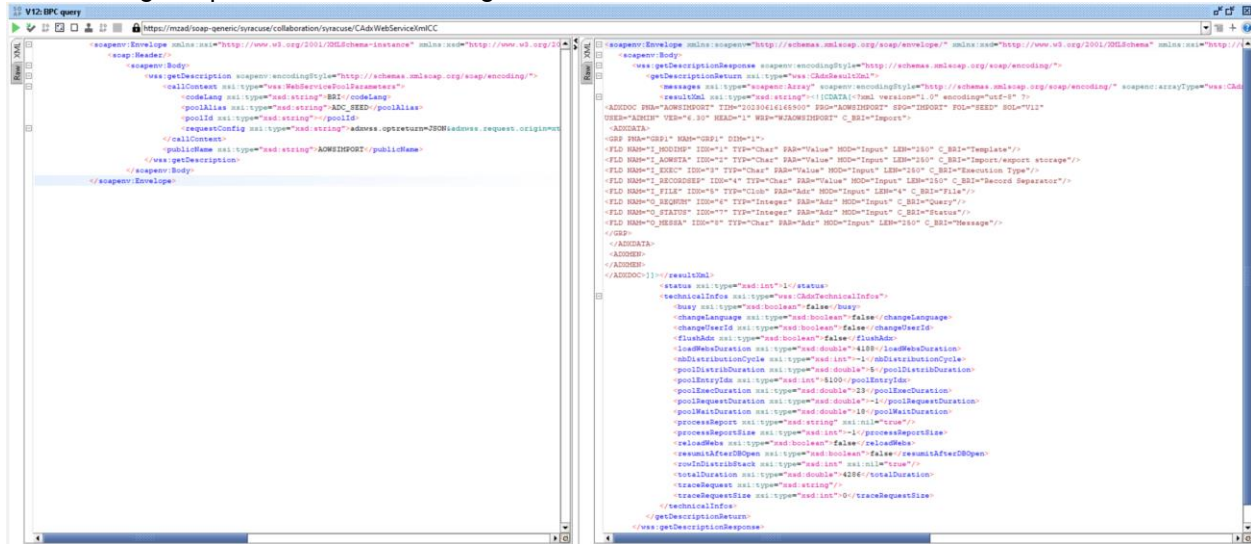
```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:wss="http://www.adonix.com/WSS"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/">
  <soap:Header/>
  <soapenv:Body>
    <wss:getDescription
soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <callContext xsi:type="wss:WebServicePoolParameters">
        <codeLang xsi:type="xsd:string">BRI</codeLang>
        <poolAlias xsi:type="xsd:string">ADC_SEED</poolAlias>
        <poolId xsi:type="xsd:string"></poolId>
      </callContext>
    </wss:getDescription>
  </soapenv:Body>
</soapenv:Envelope>
```

```

<requestConfig
xsi:type="xsd:string">adxwss.optreturn=JSON&adxwss.request.origin=xtreem</requestConfig>
</callContext>
<publicName xsi:type="xsd:string">AOWSIMPORT</publicName>
</wss:getDescription>
</soapenv:Body>
</soapenv:Envelope>

```

When using SoapUI I see the following results



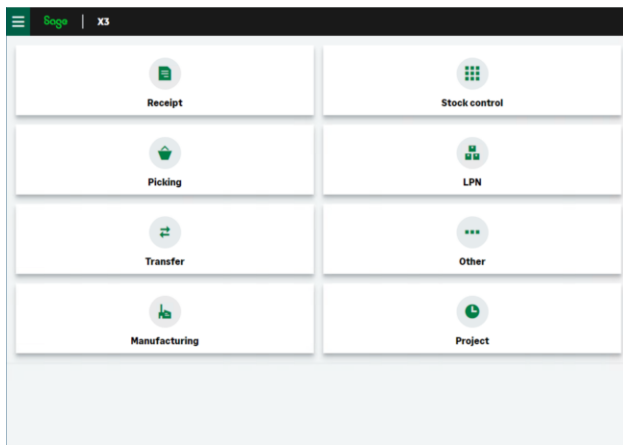
Testing

I am now ready to commence my testing to ensure everything works as expected.

Initial testing using browser

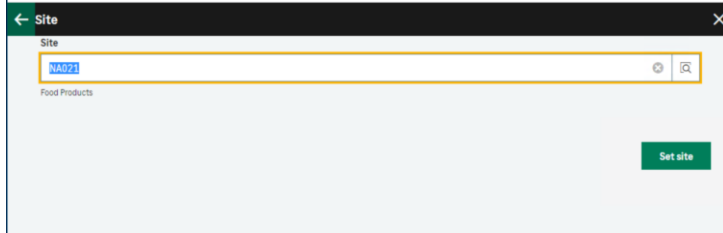
On a browser, navigate to <https://mzad/handheld/>

Login as “adctest” user who has the ADC badges assigned.

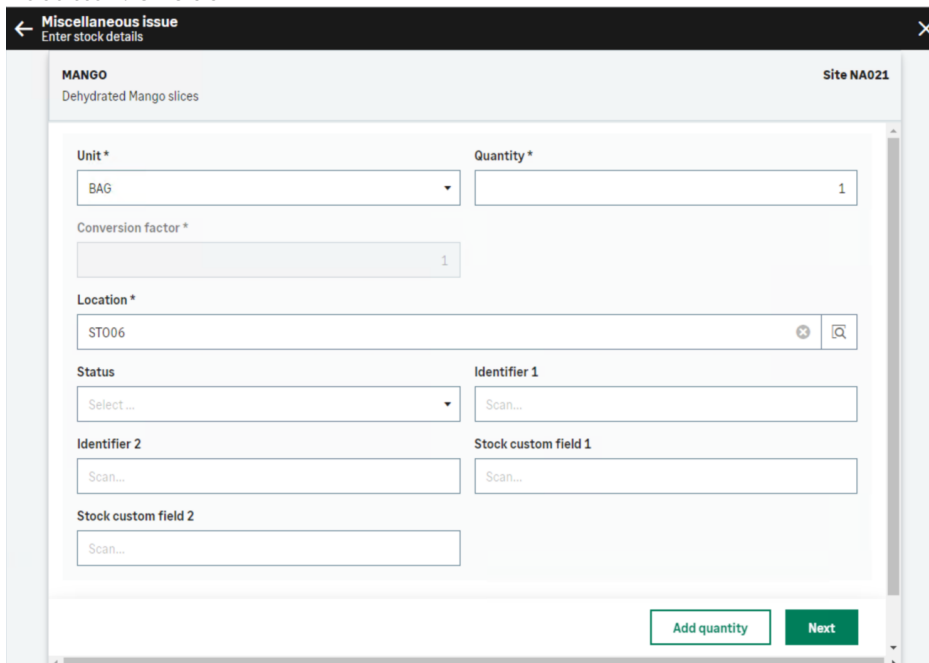


Select Settings, Site

Pick Site NA021 and “Set site”



Navigate to Other, Miscellaneous issue
Pick product MANGO
Unit: Bag
Quantity: 1
Location: ST006



NEXT

← Miscellaneous issue

Issue date * 16/06/2023 Site NA021

Transaction ALL

Product * Scan or select...

Products

Products: 1

MANGO Dehydrated Mango slices	1 BAG	
---	-------	--

Create

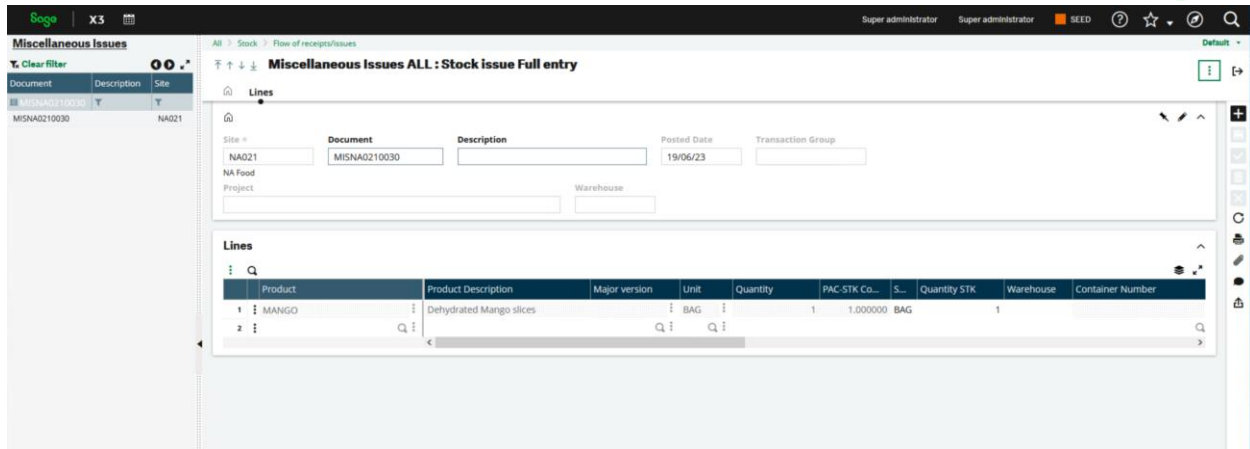
CREATE

Success

Document no. MISNA0210030 created. ({{VORNUMMISNA0210030}})

OK

Check I can see this transaction in front end



Additional testing with all makes/models of handheld devices that will be used
 It is important that all devices that will be used by the end users are tested to ensure compatibility.

Conclusion

This "Build diary" document provides the steps I took on my internal servers to install ADC.