



# SSD Deployer User Guide

---

## Document Revision History

---

Revision Date	Written/Edited By	Comments
September 2016	Andy Dunfee	Initial Release with SSD v2
February 2017	Paul Wheeler	Updated for SSD v3
June 2017	Paul Wheeler	Updated for SSD v4
December 2017	Paul Wheeler	Updated for SSD v5

© Copyright 2017 SailPoint Technologies, Inc., All Rights Reserved.

SailPoint Technologies, Inc. makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. SailPoint Technologies shall not be liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

**Restricted Rights Legend.** All rights are reserved. No part of this document may be photocopied, reproduced, or translated to another language without the prior written consent of SailPoint Technologies. The information contained in this document is subject to change without notice.

Use, duplication or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 for DOD agencies, and subparagraphs (c) (1) and (c) (2) of the Commercial Computer Software Restricted Rights clause at FAR 52.227-19 for other agencies.

**Regulatory/Export Compliance.** The export and reexport of this software is controlled for export purposes by the U.S. Government. By accepting this software and/or documentation, licensee agrees to comply with all U.S. and foreign export laws and regulations as they relate to software and related documentation. Licensee will not export or reexport outside the United States software or documentation, whether directly or indirectly, to any Prohibited Party and will not cause, approve or otherwise intentionally facilitate others in so doing. A Prohibited Party includes: a party in a U.S. embargoed country or country the United States has named as a supporter of international terrorism; a party involved in proliferation; a party identified by the U.S. Government as a Denied Party; a party named on the U.S. Government's Entities List; a party prohibited from participation in export or reexport transactions by a U.S. Government General Order; a party listed by the U.S. Government's Office of Foreign Assets Control as ineligible to participate in transactions subject to U.S. jurisdiction; or any party that licensee knows or has reason to know has violated or plans to violate U.S. or foreign export laws or regulations. Licensee shall ensure that each of its software users complies with U.S. and foreign export laws and regulations as they relate to software and related documentation.

**Trademark Notices.** Copyright © 2017 SailPoint Technologies, Inc. All rights reserved. SailPoint, the SailPoint logo, SailPoint IdentityIQ, and SailPoint Identity Analyzer are trademarks of SailPoint Technologies, Inc. and may not be used without the prior express written permission of SailPoint Technologies, Inc. All other trademarks shown herein are owned by the respective companies or persons indicated.

## Table of Contents

Overview .....	4
Beginning State.....	5
Run the Deployment Tool.....	6
Go to SSD Project .....	6
Run Tool and Answer Prompts .....	6
Deployment Output.....	11
End State .....	15

## Overview

The Services Standard Deployment (SSD) is a set of tools and artifacts designed to increase the ease of installation and add a level of standardization across IdentityIQ deployments. These artifacts and tools are used by the SailPoint Services team on projects deploying IdentityIQ.

The SSD includes a broad set of resources to help you and your team with the process of deploying IdentityIQ. The SSD package can be thought of as a "kit", containing different important parts of the deployment process.

After starting a project and downloading and unpacking the latest SSD zip file, there are a handful of initial steps required to setup different parts of the SSD, such as creating new target, ignorefiles, and iiq properties files; merging SSF-specific properties into these new files; copying the configurable SSF and SST XML files into their proper folders; and enabling certain features in the build.properties file.

While all documentation related to the SSD should be consulted, especially the SSB and SSF User Guides, and while there will still be manual installation and setup steps, a command-line Deployer tool has been developed to streamline some of the initial deployment steps. This document covers this command-line tool, providing instructions on how to use it and what it does.

The tool can be run on Windows and UNIX-based operating systems. It mostly uses y|n prompts, with a few other entry points, and takes about a minute to run.

Some of the tools that ship with the SSD are not deployed by default with the build, and you can use the Deployer tool to enable them for deployment; others are deployed by default and do not appear as options for deployment in the Deployer. For example, the SSF Features and Frameworks are not deployed by default and can be enabled in the Deployer, but the XML Object Exporter is deployed by default with the SSD and does not have an option for deployment in the Deployer. Generally, the components that are available for deployment in the Deployer often require extra setup steps, or they change some element of IdentityIQ's behavior or appearance which might be unexpected to an implementer who is unfamiliar with the tools, so they are left for the implementer to specifically select. The components that are deployed by default are usually tools that will not cause any unexpected behavior to the general out-of-the-box functionality of IdentityIQ. You are strongly advised to read the documentation of each component that you deploy with the Deployer tool. All documentation can be found under the "doc" folder at the root of the SSD.

## Beginning State

After downloading and unpacking the SSD and before running the SSD Deployment Tool, the Configuration, Custom, Form, and RuleLibrary folders under /config will all be empty. The following properties in the build.properties file will be set to false:

- deploySSF
- deploySSTUnitTester
- deployEmailApprovals
- deployBreakGlass
- deployEmailTemplateEditor
- deployEnhancedEmailNotifier
- deployLCMAAttachments
- deployPasswordExpirationReminder
- deployTaskResultter
- deployLogLevelManager

Each of these settings affect whether or not a corresponding component will be deployed.

The root folder will contain none of your environment-specific target, ignorefiles or iiq properties files and the sample sandbox.ignorefiles.properties and sandbox.target.properties will not contain the ssf.ignorefiles.properties or ssf.target.properties values, respectively. Running the tool will change that.

TestPath	TestPath	TestPath
<ul style="list-style-type: none"> <li>TestPath           <ul style="list-style-type: none"> <li>base</li> <li>config               <ul style="list-style-type: none"> <li>Application</li> <li>Configuration</li> <li>Custom</li> <li>Form</li> <li>LocalizedAttribute</li> <li>Rule</li> <li>RuleLibrary</li> <li>SSF_Features</li> <li>SSF_Frameworks</li> <li>SSF_Tools</li> <li>SSP_Tools</li> <li>TaskDefinition</li> <li>TaskSchedule</li> </ul> </li> <li>db</li> <li>doc</li> <li>lib</li> <li>scripts</li> <li>servicetools</li> <li>src</li> <li>web</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>base</li> <li>config</li> <li>db</li> <li>doc</li> <li>lib</li> <li>scripts</li> <li>servicetools</li> <li>src</li> <li>web           <ul style="list-style-type: none"> <li>build.bat</li> <li>build.properties</li> <li>build.sh</li> <li>build.xml</li> <li>deploySSD</li> <li>deploySSD.bat</li> <li>ExportScript.txt</li> <li>Release-Notes-ServicesStandardD</li> <li>sandbox.ignorefiles.properties</li> <li>sandbox.iiq.properties</li> <li>sandbox.target.properties</li> <li>servers.properties</li> <li>ServicesStdBuild.iml</li> <li>ServicesStdBuild.ipr</li> <li>ServicesStdBuild.iws</li> <li>SSDDeployer.jar</li> <li>ssf.ignorefiles.properties</li> <li>ssf.target.properties</li> </ul> </li> </ul>	<pre> # Deploy SSF Frameworks and Features deploySSF=false  # Deploy SSF Tools deployGenericImporter=true deployObjectExporter=true deployMultiThreadedBeanShellKit=true deployReportExtractor=true deployRuleRunnerTask=true deployEmailApprovals=false deployBreakGlass=false deployCorrelationMigrator=true deployWorkgroupMigrator=true deployEmailTemplateEditor=false deployIIQLoopbackConnector=true deployEnhancedEmailNotifier=false deployExclusionFramework=true deployLCMAAttachments=false deployServerSpecificTaskLauncher=true deployPasswordExpirationReminder=false deployTaskResultter=false deployLogLevelManager=false  # Deploy SST Testing Tools deploySSTUnitTester=false  # Deploy SSP Performance Tools deploySSPTools=true           </pre>

## Run the Deployment Tool

### Go to SSD Project

Open a terminal or command prompt and cd to the root folder of your unzipped SSD project. Validate the contents:

- /config/SSF\_Features
- /config/SSF\_Frameworks
- /config/SSF\_Tools
- build.properties
- ssf.ignorefiles.properties
- ssf.target.properties
- SSDDeployer.jar
- deploySSD
- deploySSD.bat

```
Last login: Tue Aug 30 10:35:51 on ttys001
adunfee-mbpr:~ andy.dunfee$ pwd
/Users/andy.dunfee
adunfee-mbpr:~ andy.dunfee$ cd /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath
adunfee-mbpr:TestPath andy.dunfee$ ls
ExportScript.txt          config                    sandbox.target.properties
Release-Notes-ServicesStandardDeployment.txt  db                       scripts
SSDDeployer.jar           deploySSD                servers.properties
base                      deploySSD.bat            servicetools
build.bat                 doc                       src
build.properties         lib                       ssf.ignorefiles.properties
build.sh                  sandbox.ignorefiles.properties
build.xml                 sandbox.iig.properties  ssf.target.properties
                           web
```

Figure 1 - cd to SSD project and view contents

### Run Tool and Answer Prompts

The following details the prompts and inputs required, most prompts simply requiring a “y” or an “n” response. The input process includes the following steps:

1. Start the tool (using the deploySSD or deploySSD.bat script at the root of the SSD)
2. Validate or enter the SSD path
3. Enter environments
4. Enable individual components
5. Review summary and start the deployment

```
adunfee-mbpr:TestPath andy.dunfee$ ./deploySSD
*****
* Welcome to the SSD Deployer Tool. The tool should take only a minute or two to run.
* It will ask some simple questions to help setup your SSD project. It will help to
* initialize some build properties, setup your target and ignore files properties, and
* copy SSF and SST components into the correct config folders.
* If at any point you want to exit, enter x.
*****
Enter any value and hit enter to begin or x and enter to exit.
```

Figure 2 - Start the tool (deploySSD on UNIX or deploySSD.bat on Windows)

```
The tool must first determine the path to your SSD project.

The detected path is: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath

Is the detected path correct? [y|n]
y

Validating path: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath
Check for file/folder: build.properties
Check for file/folder: ssf.target.properties
Check for file/folder: ssf.ignorefiles.properties
Check for file/folder: config/SSF_Features
Check for file/folder: config/SSF_Frameworks
Check for file/folder: config/SSF_Tools
The path has been validated.
```

Figure 3 - Validate or enter the path

```
*****
* Each environment will have a unique set of target, ignorefiles, and iiq properties.
* The next steps will help to build these for your various environments/targets.
* The SPTARGET environment variable is often used for local sandboxes. The tool will
* first attempt to detect this variable. If it is found, you will have the option to
* let the tool create the properties file for your local environment based on the
* SPTARGET value. Afterward, you will have the ability to enter additional targets
* and let the tool create the properties files for each one entered.
*****

The SPTARGET environment variable is set to andybox.

Would you like to create the properties files for this SPTARGET value? [y|n]
y

Would you like to create property files for additional targets? [y|n]
y

Enter the addiitonal targets. Separate with commas. Whitespace will be removed.
dev,qa,prod
```

Figure 4 - Setup environment property files



```
*****
* The SSF Features and Frameworks is a package of reusable IdentityIQ configurations
* and code used to speed up and simplify the deployment process on IdentityIQ
* implementations. It does this by abstracting and reusing common architectures and
* best practices, eliminating 'nuts and bolts' development and allowing implementers
* to focus on the customer requirements.
* The next steps will allow you to enable the package and pick individual features and
* frameworks to turn on. Turning on a framework/feature simply means copying the
* configurable XML components to their correct file locations in the SSB.
*****

Would you like use the SSF Features and Frameworks? [y|n]
y

The SSF includes a number of features--Attribute Synch, Joiner, Mover, Leaver, Rehire,
Terminate Identity--and frameworks--Field Value, Approvals, Role Assignment,
Provision Processor and Send Emails. By default, if you turn on the SSF, the Approvals,
Role Assignment, Provision Processor and Send Emails frameworks are made available.
The configurable configuration objects for the Field Value framework and all features
can be deployed selectively or all can be deployed at once.

Would you like to deploy all frameworks and features? [y|n]
n
You have chosen to select features and frameworks individually.
Would you like to use the Attribute Synch Feature? [y|n]
y
Would you like to use the Joiner Feature? [y|n]
y
Would you like to use the Leaver Feature? [y|n]
y
Would you like to use the Rehire Feature? [y|n]
y
Would you like to use the Mover Feature? [y|n]
n
Would you like to use the Terminate Identity Feature? [y|n]
n
Would you like to use the Field Value Framework? [y|n]
y
```

Figure 5 – Enable the SSF (Features and Frameworks)

```
*****
* The Services Standard Test (SST) Unit Tester is a framework for writing and
* registering unit test rules with unit test argument files. It facilitates better
* testing of atomic units of code, such as methods in a rule library. For each unit
* test, you write a unit test rule and any number of arguments files, each argument
* file representing a different use case or set of input parameters. Then you can
* register these in a mapping object and run all unit tests from either the command
* line, a quick link, or a task.
*****

Would you like use the SST Unit Tester? [y|n]
y
```

Figure 6 - Enable SST Unit Tester



```
*****
* Break Glass Access Request is a simple use case solution intended to provide a
* way for selected users to bypass the usual approval process for access requests
* in emergency situations where the need for access cannot wait for the approval
* process to complete. After following the usual access request procedure, the
* user is presented with a form in which they must mark a checkbox to acknowledge
* use of the Break Glass procedure with no approvals; if this is not done, the
* request will be treated as a normal access request with approvals. The
* configured Security Officer will receive notification of all Break Glass requests.
* The ability to use this option is granted by membership of a 'Break Glass Users'
* workgroup.
*****

Would you like use the BreakGlass Request Tool? [y|n]
```

Figure 7 - Enable Break Glass Access Request

```
*****
* Email Approvals is a tool that allows approvers to approve or reject work items
* using email rather than accessing the UI. Requires an email server that accepts
* POP3 connections.
*****

Would you like use the Email Approvals Tool? [y|n]
```

Figure 8 - Enable Email Approvals

```
*****
* The Email Template Editor is a tool for creating and modifying both plain
* text and HTML email templates. It includes a simple HTML editing control
* that allows you to design your template in a WYSIWYG interface with no need
* to edit HTML code. Users must be members of the 'Email Template Admins'
* workgroup to use this tool.
*****

Would you like use the Email Template Editor? [y|n]
```

Figure 9 - Enable Email Template Editor

```
*****
* The Enhanced Email Notifier provides configurable options for managing the
* sending and content of email notifications through IdentityIQ. It allows the
* use of different email templates depending on the value of an identity
* attribute, such as language or country. It also provides a way to merge
* multiple emails for the same person into a single summary email to help
* reduce the number of emails users receive from IdentityIQ.
*****

Would you like use the Enhanced Email Notifier? [y|n]
```

Figure 10 - Enable Enhanced Email Notifier

```
*****
* Access Request Attachments provides the ability for a requester to add
* attachments to an access request to help to justify the reason for the
* request or provide additional information. These attachments are then
* available for the approver to view in order to assist with the approval
* decision. This is available for IdentityIQ versions 7.0 and higher but can
* be made to work with earlier versions with some manual configuration: please
* refer to the documentation.
*****

Would you like use Access Request Attachments? [y|n]
```

Figure 11 - Enable Access Request Attachments

```
*****
* Password Expiration Reminder provides a mechanism to send notifications to
* users through IdentityIQ when the passwords on their accounts in integrated
* target systems are about to expire. It is also possible to configure steps
* to be taken when an account password has expired.
*****

Would you like use Password Expiration Reminder? [y|n]
```

Figure 12 - Enable Password Expiration Reminder

```
*****
* The Task Resultter provides a simple screen to view the results of account
* aggregation tasks, including information unavailable in the main Task
* Results page, such as aggregation duration and speed (in records per
* second). It is useful for measuring performance of your account
* aggregations. The Task Resultter can only be used with the Plugin Framework
* in IdentityIQ version 7.1 and above.
*****

Would you like use the Task Resultter? [y|n]
```

Figure 13 – Enable Task Resultter

```
*****
* The Log Level Manager provides an easy-to-use administrative screen for
* modifying the Log4j log levels of Java class loggers in IdentityIQ when
* troubleshooting issues. The tool simplifies configuration of logger levels
* and can remove the need to manually edit the log4j.properties file. The
* Log Level Manager can only be used with the Plugin Framework in IdentityIQ
* version 7.1 and above.
*****

Would you like use the Log Level Manager? [y|n]
```

Figure 14 – Enable Log Level Manager

```
*****
* SUMMARY
*****

You have chosen to deploy the SSF. This means all target and ignorefiles properties will
be updated with ssf-specific properties, basic frameworks will be made available, and the
configurable configuration objects for any specific framework or feature you've chosen to
deploy will be copied into their respective config folders, usually /config/Custom and
/config/RuleLibrary, build.properties file will be updated with deploySSF=true.
You have chosen to deploy the following frameworks and features:

    Attribute Synch
    Leaver
    Field Value
    Joiner
    Rehire

You have chosen to deploy the SST Unit Tester. This means all target and ignorefiles
properties will be updated with ssf-specific properties, the unit test custom mapping
file will be copied into the /config/Custom folder, and the ZUnitTests folder will be
copied into the /config folder. It'll create an empty UTReports folder under /web,
build.properties file will be updated with deploySSTUnitTester=true.

You have chosen to initialize iiq, target, and ignorefiles properties for one or more
target environments. The files will be created for the following:
    Target: dev

*****

Are you ready to deploy? [y|n]
```

Figure 15 - Review summary, start deployment

## Deployment Output

The deployment process will write out to the terminal everything it is doing to setup the SSD. The following shows the detailed output of the deployment process:

```
Begin deployment.

*****

Updating the build.properties
Get file at path: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/build.properties
Turn on the deploySSF line
Turn on the deploySSTUnitTester line
Finished updating the build.properties

*****
```

Figure 16 - Update the build.properties file

```
*****

Create any environment property files
Creating props for: andybox
Creating any missing files for : andybox
Created file: andybox.target.properties
Created file: andybox.ignorefiles.properties
Created file: andybox.iiq.properties
Creating props for: dev
Creating any missing files for : dev
Created file: dev.target.properties
Created file: dev.ignorefiles.properties
Created file: dev.iiq.properties
Creating props for: qa
Creating any missing files for : qa
Created file: qa.target.properties
Created file: qa.ignorefiles.properties
Created file: qa.iiq.properties
Creating props for: prod
Creating any missing files for : prod
Created file: prod.target.properties
Created file: prod.ignorefiles.properties
Created file: prod.iiq.properties

*****
```

Figure 17 - Create various host/environment property files



```

*****
Updating the target.properties
Get file at path: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/ssf.target.properties
andybox.target.properties is a target props file. Append the ssf props
dev.target.properties is a target props file. Append the ssf props
prod.target.properties is a target props file. Append the ssf props
qa.target.properties is a target props file. Append the ssf props
sandbox.target.properties is a target props file. Append the ssf props
Finished updating the target.properties

*****

*****

Updating the ignorefiles.properties
Get file at path: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/ssf.ignorefiles.properties
andybox.ignorefiles.properties is an ignorefiles props file. Append the ssf props
dev.ignorefiles.properties is an ignorefiles props file. Append the ssf props
prod.ignorefiles.properties is an ignorefiles props file. Append the ssf props
qa.ignorefiles.properties is an ignorefiles props file. Append the ssf props
sandbox.ignorefiles.properties is an ignorefiles props file. Append the ssf props
Finished updating the ignorefiles.properties

*****

```

Figure 18 - Copy the ssf properties

```

*****
Copying over all the various SSF components
*****

Copy the common loc library
Copy from: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/SSF_Features/Common/Configure/SPCONF_LCE_Common_RulesLibrary.xml
to: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/RuleLibrary/SPCONF_LCE_Common_RulesLibrary.xml
*****

Copy provision processor email text object
Copy from: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/SSF_Frameworks/ProvisionProcessor/Configure/SPCONF_EmailText_Custom.xml
to: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/Custom/SPCONF_EmailText_Custom.xml
*****

Copy the approval framework files
Copy from: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/SSF_Frameworks/Approvals/Configure/SPCONF_ApprovalFramework_Custom_RuleLibrary.xml
to: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/RuleLibrary/SPCONF_ApprovalFramework_Custom_RuleLibrary.xml
Copy from: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/SSF_Frameworks/Approvals/Configure/SPCONF_ApprovalObjectMappings_Custom.xml
to: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/Custom/SPCONF_ApprovalObjectMappings_Custom.xml
Copy from: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/SSF_Frameworks/Approvals/Configure/SPCONF_ElectronicSignature_Configuration.xml
to: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/Configuration/SPCONF_ElectronicSignature_Configuration.xml
*****

```

Figure 19 - Copy common SSF components

```

*****
Loop the remaining features

*****

Migrate the attr synch components
Copy from: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/SSF_Features/AttributeSynch/Configure/SPCONF_AttrSynch_RulesLibrary.xml
to: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/RuleLibrary/SPCONF_AttrSynch_RulesLibrary.xml
Copy from: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/SSF_Features/AttributeSynch/Configure/SPCONF_AttrSynch_Mappings_Custom.xml
to: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/Custom/SPCONF_AttrSynch_Mappings_Custom.xml

*****

Migrate the field value components
Copy from: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/SSF_Frameworks/FieldValue/Configure/SPCONF_FieldsValue_RulesLibrary.xml
to: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/RuleLibrary/SPCONF_FieldsValue_RulesLibrary.xml

*****

Migrate the joiner components
Copy from: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/SSF_Features/Joiner/Configure/SPCONF_Joiner_RulesLibrary.xml
to: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/RuleLibrary/SPCONF_Joiner_RulesLibrary.xml
Copy from: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/SSF_Features/Joiner/Configure/SPCONF_Joiner_Mappings_Custom.xml
to: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/Custom/SPCONF_Joiner_Mappings_Custom.xml

*****

Migrate the leaver components
Copy from: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/SSF_Features/Leaver/Configure/SPCONF_Leaver_RulesLibrary.xml
to: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/RuleLibrary/SPCONF_Leaver_RulesLibrary.xml
Copy from: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/SSF_Features/Leaver/Configure/SPCONF_Leaver_Mappings_Custom.xml
to: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/Custom/SPCONF_Leaver_Mappings_Custom.xml

*****

Finished copying over the SSF components

*****

```

Figure 20 - Copy specific SSF components

```

*****

Copying over all the various Unit Test components
Copy the unit test custom object
Copy from: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/SSF_Tools/SST_UnitTester/Configure/SST_UnitTest_Mappings_Custom.xml
to: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/Custom/SST_UnitTest_Mappings_Custom.xml
Copy the ZUnitTests folder
Source: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/SSF_Tools/SST_UnitTester/Configure/ZUnitTests
Destination: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/ZUnitTests
Make directory: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/ZUnitTests
Make directory: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/ZUnitTests/Args
Make directory: /Users/andy.dunfee/Documents/workspace/SSDDeployer/TestPath/config/ZUnitTests/Rules
Finished copying over the unit test objects

*****

You have successfully completed the deployment process! You can begin to configure your project.

```

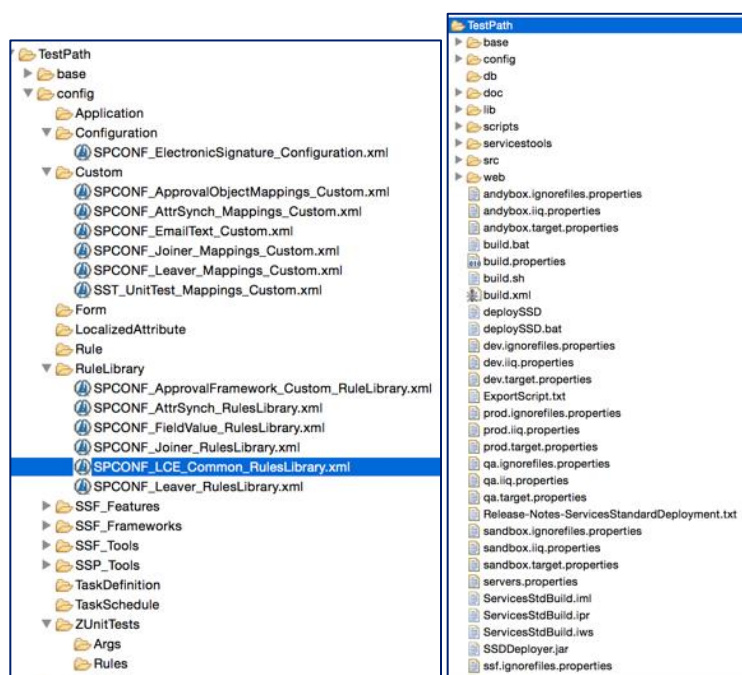
Figure 21 - Copy SST Unit Tester components



## End State

The following shows the updated package after running the SSD Deployment Tool. The following changes have been made:

- The Configuration, Custom, RuleLibrary folders under /config have been populated (if the SSF Features and Frameworks were selected for deployment)
- For each host environment entered, a new iiq.properties, target.properties, and ignorefiles.properties file has been created
- The target.properties and ignorefiles.properties have been copied into each new respective file for each component selected
- The build.properties has been updated, setting the deploy options for each component to true or false depending on which components were selected for deployment.



```
# Deploy SSF Frameworks and Features
deploySSF=true
```

```
# Deploy SSF Tools
deployGenericImporter=true
deployObjectExporter=true
deployMultiThreadedBeanShellKit=true
deployReportExtractor=true
deployRuleRunnerTask=true
deployEmailApprovals=true
deployBreakGlass=true
deployCorrelationMigrator=true
deployWorkgroupMigrator=true
deployEmailTemplateEditor=true
deployIIQoolbackConnector=true
deployEnhancedEmailNotifier=true
deployExclusionFramework=true
deployLCMAAttachments=true
deployServerSpecificTaskLauncher=true
deployPasswordExpirationReminder=true
deployTaskResulter=true
deployLogLevelManager=true
```

```
# Deploy SST Testing Tools
deploySSTUnitTester=true
```

```
# Deploy SSP Performance Tools
deploySSPTools=true
```

