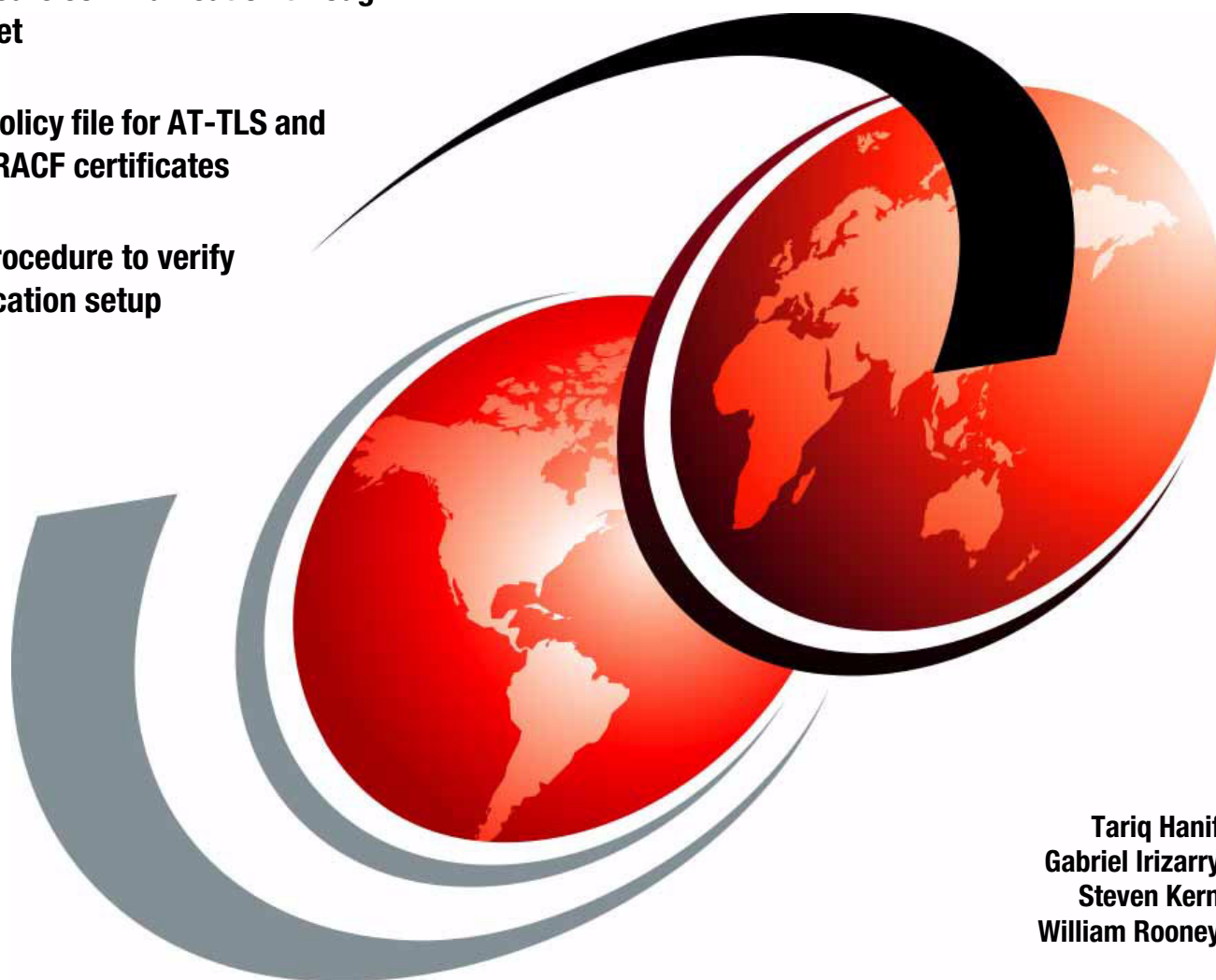


Securing IBM HyperSwap and IBM Tivoli Storage Productivity Center for Replication Communication Using AT-TLS

Enable secure communication through the Internet

Create a policy file for AT-TLS and generate RACF certificates

See the procedure to verify communication setup



Tariq Hanif
Gabriel Irizarry
Steven Kern
William Rooney



International Technical Support Organization

**Secure IBM HyperSwap and Tivoli Storage Productivity
Center for Replication Communication with AT-TLS**

July 2014

Note: Before using this information and the product it supports, read the information in “Notices” on page v.

First Edition (July 2014)

This edition applies to Version 5, Release 2 of Tivoli Storage Productivity Center (5608-PC1, 5608-PC2).

© Copyright International Business Machines Corporation 2014. All rights reserved.

Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Notices	v
Trademarks	vi
Preface	vii
Authors	vii
Now you can become a published author, too!	viii
Comments welcome	viii
Stay connected to IBM Redbooks	ix
Chapter 1. Introduction	1
1.1 Configuration overview	2
1.2 TLS and SSL basics	2
Chapter 2. Generating certificates using RACF	5
2.1 CLIST	6
2.1.1 CERTCRE8.sample	6
2.1.2 CERTIMPT.sample	7
2.1.3 Considerations: Tivoli Storage Productivity Center for Replication not on z/OS	7
2.2 Finding the user ID for HyperSwap	8
2.2.1 Using RACF to determine the user ID	8
2.2.2 Using System Display and Search Facility (SDSF)	10
2.3 Generating a self-signed certificate	11
2.3.1 Refreshing to ensure the certificate is in storage	11
2.3.2 Exporting the certificate to a data set	11
2.4 Creating a certificate for the server using your user ID	12
2.4.1 RACDCERT command	12
2.4.2 Creating a key ring for the certificates	12
2.4.3 Connecting the server certificate to the new key ring	12
2.4.4 Connecting the CA certificate to the new key ring	13
2.4.5 Giving the SETUP user permission to read its own key ring	13
2.5 Commands that are useful for validation	13
2.5.1 RACDCERT LIST command	13
2.5.2 RACDCERT LISTRING command	13
Chapter 3. Importing the certificate into a Java keystore file	15
3.1 Download the certificate	16
3.2 Download and install IBM JRE that includes iKeyman	16
3.3 Create a JKS file and import the CA into the JKS	17
3.4 Connect multiple z/OS Systems to the same Tivoli Storage Productivity Center for Replication	21
3.5 Configure replication properties and trust files for Tivoli Storage Productivity Center for Replication	21
Chapter 4. Creating a policy file for AT-TLS using the Configuration Assistant GUI ..	23
4.1 Download and install IBM Configuration Assistant V1R13	24
4.2 Create an AT-TLS policy file with IBM Configuration Assistant	24
4.3 Refresh AT-TLS settings	37
Chapter 5. Verifying the system	39

Related publications	43
IBM Redbooks	43
Other publications	43
Online resources	43
Help from IBM	44

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.


COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com) are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at <http://www.ibm.com/legal/copytrade.shtml>

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

AIX®	RACF®	Tivoli®
GDPS®	Redbooks®	WebSphere®
HyperSwap®	Redpaper™	z/OS®
IBM®	Redbooks (logo)  ®	
Parallel Sysplex®	System z®	

The following terms are trademarks of other companies:

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java, and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

Preface

IBM® Tivoli® Storage Productivity Center for Replication V5.2 can establish a connection to an IBM z/OS® server from a Tivoli Storage Productivity Center for Replication distributed installation or from another z/OS installation that can reside outside the sysplex that is being managed. This IBM Redpaper™ publication describes the steps to connect to, configure, and manage z/OS IBM HyperSwap® from Tivoli Storage Productivity Center for Replication V5.2.

This paper helps you configure IBM HyperSwap to communicate with IBM Tivoli Storage Productivity Center for Replication securely through the Internet by using Secure Sockets Layer (SSL) or Application Transparent Transport Layer Security (AT-TLS).

This document is intended for storage administrators responsible for configuring and maintaining the Tivoli Storage Productivity Center for Replication environment.

Authors

This paper was produced by a team of specialists from around the world working at the International Technical Support Organization, Poughkeepsie Center.

Tariq Hanif is a Senior Software Engineer in the z/OS System Verification Test group. He received his PhD degree from Kings College, University of London, UK in 1995. During his research at Kings, he worked on algorithm-based architecture, signal processing, and parallel processing architecture for image-processing algorithms. After his PhD, he moved to Canada where he worked for Royal Bank of Canada as Software Analyst. In 2000, he joined Nortel Networks Inc. in the US as a Software Engineer, where his focus was to develop efficient embedded signal processing software for VoIP gateways. He has several publications on Specialized Computer Architecture for Computer Vision, and Parallel Processing. In 2003, he joined the IBM z/OS System Verification group where he tests the z/OS operating system, with a focus on z/OS HyperSwap and Tivoli Storage Productivity Center for Replication for IBM System z®. In z/OS, his areas of interest are disaster recovery, IBM GDPS®, and Parallel Sysplex®.

Gabriel Irizarry is a Master's student at the Georgia Institute of Technology, where he is studying Computer Science with a specialization in social computing. He obtained a Bachelor of Science degree in Computer Engineering from the University of Puerto Rico at Mayagüez. During his studies, he has twice worked as an intern with IBM, once at the Linux Technology Center in Austin, Texas and then at IBM Poughkeepsie. At IBM Poughkeepsie, Gabriel worked on the zSeries System test department. His interests include Linux and free and open-source software, social systems, web development, and systems programming.

Steven Kern is an Advisory Software Engineer in the Cloud and Smarter Infrastructure organization in Tucson, Arizona. He has over seven years of experience with IBM in Storage Software. He is currently a Software Developer on Tivoli Storage Productivity Center for Replication. His areas of expertise include storage replication and z/OS HyperSwap.

William Rooney is a Senior Technical Staff Member in the System z Operating Systems Development organization in Poughkeepsie, New York. He has over 35 years of experience with IBM in System z and z/OS. He is currently the Software Architect for z/OS HyperSwap. His areas of expertise include I/O configuration, I/O qualities of service, and storage replication.

Thanks to the following people for their contributions to this project:

Tien Nguyen
Andrew Tracy
IBM Software Group

Michael Fitzpatrick
Nicholas Jones
Peter McCutcheon
Bruce Wells
IBM Systems and Technology Group

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our papers to be as helpful as possible. Send us your comments about this paper or other IBM Redbooks® publications in one of the following ways:

- ▶ Use the online **Contact us** review Redbooks form found at:

ibm.com/redbooks

- ▶ Send your comments in an email to:

redbooks@us.ibm.com

- ▶ Mail your comments to:

IBM Corporation, International Technical Support Organization
Dept. HYTD Mail Station P099
2455 South Road
Poughkeepsie, NY 12601-5400

Stay connected to IBM Redbooks

- ▶ Find us on Facebook:
<http://www.facebook.com/IBMRedbooks>
- ▶ Follow us on Twitter:
<http://twitter.com/ibmredbooks>
- ▶ Look for us on LinkedIn:
<http://www.linkedin.com/groups?home=&gid=2130806>
- ▶ Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:
<https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm>
- ▶ Stay current on recent Redbooks publications with RSS Feeds:
<http://www.redbooks.ibm.com/rss.html>



Introduction

This chapter describes a configuration that enables IBM HyperSwap to communicate with IBM Tivoli Storage Productivity Center for Replication securely across the Internet by using Secure Sockets Layer (SSL) or Transport Layer Security (TLS).

1.1 Configuration overview

HyperSwap is not aware of Secure Sockets Layer (SSL) or Transport Layer Security (TSL) and does not support it. However, IBM Communications Server's Application Transparent TLS (AT-TLS) provides SSL and TLS support for IBM z/OS applications that are not aware of SSL or TLS. The configuration consists mainly of creating certificates for SSL or TLS with IBM Resource Access Control Facility (RACF®) and configuring AT-TLS by creating an AT-TLS policy file. Figure 1-1 shows the configuration.

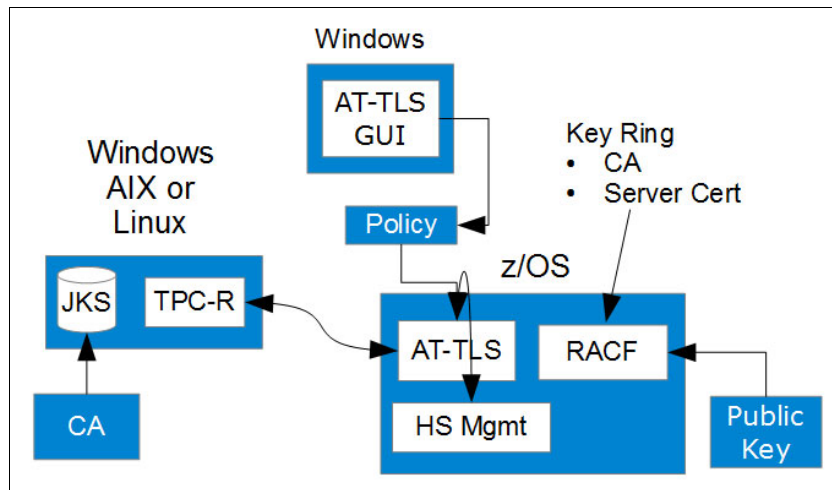


Figure 1-1 System diagram

1.2 TLS and SSL basics

TLS and SSL are both protocols that provide security to services accessed over a network (LAN or WAN). TLS is the Internet Engineering Task Force (IETF) standardized version of SSL, which was originally implemented on Netscape Navigator. For brevity, TSL/SSL is referred to as TLS for the remainder of this document.

Most people who use a web browser, such as Mozilla Firefox, frequently use TLS without realizing it. When HTTPS is listed on the address bar on a web browser, it means that TLS is being used for communicating with the web server that is specified on the address bar. TLS ensures that data that goes back and forth between the client and server is encrypted when it travels over the network. TLS can ensure that the client and the server are authenticated websites such as those hosted by Internet banking and Internet retail stores, which deploy TLS to ensure that credit card numbers and credentials of their customers are transported securely over the Internet.

This paper describes how to create a configuration that provides secure communication between a HyperSwap server and Tivoli Storage Productivity Center for Replication, by using the same technology that protects your credit cards when you interact with an Internet retailer.

The following steps describe a simplified process:

1. The server and client decide to use TLS.
2. The client initiates the *handshake* process by sending a *client hello* message. The message includes the list of encryption algorithms that are supported by the client.

3. The server selects its preferred encryption algorithm from the list that is provided by the client and sends the client its server certificate along with its choice of encryption algorithm for the session.
4. The client verifies the validity of the server certificate by checking the cryptographic signature of the certificate authority (CA). It also checks whether the certificate authority that signed the server certificate is among the database of trusted authorities in its database of certificates.
5. When the certificate is known to be valid, the client proceeds to generate a new encryption key, based on the scheme that is selected by the server. This new key is shared between the server and the client for the encryption of the communication only in this specific session. To securely send this new key to the server, the client encrypts the new key with the public key that is embedded in the server certificate.
6. The server receives the new key for the session and decrypts it using its private key.
7. The client proceeds to send a message telling the server that it is ready for secured communication and that further messages will be encrypted.
8. The server reciprocates.
9. The encrypted session begins.

Chapter 2, “Generating certificates using RACF” on page 5 explains how to generate the required certificates.



Generating certificates using RACF

In general, the best place to obtain certificates is from a well-known Internet certificate authority (CA). However, in certain situations, such as within your test environment, you might want to create your own CA. This chapter explains how to create a self-signed server certificate on z/OS by using IBM Resource Access Control Facility (RACF). The CA is prepared so that it can be exported and stored in the client. The client requires the CA to verify the authenticity of the server certificate.

2.1 CLIST

Tivoli Storage Productivity Center for Replication provides two command lists (CLISTs) that may be used to perform the steps outlined in the following sections:

- ▶ 2.3, “Generating a self-signed certificate” on page 11
- ▶ 2.4, “Creating a certificate for the server using your user ID” on page 12

Both sample CLISTs (CERTCRE8.sample and CERTIMPT.sample) are in the following folder:

```
<TPC-R INSTALL ROOT>/wlp/usr/servers/replicationServer/etc/
```

The default certificate file (zosKey.p12) that is included with Tivoli Storage Productivity Center for Replication is also in this folder. To use this certificate with the CERTIMPT CLIST you must first move it into a z/OS data set.

To move the default certificate into a z/OS data set, use the **0GET** command, specifying the binary option. The **0GET** command allows you to copy a z/OS UNIX file from the z/OS UNIX System Services hierarchical file system (HFS) into a z/OS data set.

2.1.1 CERTCRE8.sample

The CERTCRE8.sample CLIST generates a unique self-signed certificate to represent the certificate authority (CA) and then generates another certificate, which is signed with the first certificate to authenticate the HyperSwap instance on the system (or sysplex) where this CLIST is executed. This certificate is then connected to a key ring on the system where the CLIST is executed so that is available to all systems within the scope of RACF, typically a single sysplex or RACF remote sharing facility (RRSF).

If both Tivoli Storage Productivity Center for Replication and HyperSwap are not running within the scope of RACF, which is likely the case, you must import this certificate on all other systems that are managed by Tivoli Storage Productivity Center for Replication or where Tivoli Storage Productivity Center for Replication might run.

For example, if Tivoli Storage Productivity Center for Replication that is running in Sysplex A is managing HyperSwap that is running in Sysplex B and another HyperSwap in Sysplex C, you might want to create a single certificate that Sysplexes A, B, and C share. Alternatively, you might want to create one certificate for the sockets connection between Sysplex A and B and create a second certificate for the connection between Sysplex A and C.

In the first case, you execute the CERTCRE8 CLIST on Sysplex B and then use CERTIMPT CLIST to import the certificate that you created on Sysplex B to Sysplex C. You might also need to import the certificate to Sysplex A if that sysplex is also running HyperSwap and can be managed by a standby Tivoli Storage Productivity Center for Replication that is running outside of the sysplex.

In the second case, you execute the CERTCRE8 CLIST once on Sysplex B and once on Sysplex C. You might also need to execute the CERTCRE8 CLIST on Sysplex A if that sysplex is also running HyperSwap and can be managed by a standby Tivoli Storage Productivity Center for Replication running outside of the sysplex.

This CLIST takes two parameters:

- ▶ **USERID:** This is the user ID that is associated with the HyperSwap Address Space.
- ▶ **DATASETNAME:** The name of the data set to where the certificate will be exported. You may specify any data set name you want, for example IBMUSER.ZOSKEY.P12.

2.1.2 CERTIMPT.sample

The CERTIMPT.sample CLIST imports the Tivoli Storage Productivity Center for Replication supplied default certificate, a self-signed certificate that was created by using the CERTCRE8 CLIST, or a certificate that was created in some other way. If you choose to import the default certificate that is provided with Tivoli Storage Productivity Center for Replication, this is a less secure option, because it uses the same certificate for all Tivoli Storage Productivity Center for Replication servers. Using the Tivoli Storage Productivity Center for Replication supplied default certificate is less secure than a self-signed certificate, which is less secure than a CA-signed certificate. However it is also an easy option to use.

This CLIST might also be useful to connect two z/OS Systems to the same Tivoli Storage Productivity Center for Replication server, and to use the same certificate for both.

This CLIST takes two parameters:

- ▶ **USERID:** This is the user ID that is associated with the HyperSwap address space.
- ▶ **DATASETNAME:** This is the name of data set from where you will import the certificate.

This is typically the same data set name that is specified for the CERTCRE8.sample CLIST (2.1.1, “CERTCRE8.sample” on page 6). Tivoli Storage Productivity Center for Replication also includes a zosKey.p12 file in the same folder. If you use the default certificate, this file must be imported if you are using this CLIST. This file must be put into the data set prior to running the CLIST.

2.1.3 Considerations: Tivoli Storage Productivity Center for Replication not on z/OS

Sometimes, you might want to deploy Tivoli Storage Productivity Center for Replication on a distributed platform. For example, in the case of Metro Global Mirror with HyperSwap, the remote data center might be dark, where z/OS is not normally active unless the workload must be moved there. In this case, having the Tivoli Storage Productivity Center for Replication standby running on a distributed server is preferred so that the **RECOVER** command can be issued, allowing the volumes to be placed in a state where z/OS can be started from them.

In this environment, if you are using the default certificate file that is included with Tivoli Storage Productivity Center for Replication, zosKey.p12, it already is located in the proper directory.

If you created a unique certificate, it must be installed in the following directory:

```
<TPC-R INSTALL ROOT>/wlp/usr/servers/replicationServer/etc/
```

You may do this by using FTP to transfer the file in binary directly from the MVS data set to the file system of the distributed system.

Another option is to use the **OPUT** command, specifying the binary option to place the certificate in the z/OS UNIX System Services HFS. Use the **OPUT** command to copy an MVS data set member into the z/OS UNIX System Services HFS. When it is in the z/OS UNIX System Services HFS, you may use FTP to transfer the file from the z/OS HFS to the distributed system's HFS.

2.2 Finding the user ID for HyperSwap

In general, the best approach is to ask your security administrator or the person that installed and configured HyperSwap on your system for the user IDs that are associated with HyperSwap. However if that is not possible, here are methods to determine the user ID.

2.2.1 Using RACF to determine the user ID

RACF provides two ways to assign RACF identities to started procedures:

- ▶ The STARTED class
- ▶ The started procedures table (ICHRIN03)

The STARTED class

The STARTED class allows you to modify the security definitions for started procedures dynamically, using the **RDEFINE** and **RALTER** commands, with no need to modify code or use IPL again. Also, with the STARTED class, you can process job names in addition to started procedure names.

RACF can assign different user IDs and group names to the same started member, depending on the job name that is used.

Profiles in the STARTED class have a segment, STDATA, that contains fields for user ID, group name, trusted flag, privileged flag, and a trace flag. The user ID can be a RACF user ID or the character string =MEMBER, which indicates that the member name is to be used as the user ID. The group name can be a RACF group name or the =MEMBER character string, which indicates that the member name is to be used as the group name.

STDATA segment

This segment is used to control security for started tasks. Specify STDATA only for profiles in the STARTED class.

Example

Issue **SETROPTS GENERIC(STARTED)** command, if not already issued, to allow generic profiles to be created in the STARTED (started task) class.

Issue the RACF **RDEFINE** command (Example 2-1) to define the HSIB started task profile.

Example 2-1 RDEFINE command to define HSIB started task to RACF

```
RDEFINE STARTED HSIB.* STDATA(USER(userid) GROUP(group-name) TRUSTED(YES))
```

To see what was specified for the started task HSIB, use the RACF **RLIST** command (Example 2-2 on page 9).

Example 2-2 RACF RLIST command output

```
rlist started hsib stdata

CLASS      NAME
-----    ----
STARTED    HSIB

LEVEL  OWNER      UNIVERSAL ACCESS  YOUR ACCESS  WARNING
-----  -
 00    USER01      NONE              NONE         NO

INSTALLATION DATA
-----
NONE

APPLICATION DATA
-----
NONE

AUDITING
-----
FAILURES(READ)

GLOBALAUDIT
-----
NONE

NOTIFY
-----
NO USER TO BE NOTIFIED

STDATA INFORMATION
-----
USER= USER01
GROUP= MYGROUP
TRUSTED= NO
PRIVILEGED= NO
TRACE= NO
***
```

In this case, the user ID specified under STDATA INFORMATION is the user ID that is associated with the started task HSIB.

The **RDEFINE**, **RALTER**, and **RLIST** commands define, modify, and list profiles in the STARTED class. For more information about these commands, see *z/OS Security Server RACF Command Language Reference*, SA22-7687.

Also, for further details, see the following resources:

- ▶ The “Associating started procedures and jobs with user IDs” topic in *z/OS Security Server RACF System Programmer’s Guide*, SA22-7681
- ▶ The “Using Started Procedures” topic in *z/OS Security Server RACF Security Administrator’s Guide*, SA22-7683

The started procedures table

If your security administrator defines the security definitions for started procedures by using the started procedures table, look at the LPALIB module, ICHRIN03. See the RACTABLE member in SYS1.SAMPLIB for a sample started procedures table, which looks similar to Example 2-3.

Example 2-3 Sample started procedures table

```

ICHRIN03  CSECT
COUNT    DC      AL2(((ENDRIN03-COUNT-2)/32)+32768)
*----- First Entry -----
ENTRY1    EQU      *
PROC1     DC      CL8'PROC1  '
USERID1   DC      CL8'TSO1   '
GROUP1    DC      CL8'SYS1   '
FLAGS1    DC      XL1'00'
          DC      XL7'00'
*----- Last Entry -----
ENTRY2    EQU      *
PROC2     DC      CL8'*      '
USERID2   DC      CL8'TSO2   '
GROUP2    DC      CL8'='     '
FLAGS2    DC      XL1'00'
          DC      XL7'00'
*-----
ENDRIN03  EQU      *
          END

```

Using the started procedures table is not preferred because you must edit the table, assemble, and link-edit the updated table, and then restart the system again. Instead, use the RACF STARTED class.

2.2.2 Using System Display and Search Facility (SDSF)

You will need the user ID that is associated with the HyperSwap address spaces. Complete the following steps to find the user ID that HyperSwap is using.

1. Open SDSF.
2. Enter the SDSF Status Display by entering **st** on the command line.
3. Filter the jobs shown by entering **prefix hsib** on the command line.
4. Check the output table and look for the owner of HSIB. This owner is the user ID that you need.

Typically, the two started tasks for z/OS HyperSwap are called HSIB and HSIBAPI. If your system programmer chose different names, use the name that is associated with the program IOSHMCTL.

A sample SDSF display is shown in Example 2-4.

Example 2-4 Sample SDSF display

```

Display Filter View Print Options Search Help
-----
SDSF STATUS DISPLAY ALL CLASSES                LINE 1-1 (1)
COMMAND INPUT ===>                            SCROLL ==> CSR
PREFIX=HSIB  DEST=(ALL) OWNER=*  SYSNAME=
NP  JOBNAME  JobID  Owner  Prty Queue  C Pos  SAff  ASys Status
    HSIB     STC25535  SETUP    15 EXECUTION    SYSA  SYSA

```

2.3 Generating a self-signed certificate

Generate a self-signed certificate to represent the local certificate authority. This certificate is used as the certificate-authority certificate. In our example, we used the default encryption algorithm, which is RSA, and the default encryption strength, which is 1024 bits.

Important: Be careful which user ID you use for these commands. Use the same user ID as HyperSwap. If another user ID is used, AT-TLS cannot access the necessary key ring to fetch the certificates and the setup will not work. In this example, a user ID of SETUP is used.

The RACF **RACDCERT** command to generate a certificate has the following parameters:

OU organization-unit-name
O organization-name
C country

Issue the RACF command shown in Example 2-5 to generate the certificate.

Example 2-5 Sample RACF RACDCERT command

```
RACDCERT GENCERT CERTAUTH SUBJECTSDN (OU('TPCR Certificate Authority')
O('tpcr') C('us')) KEYUSAGE(CERTSIGN)
WITHLABEL('TPCR Local Certificate Authority')
```

2.3.1 Refreshing to ensure the certificate is in storage

Run the RACF commands shown in Example 2-6 to refresh and ensure that the certificate is in storage.

Example 2-6 Sample RACF SETR commands

```
SETR CLASSACT(DIGTCERT)
SETR RACLIST(DIGTCERT)
SETR RACLIST(DIGTCERT) REFRESH
```

2.3.2 Exporting the certificate to a data set

Next, we export this certificate to a data set so that we can use it on the system where Tivoli Storage Productivity Center for Replication is running. Use the DSN parameter to specify the output-data-set-name.

You may use any data set name. In our case we chose 'TPCR.LOCCERTA.CERT' as the name. Issue the command in Example 2-7 to create the 'TPCR.LOCCERTA.CERT' data set.

Example 2-7 Sample RACF commands to create

```
RACDCERT EXPORT (LABEL('TPCR Local Certificate Authority')
CERTAUTH DSN('TPCR.LOCCERTA.CERT') FORMAT(CERTDER))
```

2.4 Creating a certificate for the server using your user ID

This section provides an example of how to create a server certificate with the default size for the private key, 1024 bits, and the default key type, ICSF RSA. The certificate is signed by the CA that was created in 2.3, “Generating a self-signed certificate” on page 11.

2.4.1 RACDCERT command

The following parameters are used in the **RACDCERT** command:

CN	common-name
OU	organization-unit-name
O	organization-name
C	country

You may use the following optional parameters:

NOTAFTER (DATE(yy-yy-mm-dd) TIME(hh:mm:ss))	The local date and time after which the certificate is no longer valid
SIZE	Key-size
KEYUSAGE	A combination of the following possible values: HANDSHAKE, DATAEN-CRYPT, DOCSIGN, CERTSIGN

Issue the RACF command shown in Example 2-8.

Example 2-8 Sample RACF RACDCERT command

```
RACDCERT GENCERT ID(SETUP) SUBJECTSDN(CN('TPCR Client') OU('Hyperswap Server')
O('TPCR') C('US')) WITHLABEL('Hyperswap Manager')
SIGNWITH(CERTAUTH LABEL('TPCR Local Certificate Authority')) KEYUSAGE(HANDSHAKE)
```

2.4.2 Creating a key ring for the certificates

Issue the RACF command shown in Example 2-9 to create a key ring for the certificates.

Example 2-9 RACF RACDCERT command to create a key ring

```
RACDCERT ADDRING(tpcrkeyring) ID(SETUP)
```

2.4.3 Connecting the server certificate to the new key ring

Issue the RACF command shown in Example 2-10 to place the server certificate into the new key ring.

Example 2-10 RACF RACDCERT command to place server certificate into the key ring

```
RACDCERT CONNECT(LABEL('Hyperswap Manager')
RING(tpcrkeyring) DEFAULT) ID(SETUP)
```

2.4.4 Connecting the CA certificate to the new key ring

Issue the RACF command shown in Example 2-11 to place the CA certificate into the new key ring.

Example 2-11 RACF RACDCERT command to place CA certificate into key ring

```
RACDCERT CONNECT(CERTAUTH LABEL('TPCR Local Certificate Authority')
RING(tpcrkeyring)) ID(SETUP)
```

2.4.5 Giving the SETUP user permission to read its own key ring

Issue the command shown in Example 2-12 to give the user ID the SETUP permission to read its own key ring.

Example 2-12 Sample RACF RDEFINE command

```
RDEFINE FACILITY IRR.DIGTCERT.LISTRING UACC(NONE)
PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ID(SETUP) ACCESS(READ)
```

Issue the command shown in Example 2-13 to perform a refresh so that the RACF profile changes can take effect.

Example 2-13 Sample RACF RACLIST command

```
SETR RACLIST(DIGTCERT) REFRESH
```

2.5 Commands that are useful for validation

The following commands are useful for validation:

- ▶ **RACDCERT LIST**
- ▶ **RACDCERT LISTRING**

2.5.1 RACDCERT LIST command

Issue the command shown in Example 2-14 to list certificates with a given label.

Example 2-14 RACDCERT LIST command


```
RACDCERT LIST(LABEL('HyperSwap Manager'))
```

2.5.2 RACDCERT LISTRING command

Issue the command shown in Example 2-15 to list the certificates in the specified key ring.

Example 2-15 RACDCERT LISTRING command

```
RACDCERT LISTRING[(ring-name)] [ID(ring-owner)]
```



Importing the certificate into a Java keystore file

Tivoli Storage Productivity Center for Replication cannot access raw certificates. Certificates must be in the Java keystore (JKS) format. This chapter explains the tasks to import the certificate authority's certificate into a JKS database file with the iKeyman utility so that Tivoli Storage Productivity Center for Replication can access the certificate.

3.1 Download the certificate

Use binary FTP to download the certificate that was exported in 2.3.2, “Exporting the certificate to a data set” on page 11.

On z/OS, you can place the certificate in the z/OS UNIX System Services hierarchical file system (HFS) by using the **OPUT** command shown in Example 3-1.

Example 3-1 OUPUT command example

```
OPUT 'TPCR.LOCCERTA.CERT'  
'var/opt/Tivoli/RM/wlp/usr/servers/replicationServer/etc/zostrust'
```

3.2 Download and install IBM JRE that includes iKeyman

In our example, we use iKeyman, running on a Windows system, to create the JKS file.

If you are running Tivoli Storage Productivity Center for Replication on IBM AIX®, Windows, or Linux operating systems, iKeyman is already installed.

iKeyman is located in the <JdkInstallRoot>/bin directory. On Linux operating systems, <JdkInstallRoot> is typically /opt/IBM/TPC/jre. Therefore, to locate iKeyman, use /opt/IBM/TPC/jre/bin. If performing this step on z/OS, use the IKEYCMD command-line interface.

If iKeyman is not installed yet, download and install the IBM JRE that includes iKeyman. For more information about using IKEYCMD, see “Appendix B. Using the IKEYCMD command-line interface” in *IBM WebSphere® Host On-Demand Version 10.0: Planning, Installing, and Configuring Host On-Demand*, SC31-6301. This publication is located at the following address:

<http://www.ibm.com/support/docview.wss?uid=pub1sc31630104>

3.3 Create a JKS file and import the CA into the JKS

To create a JKS file and import the CA into the JKS, complete the following steps:

1. Start iKeyman, as shown in Figure 3-1.

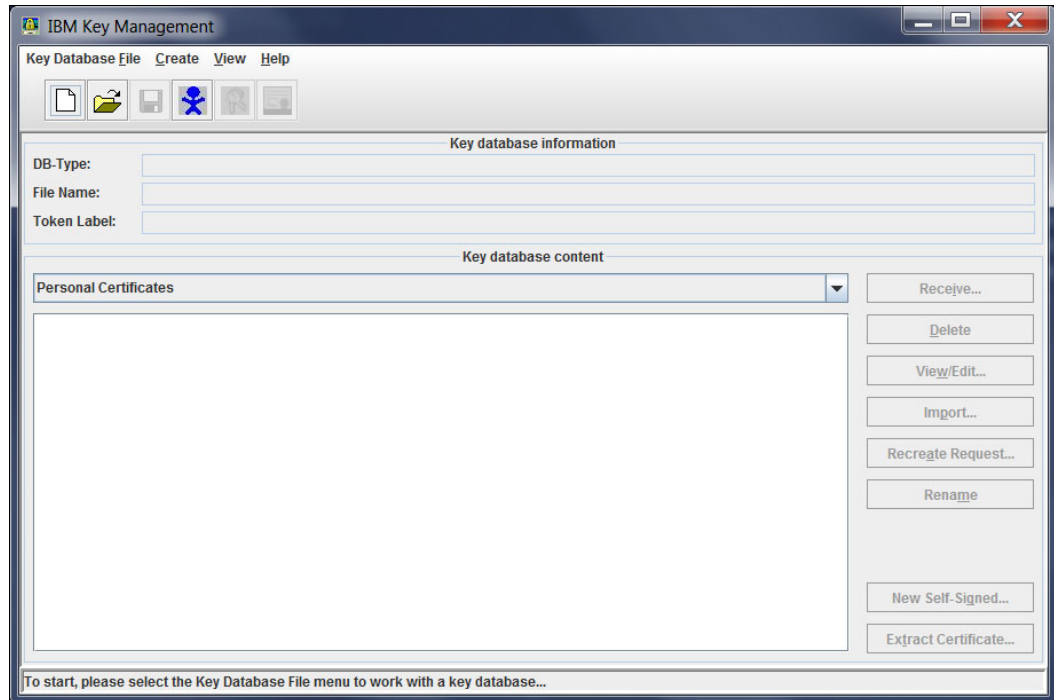


Figure 3-1 Starting iKeyman

2. Create a new database file (Figure 3-2).

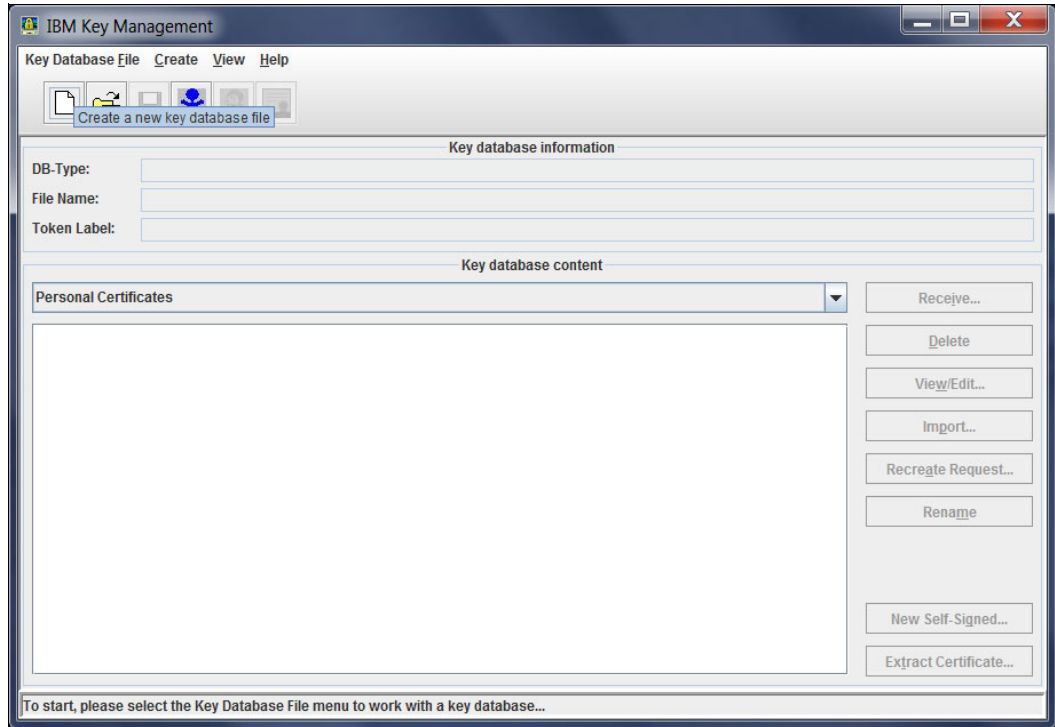


Figure 3-2 Creating a new database file

3. Select **JKS** as the database type, enter a name for the JKS file (Figure 3-3).

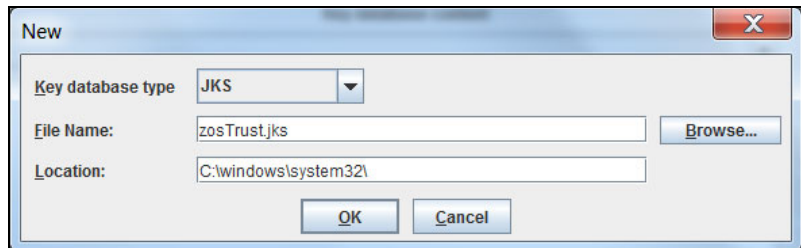


Figure 3-3 Selecting a database type and entering a name for the JKS file

4. Choose a new password for the database (Figure 3-4).

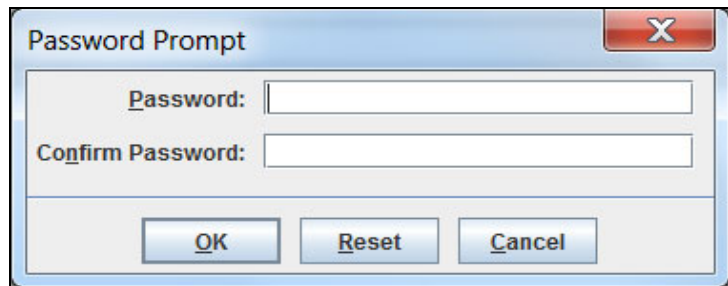


Figure 3-4 Specifying a password for the database

5. Select **Signer Certificates** (Figure 3-5).

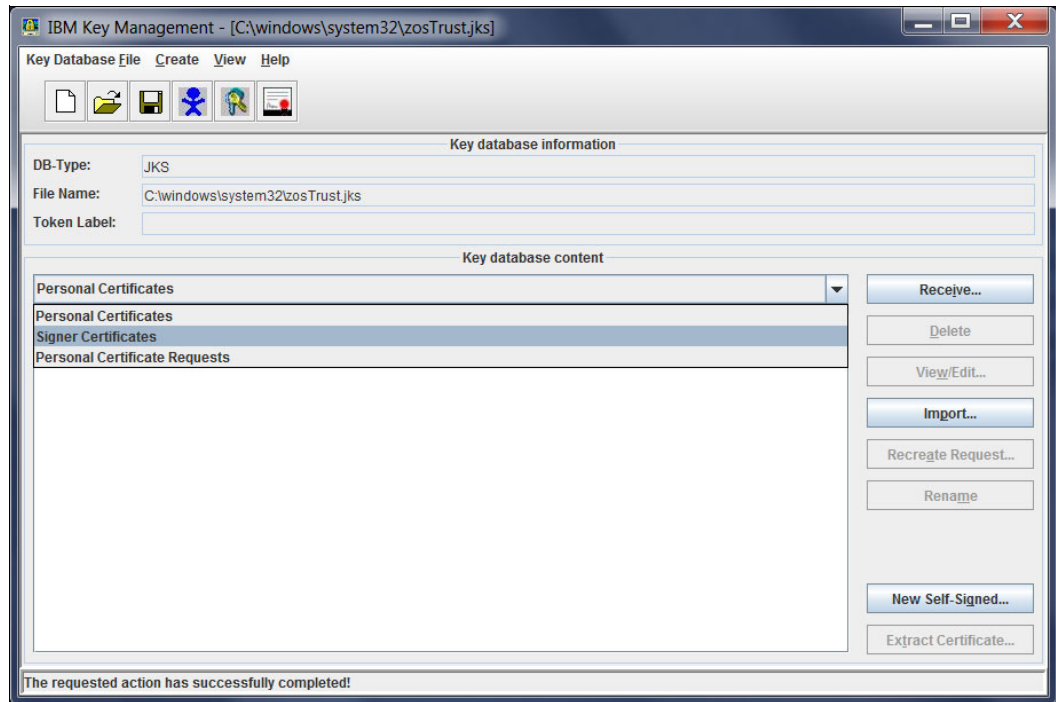


Figure 3-5 Selecting signer certificates

6. Add the certificate for the new CA (Figure 3-6).

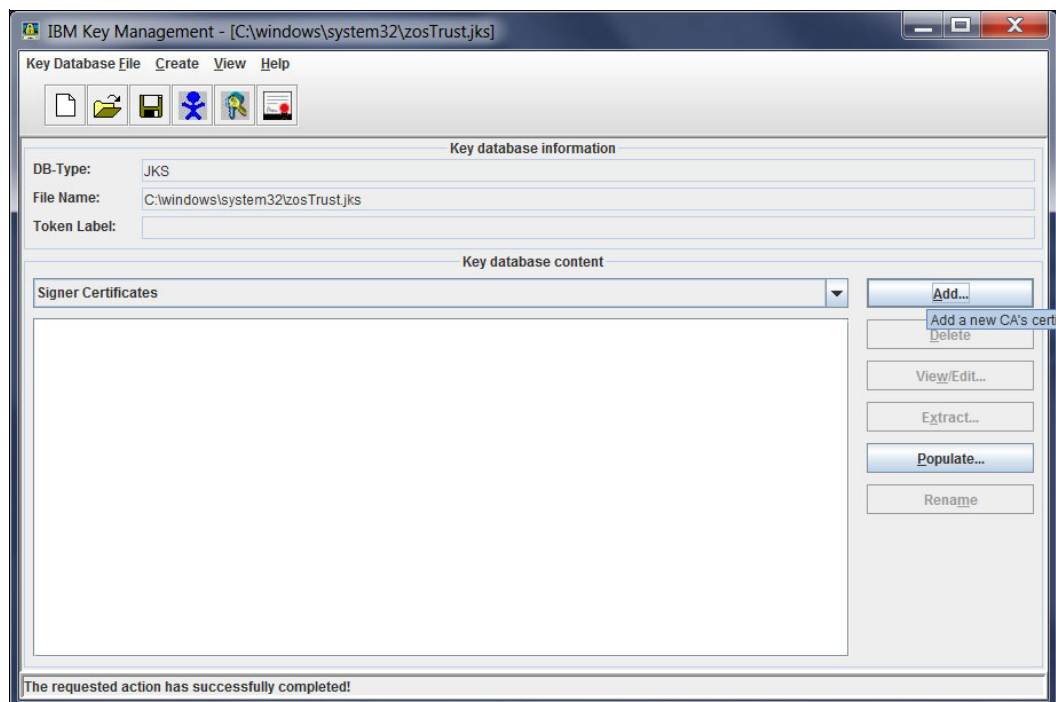


Figure 3-6 Adding the certificate for the new CA

7. Locate the certificate (downloaded in 3.1, “Download the certificate” on page 16), as shown in Figure 3-7.

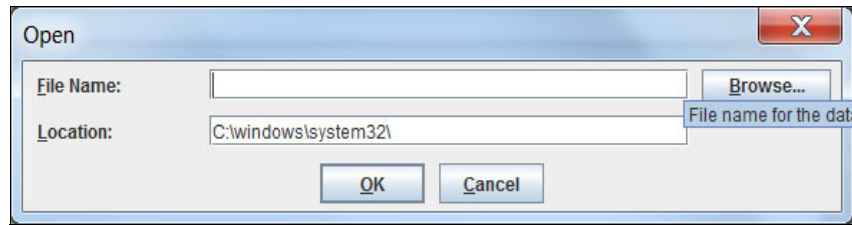


Figure 3-7 Locating the downloaded certificate

8. Open the certificate (Figure 3-8).

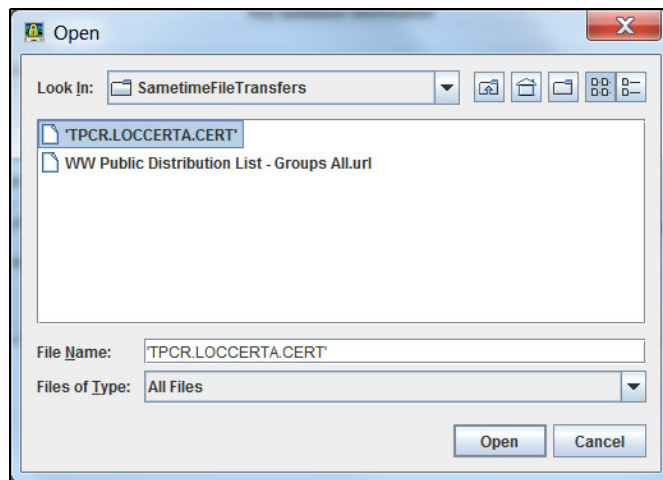


Figure 3-8 Opening the certificate

9. Confirm the certificate (Figure 3-9).

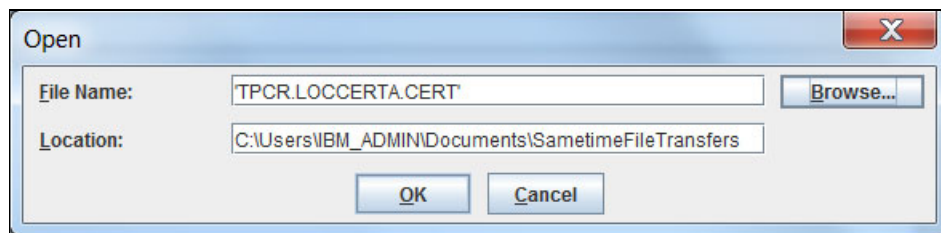


Figure 3-9 Confirming the certificate

10. Enter a label for the certificate (Figure 3-10). This label can be anything that you choose.

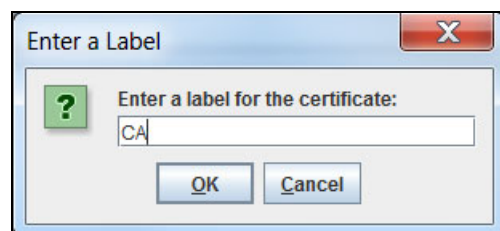


Figure 3-10 Entering a label for the certificate

The JKS file with the CA is ready (Figure 3-11).

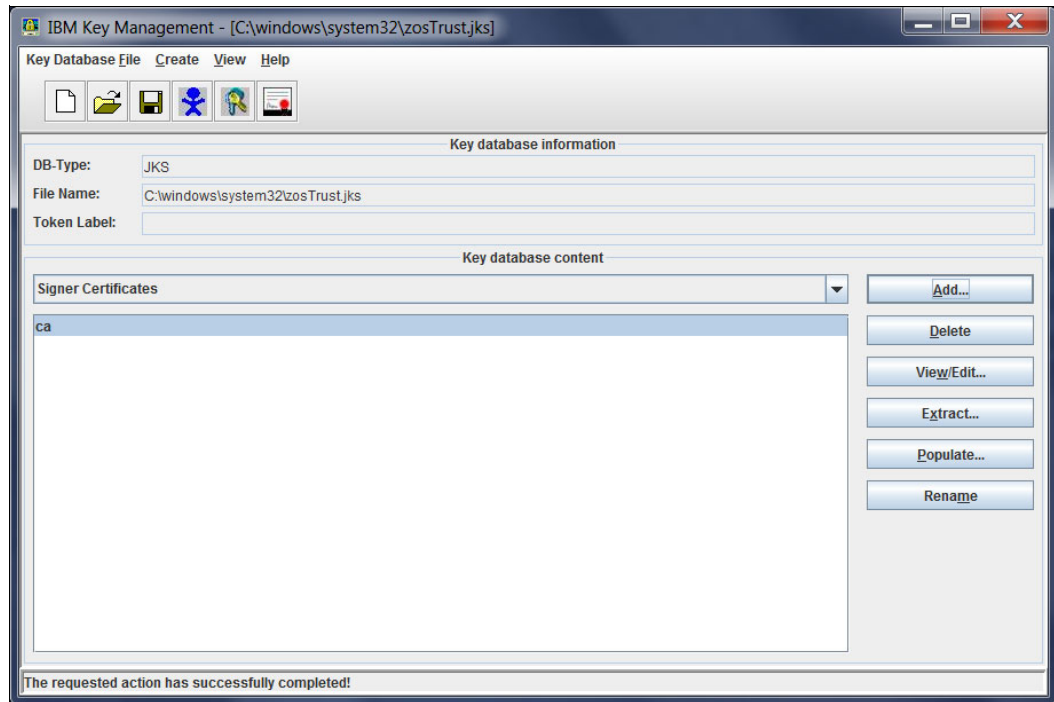


Figure 3-11 JKS file with the CA

3.4 Connect multiple z/OS Systems to the same Tivoli Storage Productivity Center for Replication

If you want to connect two or more z/OS Servers to the same Tivoli Storage Productivity Center for Replication Server and want to use different certificates for each, you must create a single JKS file (.jks) with both certificates imported. Use iKeyman and repeat steps 5 on page 19 through 10 on page 20 in 3.3, "Create a JKS file and import the CA into the JKS" on page 17 for the second certificate.

3.5 Configure replication properties and trust files for Tivoli Storage Productivity Center for Replication

To configure replication properties and trust files for Tivoli Storage Productivity Center for Replication, place the JKS file that you created in 3.3, "Create a JKS file and import the CA into the JKS" on page 17 in the replicationServer/etc directory with the name zosTrust.jks.



Creating a policy file for AT-TLS using the Configuration Assistant GUI

This chapter explains how to create the Application Transparent Transport Layer Security (AT-TLS) policy file and upload it to the necessary systems. This policy was created using the IBM Configuration Assistant for z/OS Communications Server V1R13.

Support note: In z/OS V1R11 and later, the Configuration Assistant is available as a fully supported task in the z/OS Management Facility (z/OSMF) product. After z/OS V1R13, the stand-alone Configuration Assistant is no longer included.

For more information about AT-TLS, view the introduction to AT-TLS presentation:

<http://www.ibm.com/support/docview.wss?uid=swg27028558&aid=1>

This YouTube video also provides a good introduction:

<http://www.youtube.com/watch?v=YKEzX70mo0Q>

Important: The assumption in this chapter is that AT-TLS is already on your system. Thus, the steps outlined here cover only the creation of the policy file. For any z/OS LPARs where you want to configure AT-TLS, you must ensure that the TCP/IP profile is updated to enable AT-TLS and that the policy agent (PAGENT) is started.

To configure AT-TLS in the TCP/IP profile, issue the **TCPCONFIG TTLS** command. For more information about the TCPCONFIG statement or to start PAGENT, see the *z/OS Communications Server IP Configuration Reference*, SC31-8776.

4.1 Download and install IBM Configuration Assistant V1R13

To download IBM Configuration Assistant for z/OS Communications Server, go to the following address:

<http://www.ibm.com/support/docview.wss?uid=swg24013160>

4.2 Create an AT-TLS policy file with IBM Configuration Assistant

Complete the following steps to create an AT-TLS policy file with Configuration Assistant:

1. Start Configuration Assistant as shown in Figure 4-1.

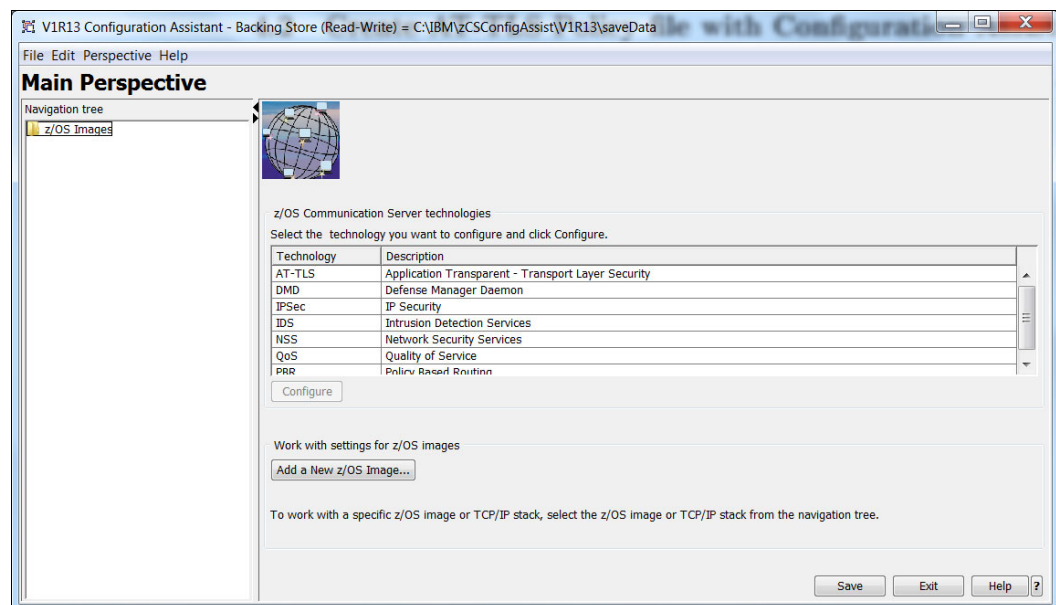


Figure 4-1 Starting configuration assistant

2. Switch to AT-TLS perspective by selecting **AT-TLS** from the Perspective menu, and clicking **Add a New z/OS Image** as shown in Figure 4-2 on page 25.

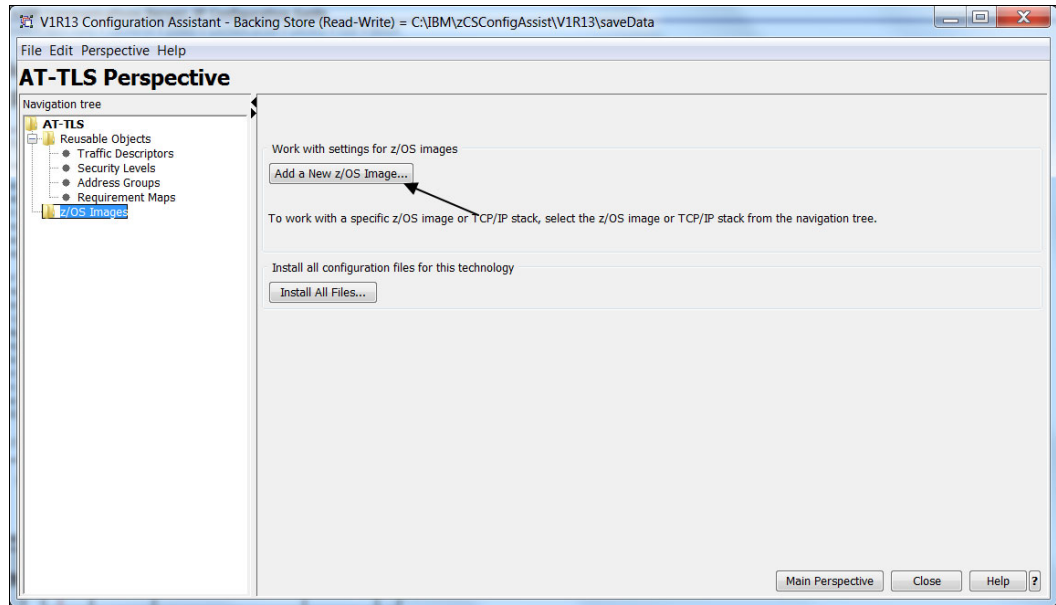


Figure 4-2 AT-TLS perspective

3. Specify the image name, description, z/OS release, and the default key ring database as shown in Figure 4-3. This must be the same key ring name as when the key ring was generated in 2.4.2, “Creating a key ring for the certificates” on page 12. The image name and release are for the z/OS LPAR where HyperSwap is running.

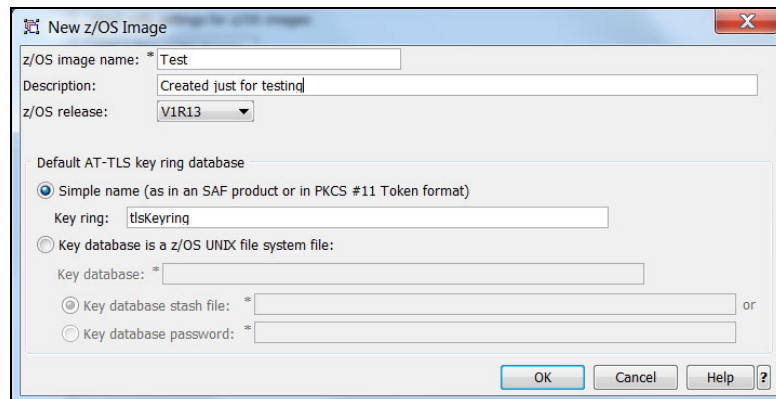


Figure 4-3 Specifying information for a new z/OS image

4. Click **Yes** to add a TCP/IP stack as shown in Figure 4-4.

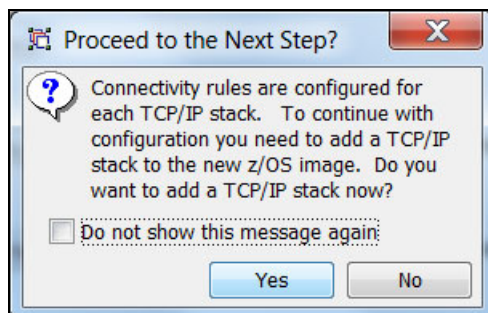


Figure 4-4 Confirmation for adding a TCP/IP stack

- Enter a new stack name as shown in Figure 4-5.
The TCP/IP stack is the stack name on the z/OS system where HyperSwap is running.

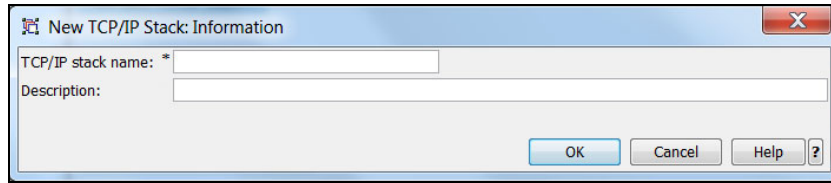


Figure 4-5 Entering a new stack name

- Click **Add** to add a new connectivity rule to define AT-TLS policies for the HyperSwap connections (Figure 4-6).

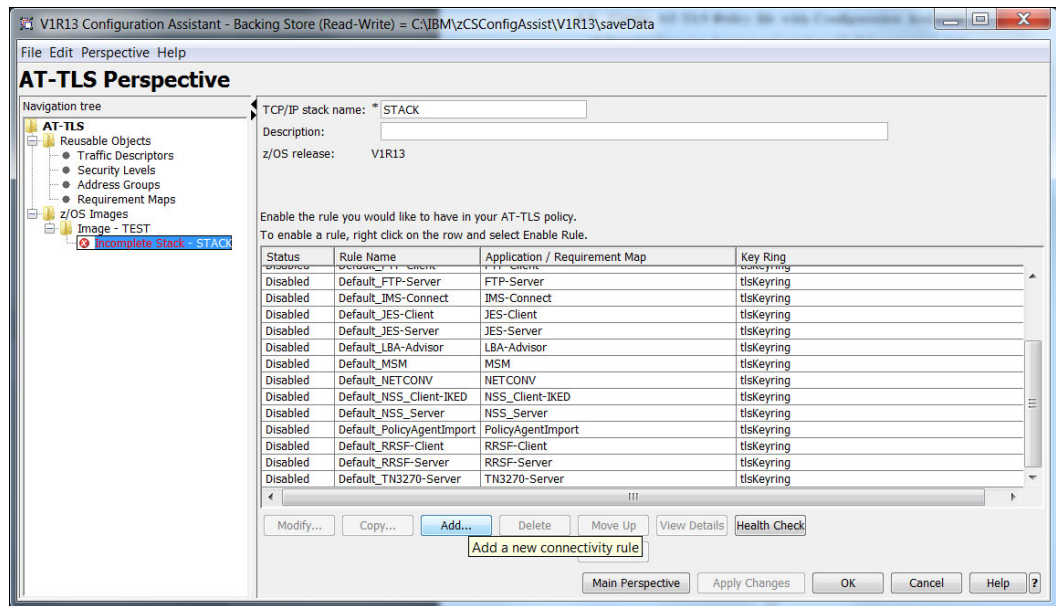


Figure 4-6 Adding a connectivity rule

- The Connectivity Rule wizard opens (Figure 4-7). Click **Next** to continue.

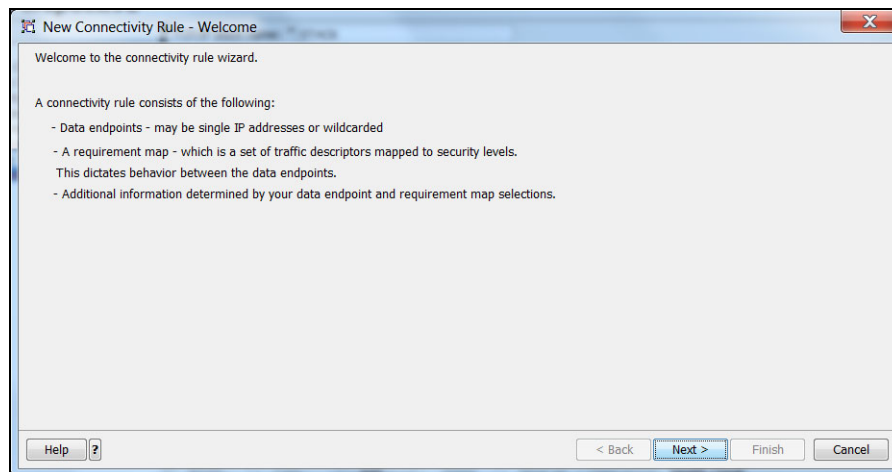


Figure 4-7 Welcome window for the connectivity rule wizard

- Specify the connectivity rule name and IP address groups, and click **Next** (Figure 4-8). In our example, we protect connections by using both IPv4 and IPv6 addresses.

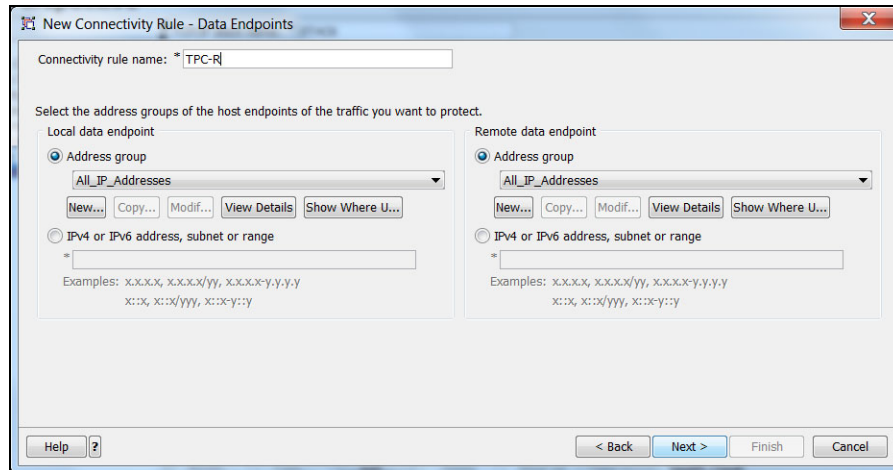


Figure 4-8 Specifying the connectivity rule name and address groups

- Click **Traffic Descriptors** (Figure 4-9) to open the Traffic Descriptor Objects window.

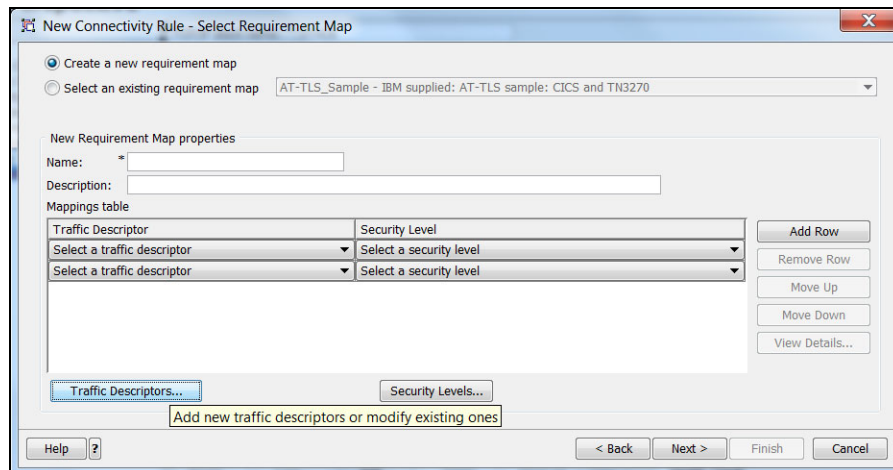


Figure 4-9 Opening the Traffic Descriptor window

- The Traffic Descriptor Objects window opens (Figure 4-10). Click **Add** to add a new Traffic Descriptor to protect both inbound and outbound traffic for the HyperSwap application.

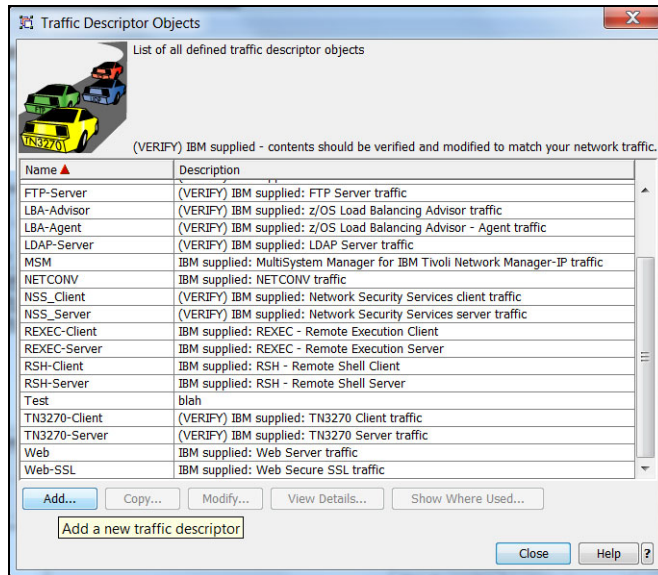


Figure 4-10 Adding a traffic descriptor

- Enter the name for the Traffic Descriptor and click **Add** to specify the traffic type. In our example, we specify TPC-R as the descriptor name, as shown in Figure 4-11.

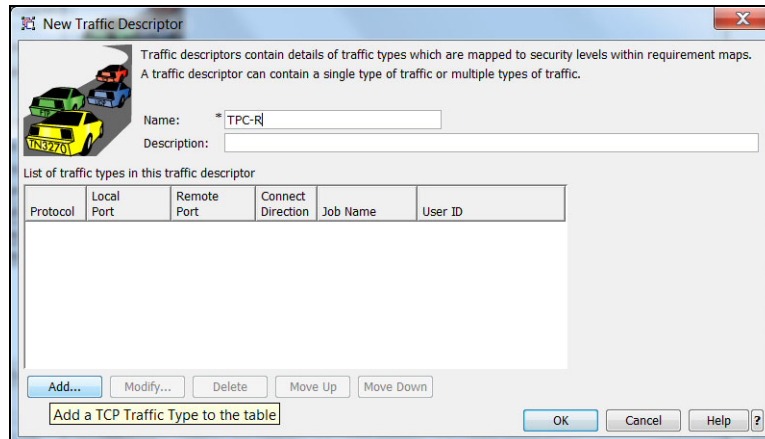


Figure 4-11 Entering the name for the traffic descriptor

12. For the traffic type, specify the port number used by the HyperSwap application. Indicate the TCP connection direction by selecting **Either**, to protect traffic in both directions. Also select **Server** for the handshake role, to allow the HyperSwap application to determine whether a TLS connection needs to be established with Tivoli Storage Productivity Center for Replication. See Figure 4-12.

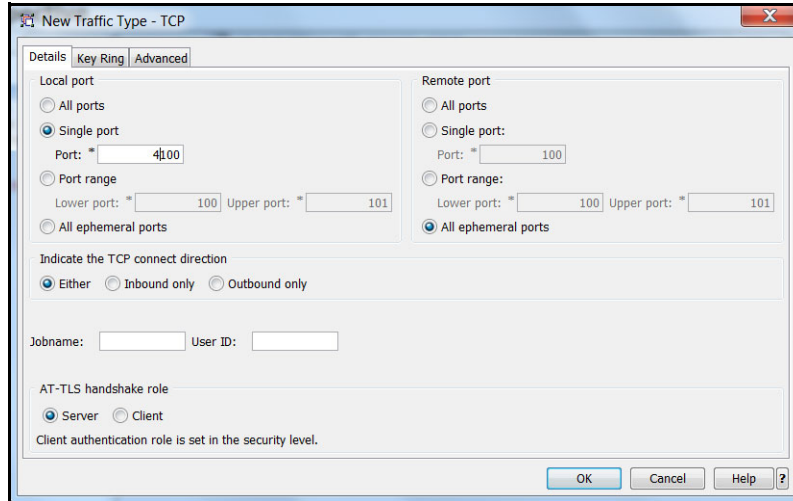


Figure 4-12 Specifying information for the port on the HyperSwap side of the configuration

The HyperSwap Management Address Space procedure is typically member HSIB in SYS1.PROCLIB. A sample HSIB member is shown in Example 4-1.

Example 4-1 Sample SYS1.PROCLIB member HSIB

```
//HSIB      PROC  PORT=14000
//STEP      EXEC  PGM=IOSHMCTL,TIME=NOLIMIT,REGION=OM,
//          PARM= 'SOCKPORT=&PORT'
```

- Specify the name of the key ring (created in 2.4.2, “Creating a key ring for the certificates” on page 12) and the corresponding certificate label (see Figure 4-13). If your environment is likely to have other key rings that are defined in RACF, select **Use a Simple name (as in an SAF product or in PKCS #11 Token format)** and specify tpcrkeyring as the key ring name. Click **OK**.

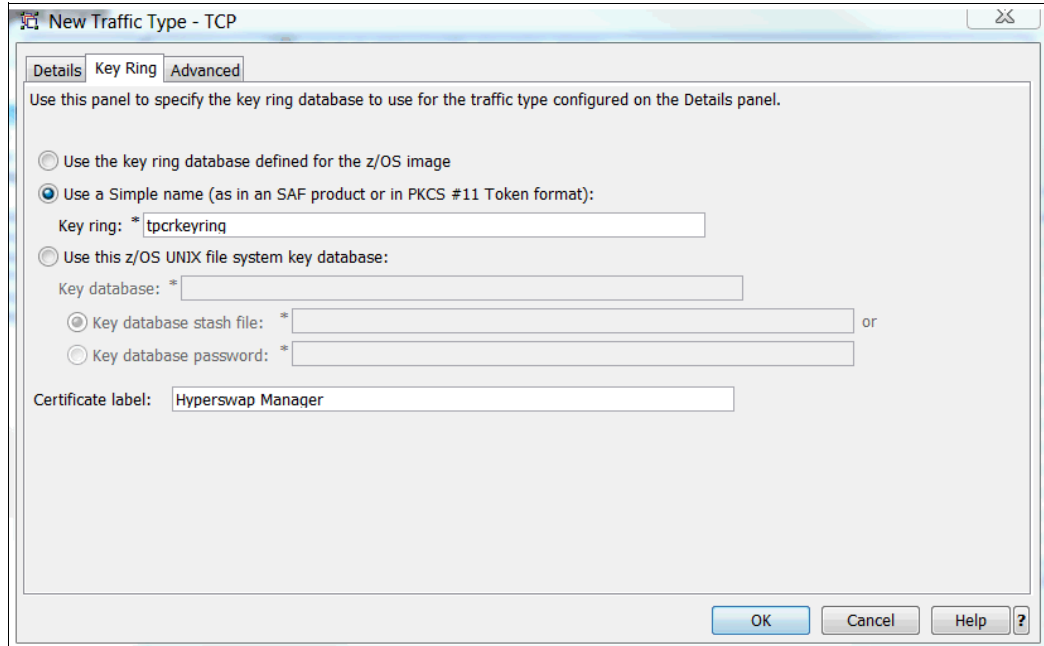


Figure 4-13 Specifying the Certificate label for the key ring

- Click **OK** to exit the New Traffic Descriptor wizard (Figure 4-14).

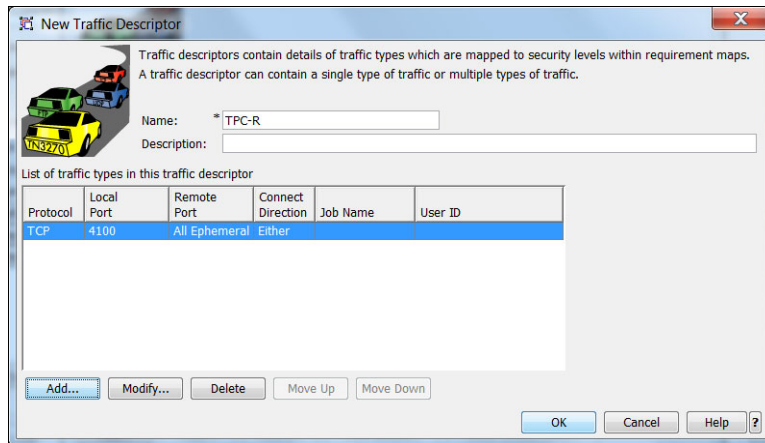


Figure 4-14 Exiting the New Traffic Descriptor wizard

15. Click **Close** to exit the Traffic Descriptor Objects window (Figure 4-15).

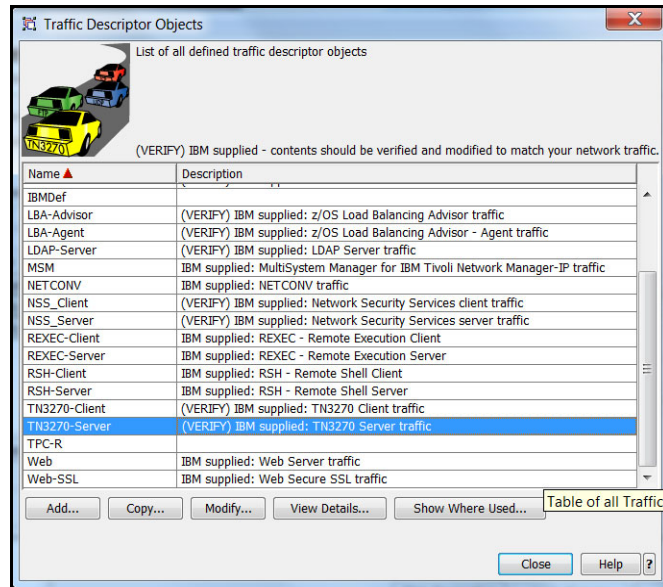


Figure 4-15 Closing the Traffic Descriptors window

16. Back in the Select Requirement Map window (Figure 4-16), select the traffic descriptor that you created. The Name field is populated. In our example, we select TPC-R. Choose the security level for this traffic descriptor. In our example, we select AT-TLS_SILVER (not shown in the figure), which uses cipher suites that provide a medium level of security. Click **Next**.

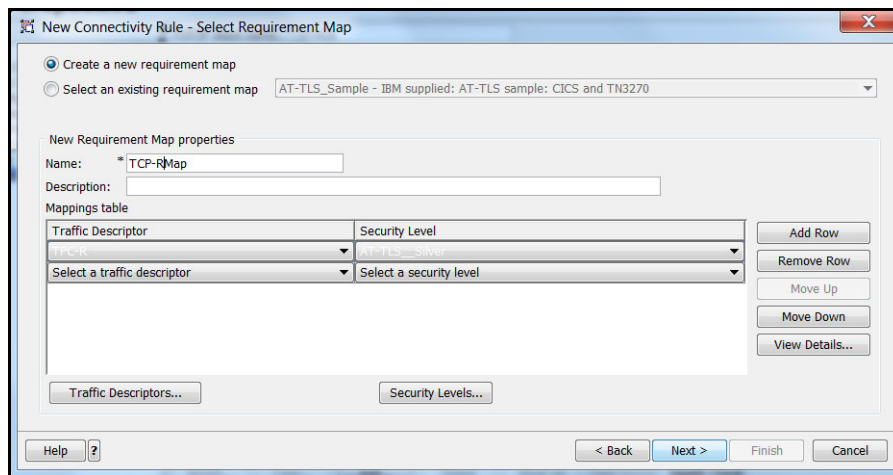


Figure 4-16 Creating a new requirement map

- In the Advanced window (Figure 4-17), click **Finish** to exit the New Connectivity Rule wizard. If any other HyperSwap applications are active on other TCP/IP stacks, repeat these steps for the other stacks.

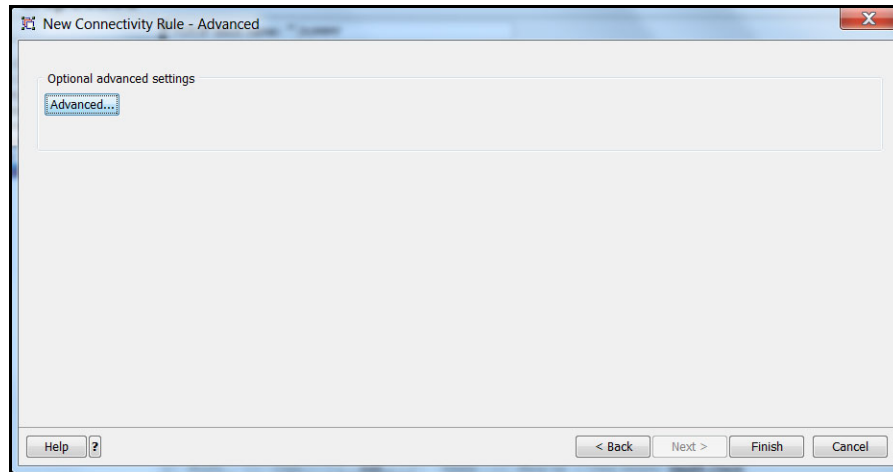


Figure 4-17 Finishing the new connectivity wizard

- Click **Apply Changes** (Figure 4-18).

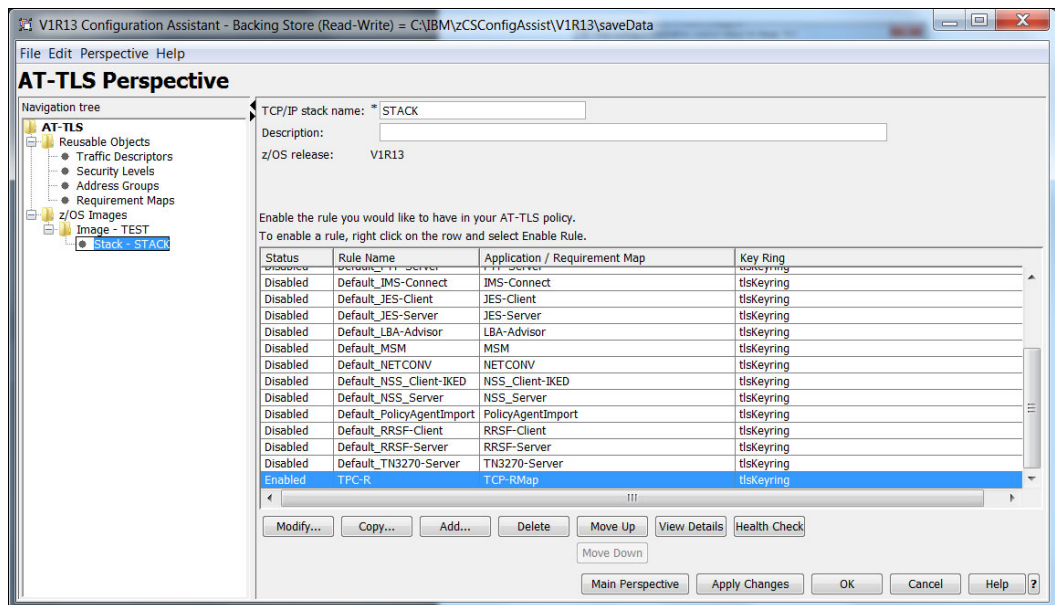


Figure 4-18 Applying changes

19. Click **Application Setup Tasks** (Figure 4-19).

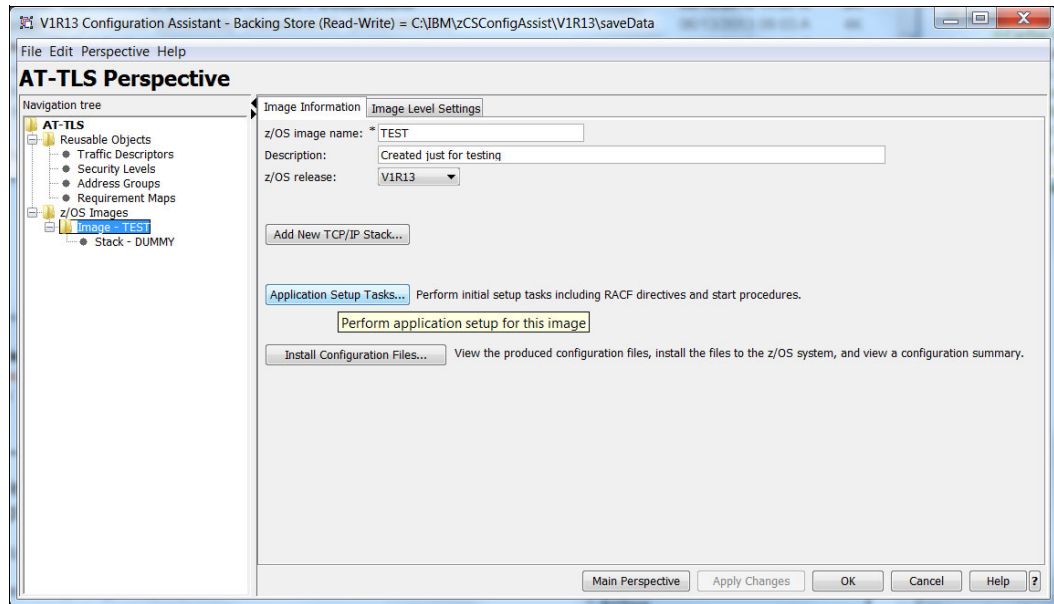


Figure 4-19 Opening application setup tasks

20. Go to **Installation Location Setup** to input the installation setup (Figure 4-20).

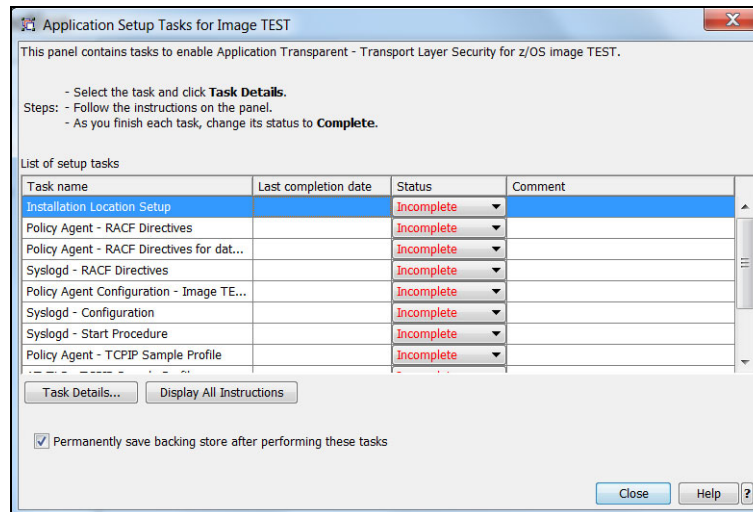


Figure 4-20 Installation location setup

21. Click **Location Information** to provide the FTP information needed to upload this configuration file to the z/OS system where the HyperSwap application is active. See Figure 4-21.

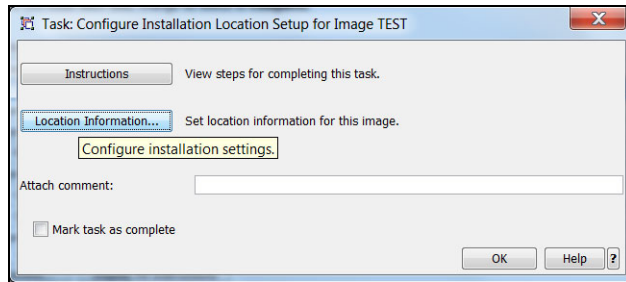


Figure 4-21 Opening the Installation Location Setup panel

22. Enter the FTP information as shown in Figure 4-22.

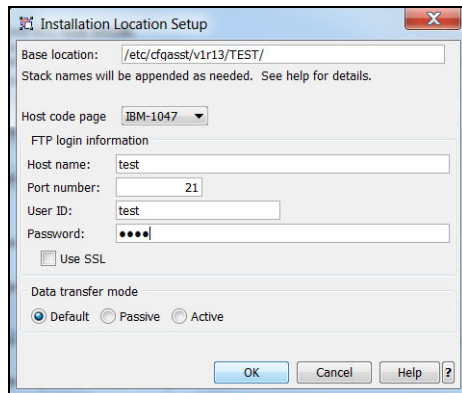


Figure 4-22 Entering location information

23. In the Task Configuration Location Setup window, click **OK** (Figure 4-23).

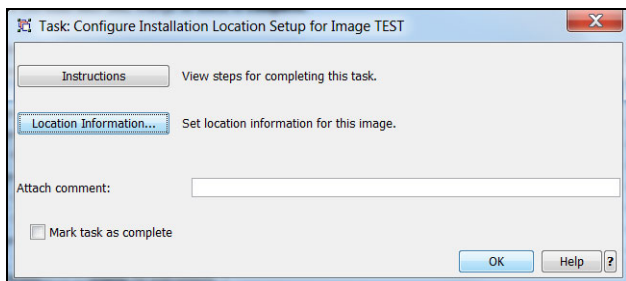


Figure 4-23 Task Configure Installation Location Setup panel

24. Click **Close** to exit from the Application Setup Tasks window (Figure 4-24).

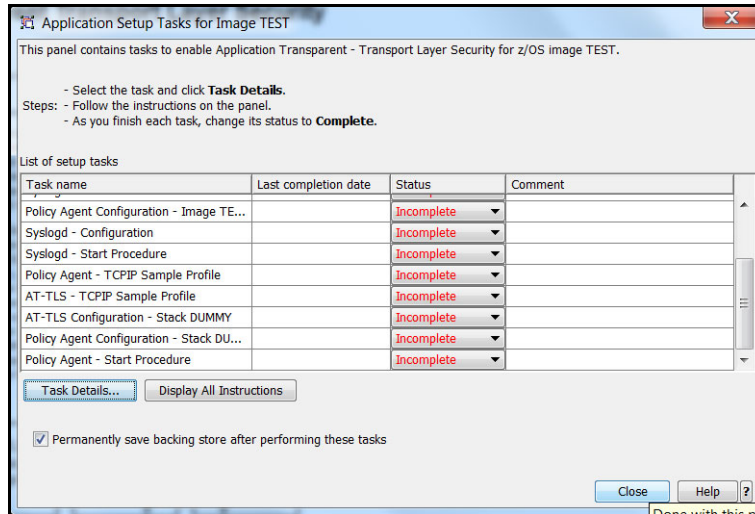


Figure 4-24 Closing the Application Setup Tasks window

The configuration tasks are now completed and you can proceed with the installation of the configuration files.

25. Click **Install Configuration Files** to upload the configuration file (Figure 4-25).

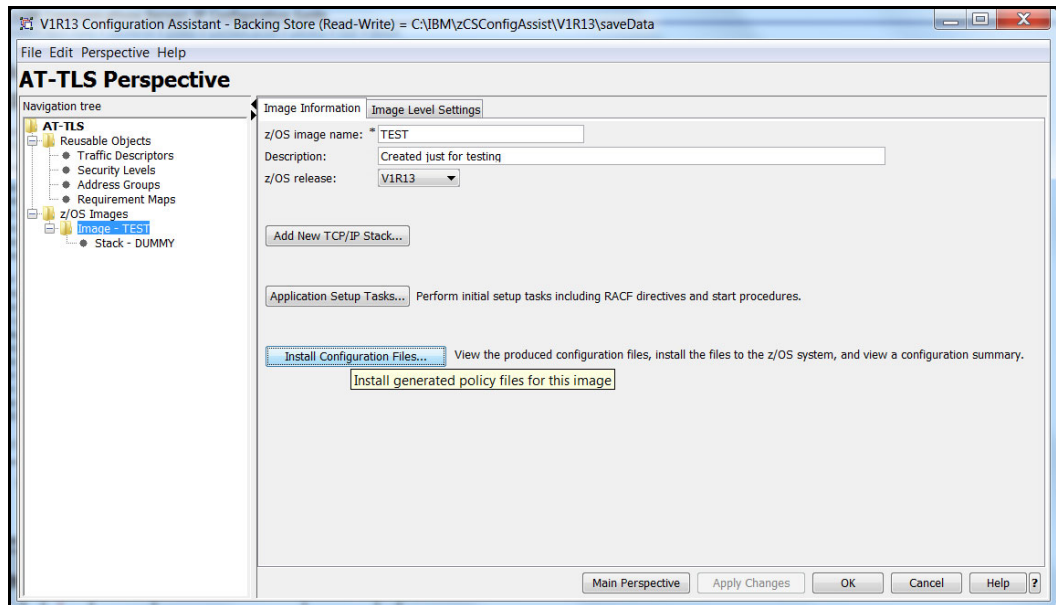


Figure 4-25 Installing configuration files

26. Select a Configuration file and click **Install** (Figure 4-26).

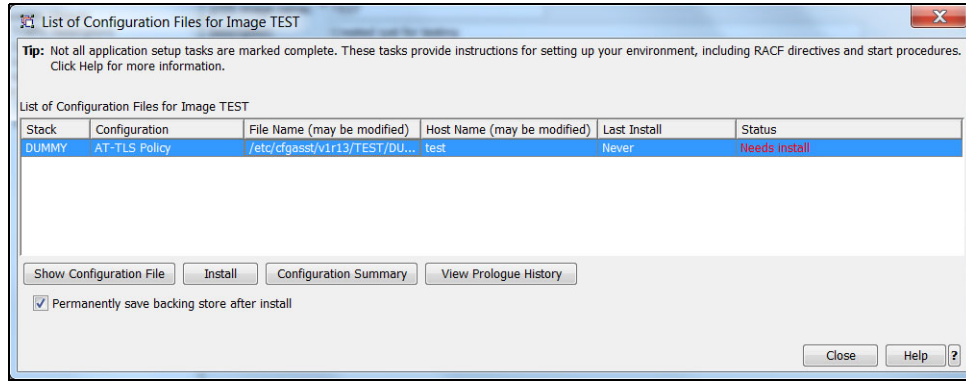


Figure 4-26 Starting to install configuration files

27. Confirm the credentials and the remaining FTP login information that was entered previously (Figure 4-27).

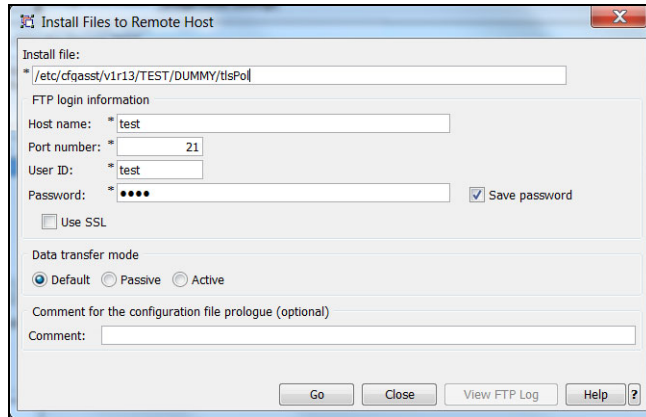


Figure 4-27 Confirming credentials

28. Click **Go** to install the policy file (Figure 4-28).

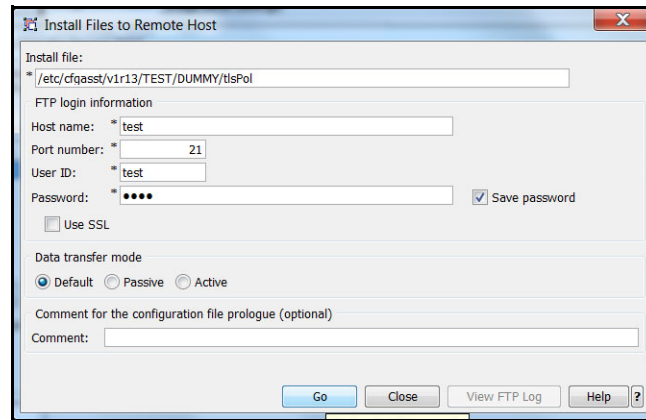


Figure 4-28 Installing the policy file

4.3 Refresh AT-TLS settings

To refresh the AT-TLS settings in the TCP/IP stack where the configuration file was uploaded, recycle the policy agent (PAGENT) address space.

Verifying the system

Complete the following steps to verify the system setup that is described in this paper:

1. Log in to the Tivoli Storage Productivity Center for Replication administrator console as shown in Figure 5-1.

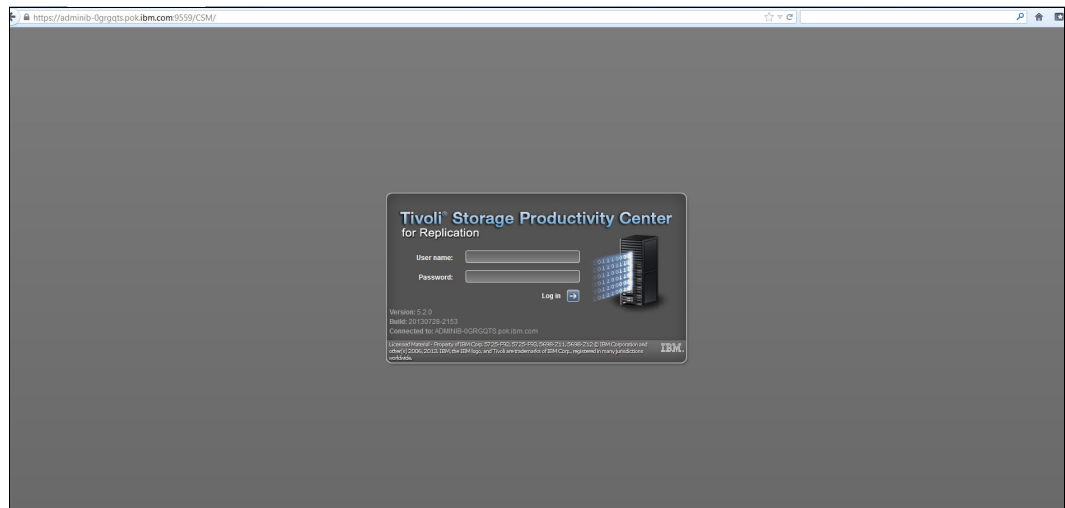


Figure 5-1 Tivoli Storage Productivity Center for Replication console

- From the Health Overview pane (Figure 5-2), click **Host Systems** to go to the Host Systems dialog box.

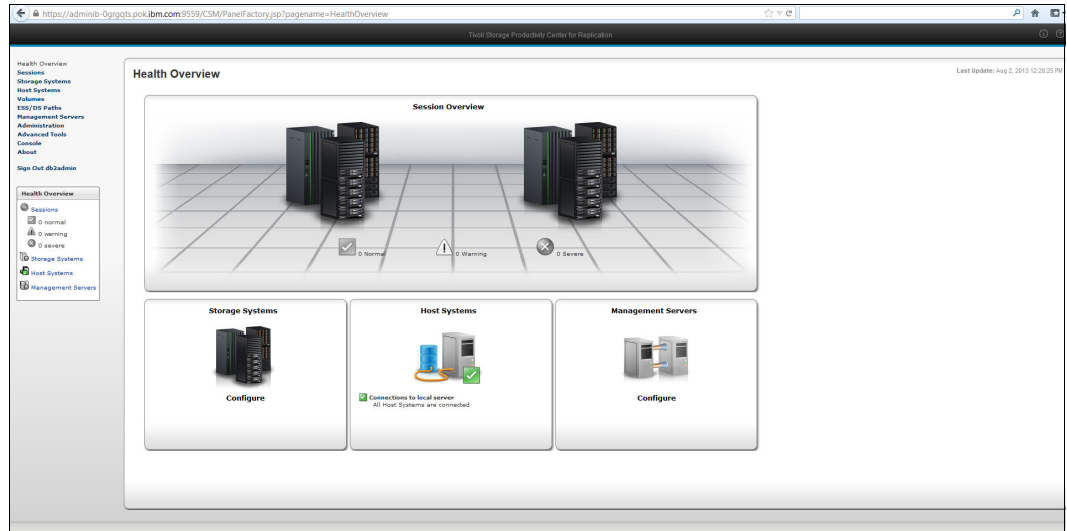


Figure 5-2 Host systems

- Click **Add Host Connection** (Figure 5-3).

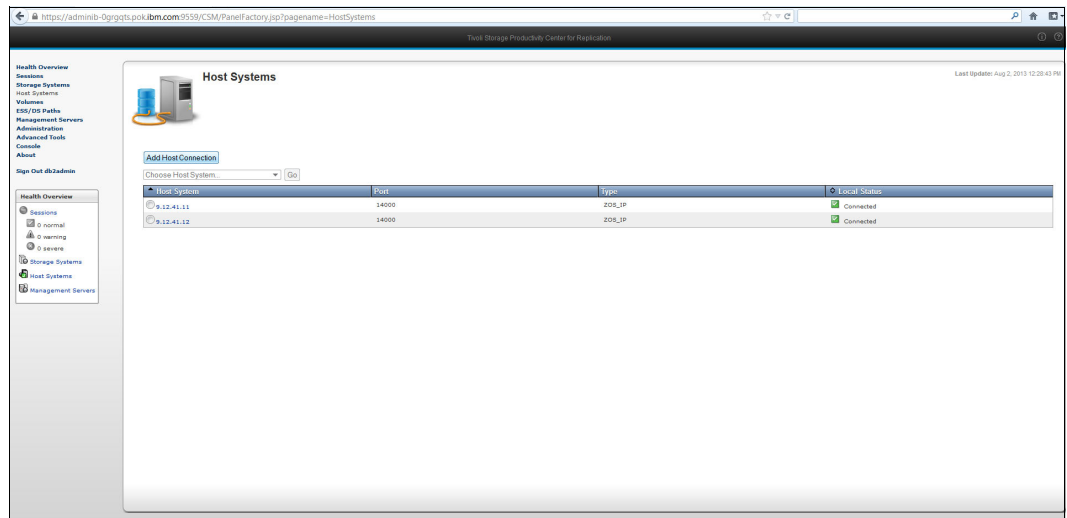


Figure 5-3 Adding a host connection

4. Add the host connection information (Figure 5-4): connection type (select **z/OS**), host name or IP address, port number from the HyperSwap Management Address Space started task PROC, user name (user ID) and password. If the information is loaded successfully, Local Status indicates Connected.

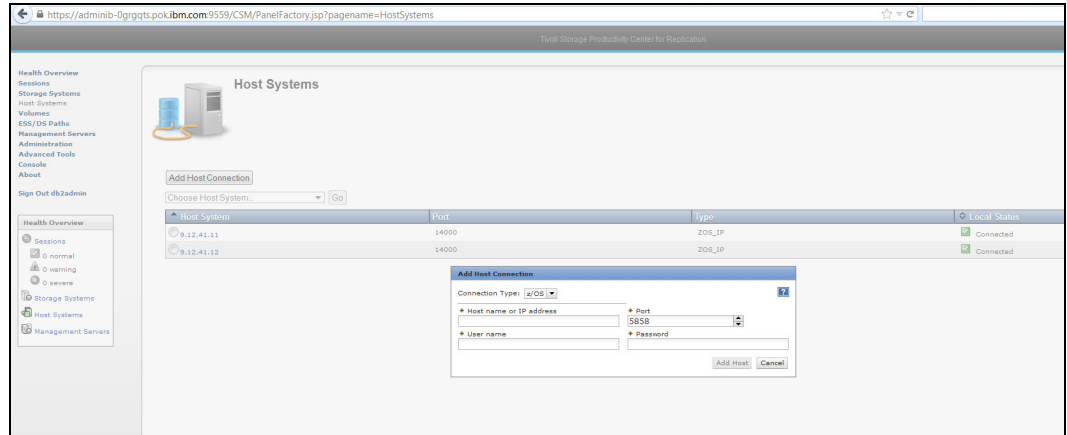


Figure 5-4 Host systems panel

The HyperSwap Management Address Space procedure is typically member HSIB in 'SYS1.PROCLIB'. Example 5-1 shows a sample HSIB member.

Example 5-1 Sample HyperSwap SYS1.PROCLIB member HSIB

```
//HSIB      PROC PORT=14000
//STEP      EXEC PGM=IOSHMCTL,TIME=NOLIMIT,REGION=0M,
//          PARM='SOCKPORT=&PORT'
```

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this paper.

IBM Redbooks

The following IBM Redbooks publications offer more information about the topic in this paper. Note that some publications referenced in this list might be available in softcopy only.

- ▶ *IBM Tivoli Storage Productivity Center for Replication for System z*, SG24-7563
- ▶ *Tivoli Storage Productivity Center for Replication for Open Systems*, SG24-8149

You can search for, view, download or order these documents and other Redbooks, Redpapers, Web Docs, draft and additional materials, at the following website:

ibm.com/redbooks

Other publications

These publications are also relevant as further information sources:

- ▶ *z/OS V1R13.0 Communications Server IP Configuration Reference*, SC31-8776
<http://pic.dhe.ibm.com/infocenter/zos/v1r13/index.jsp?topic=%2Fcom.ibm.zos.r13.halz001%2Ftoc.htm>
- ▶ *IBM WebSphere Host On-Demand Version 10.0: Planning, Installing, and Configuring Host On-Demand*, SC31-6301-04
<http://www-01.ibm.com/support/docview.wss?uid=pub1sc31630104>
- ▶ *z/OS Security Server RACF Command Language Reference*, SA22-7687
- ▶ *z/OS Security Server RACF System Programmer's Guide*, SA22-7681
- ▶ *z/OS Security Server RACF Security Administrator's Guide*, SA22-7683.
- ▶ *z/OS Internet Library* (view and print online versions of the z/OS publications:
<http://www.ibm.com/systems/z/os/zos/bkserv/>

Online resources

These websites are also relevant as further information sources:

- ▶ Tivoli Storage Productivity Center
<http://pic.dhe.ibm.com/infocenter/tivihelp/v59r1/index.jsp>
- ▶ Tivoli Storage Productivity Center Support Portal
http://www-947.ibm.com/support/entry/portal/product/tivoli/tivoli_storage_productivity_center?productContext=1039251977

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services



Securing IBM HyperSwap and IBM Tivoli Storage Productivity Center for Replication Communication Using AT-TLS



Enable secure communication through the Internet

Create a policy file for AT-TLS and generate RACF certificates

See the procedure to verify communication setup

IBM Tivoli Storage Productivity Center for Replication V5.2 can establish a connection to an IBM z/OS server from a Tivoli Storage Productivity Center for Replication distributed installation or from another z/OS installation that can reside outside the sysplex that is being managed. This IBM Redpaper publication describes the steps to connect to, configure, and manage z/OS IBM HyperSwap from Tivoli Storage Productivity Center for Replication V5.2.

This paper helps you configure IBM HyperSwap to communicate with IBM Tivoli Storage Productivity Center for Replication securely through the Internet by using Secure Sockets Layer (SSL) or Application Transparent Transport Layer Security (TLS).

This document is intended for storage administrators responsible for configuring and maintaining the Tivoli Storage Productivity Center for Replication environment.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks