Leveraging z/OS Communications Server Application Transparent Transport Layer Security (AT-TLS) for a Lower Cost and More Rapid TLS Deployment

SHARE Session 16082

August 7, 2014

Lin Overby – overbylh@us.ibm.com

z/OS Communications Server

Trademarks, notices, and disclaimers

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

- Advanced Peer-to-Peer Networking®
- AIX®
- alphaWorks®
- AnyNet®
- AS/400®
- BladeCenter®
- Candle®
- CICS®
- DataPower®
- DB2 Connect
- DB2®
- DRDA®
- e-business on demand®
- e-business (logo)
- e business(logo)®
- ESCON®
- FICON®
- GDDM®
- GDPS®
- Geographically Dispersed Parallel Sysplex
- HiperSockets
- HPR Channel Connectivity
- HyperSwap
- i5/OS (logo)
- i5/OS®
- IBM eServer
- IBM (logo)®
- IBM®
- IBM zEnterprise™ System
- IMS
- InfiniBand ®
- IP PrintWay
- IPDS
- iSeries
- LANDP®
- Language Environment®
- MQSeries®
- MVS
- NetView®
- OMEGAMON®
- Open Power
- OpenPower
- Operating System/2®
- Operating System/400®
- OS/2®
- OS/390®
- OS/400®
- Parallel Sysplex®
- POWER®
- POWER7®
- PowerVM
- PR/SM
- pSeries®
- RACF®
- Rational Suite®
- Rational®
- Redbooks
- Redbooks (logo)
- Sysplex Timer®
- System i5
- System p5
- System x®
- System z®
- System z9®
- System z10
- Tivoli (logo)®
- Tivoli®
- VTAM®
- WebSphere®
- xSeries®
- z9®
- z10 BC
- z10 EC
- zEnterprise
- zSeries®
- z/Architecture
- z/OS®
- z/VM®
- z/VSE

* All other products may be trademarks or registered trademarks of their respective companies.

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

- Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.
- Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license there from.
- Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
- Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.
- InfiniBand is a trademark and service mark of the InfiniBand Trade Association.
- Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
- UNIX is a registered trademark of The Open Group in the United States and other countries.
- Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.
- ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.
- IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency, which is now part of the Office of Government Commerce.

Notes:

- Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.
- IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.
- All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.
- This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.
- All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.
- Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the 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.
- Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

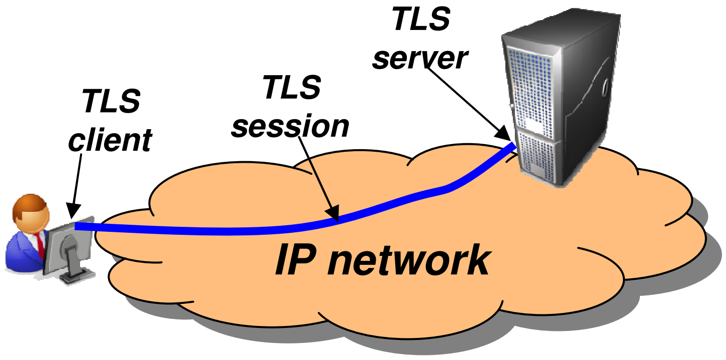
Refer to www.ibm.com/legal/us for further legal information.

Agenda

- **SSL/TLS Overview**
- **What is AT-TLS?**
- **Why use AT-TLS?**
- **How does AT-TLS work?**
- **Configuring AT-TLS**

Transport Layer Security (TLS/SSL) overview

- Transport Layer Security (TLS) is defined by the IETF **
  - Based on Secure Sockets Layer (SSL)
    - TLS defines SSL as a version of TLS for compatibility
- Provides secure connectivity between two TLS security session endpoints
  - TLS session
- Full application payload encryption and data authentication / integrity
- TLS security session endpoint plays either a client or server role
- Session endpoint authentication typically via X.509 certificates
  - Server authentication required
  - Client authentication optional (mutual authentication)



Full application payload encryption

TLS/SSL encryption:

SrcIP	DestIP	SrcPort	DestPort	Data
192.168.100.1	192.168.1.1	50002	443	@%$#*&&^^!:"J)*GVM><

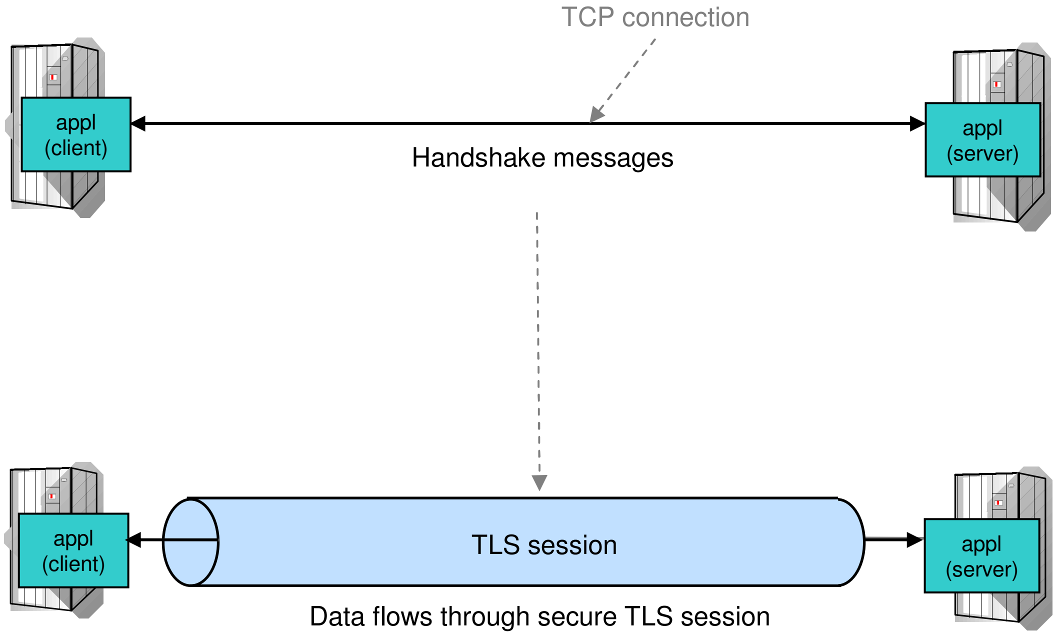
**** For our purposes, SSL and TLS are equivalent and one term implies the other**

TLS/SSL protocol basics

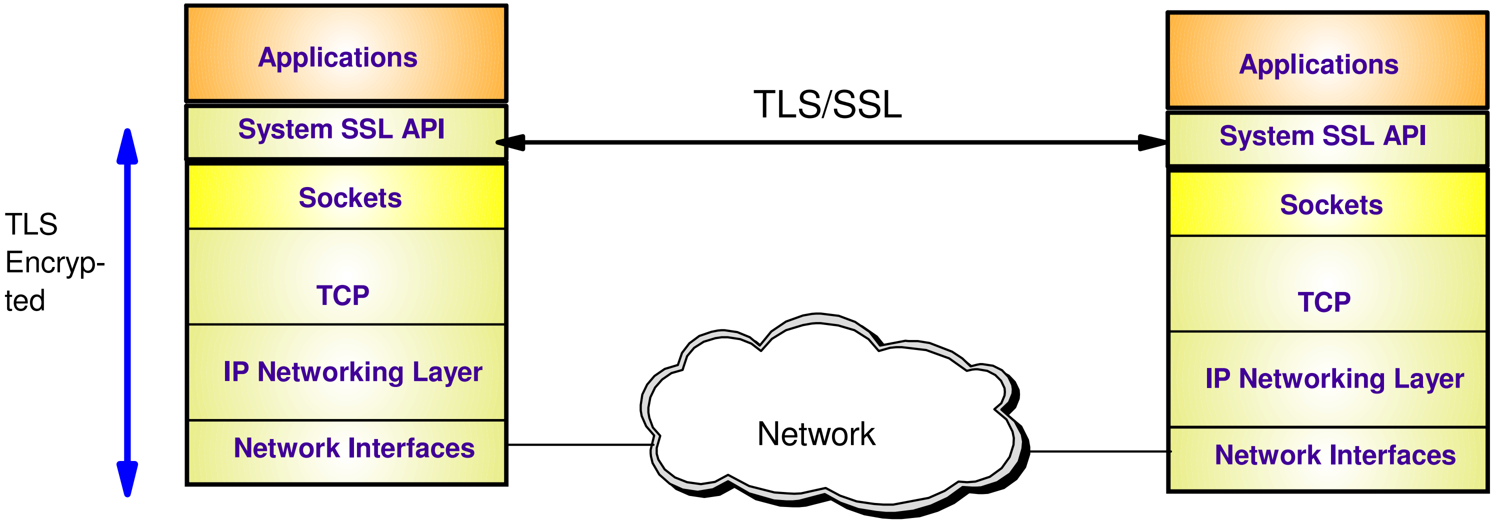
1 Client application initiates TLS handshake which authenticates the server (and, optionally, client) and negotiates a cipher suite to be used to protect data

Upon successful completion of the handshake, a secure TLS session exists for the application partners

2 Data flows through secure session using symmetric encryption and message authentication negotiated during handshake



Transport Layer Security enablement



- TLS traditionally provides security services as a socket layer service
  - TLS requires reliable transport layer,
    - Typically TCP (but architecturally doesn't have to be TCP)
  - UDP applications cannot be enabled with traditional TLS
    - There is now a TLS variant called Datagram Transport Layer Security (DTLS) which is defined by the IETF for unreliable transports
- On z/OS, System SSL (a component of z/OS Cryptographic Services) provides an API library for TLS-enabling your C and C++ applications
- Java Secure Sockets Extension (JSSE) provides libraries to enable TLS support for Java applications
  - However, there is an easier way…

… Application Transparent TLS!

z/OS Application Transparent TLS overview

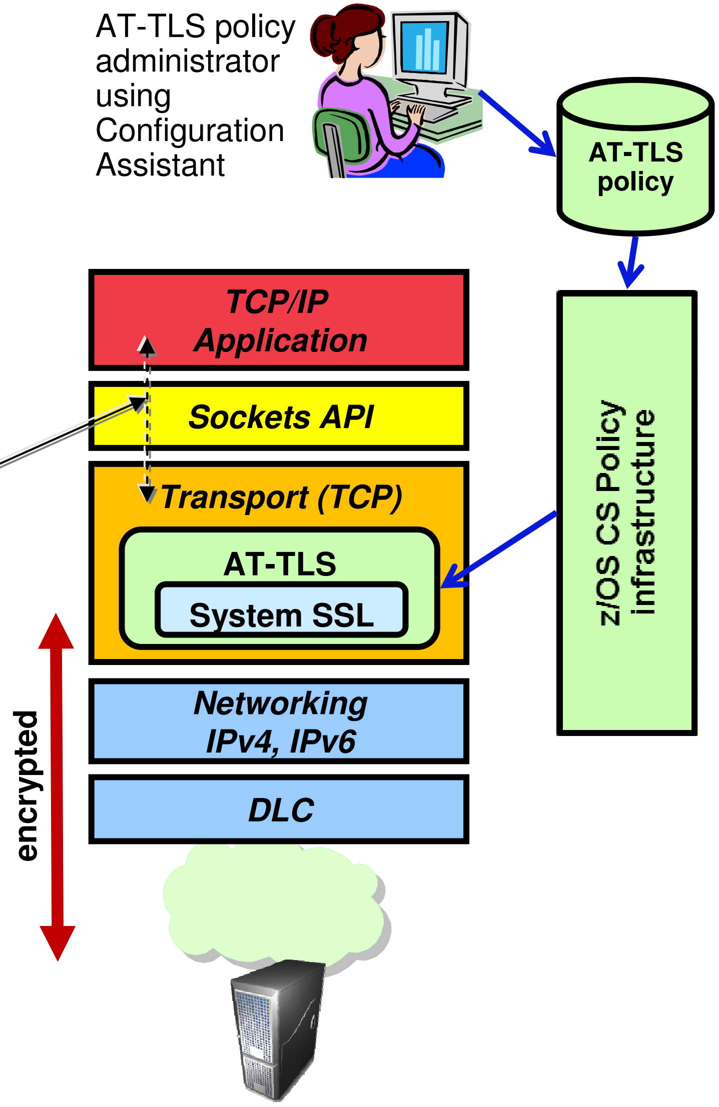
- **Stack-based TLS**
  - TLS process performed in TCP layer (via System SSL) without requiring any application change (transparent)
  - AT-TLS policy specifies which TCP traffic is to be TLS protected based on a variety of criteria
    - Local address, port
    - Remote address, port
    - Connection direction
    - z/OS userid, jobname
    - Time, day, week, month

- **Application transparency**
  - Can be fully transparent to application
  - An optional API allows applications to inspect or control certain aspects of AT-TLS processing – "application-aware" and "application-controlled" AT-TLS, respectively

- **Available to TCP applications**
  - Includes CICS Sockets
  - Supports all programming languages except PASCAL

- **Supports standard configurations**
  - z/OS as a client or as a server
  - Server authentication (server identifies self to client)
  - Client authentication (both ends identify selves to other)

- **Uses System SSL for TLS protocol processing**
  - Remote endpoint sees an RFC-compliant implementation
  - interoperates with other compliant implementations



Some z/OS applications that use AT-TLS

- CommServer applications
  - TN3270 Server
  - FTP Client and Server
  - CSSMTP
  - Load Balancing Advisor
  - IKE NSS client
  - NSS server
  - Policy agent
- DB2 DRDA
- IMS-Connect
- JES2 NJE
- Tivoli Netview applications
  - MultiSystem Manager
  - NetView Management Console
- RACF Remote Sharing Facility
- CICS Sockets applications
- 3rd Party applications
- Customer applications

Advantages of using AT-TLS

- **Reduce costs**
  - Application development
    - Cost of System SSL integration
    - Cost of application's TLS-related configuration support
  - Consistent TLS administration across z/OS applications
  - Gain access to new features with little or no incremental development cost

- **Complete and up-to-date exploitation of System SSL features**
  - AT-TLS makes the vast majority of System SSL features available to applications
  - AT-TLS keeps up with System SSL enhancements – as new features are added, your applications can use them by changing AT-TLS policy, not code

- **Ongoing performance improvements**

  Focus on efficiency in use of System SSL

- **Great choice if you haven't already invested in System SSL integration**

  Even if you have, consider the long-term cost of keeping up vs. short term cost of conversion

AT-TLS application types

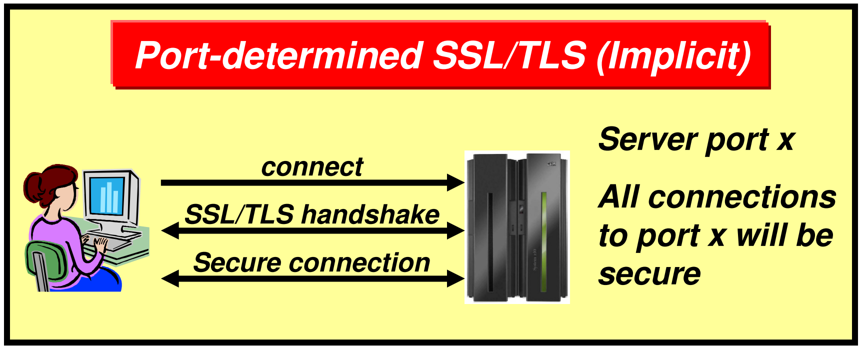
- **Not enabled**
  - No policy or policy explicitly disables AT-TLS for application traffic
  - Application may optionally use System SSL directly
  - Applications that use the Pascal API and Web Fast Response Cache Accelerator (FRCA) fall into this category

- **Basic**
  - Policy enables AT-TLS for application traffic
  - Application is unchanged and unaware of AT-TLS
  - Application protocol unaffected by use of AT-TLS (think HTTP vs. HTTPS)

- **Aware**
  - Policy enables AT-TLS for application traffic
  - Application uses the SIOCTTLSCTL ioctl to extract AT-TLS information such as partner certificate, negotiated version and cipher, policy status, etc.

- **Controlling**
  - Policy enables AT-TLS and specifies ApplicationControlled ON for application traffic
  - Application protocol may negotiate the use of TLS in cleartext with its partner
  - Application uses the SIOCTTLSCTL ioctl to extract AT-TLS information (like an aware application) and to control TLS operations:
    - Start secure session
    - Reset session
    - Reset cipher

SSL/TLS application types



Port-determined SSL/TLS (Implicit)

connect

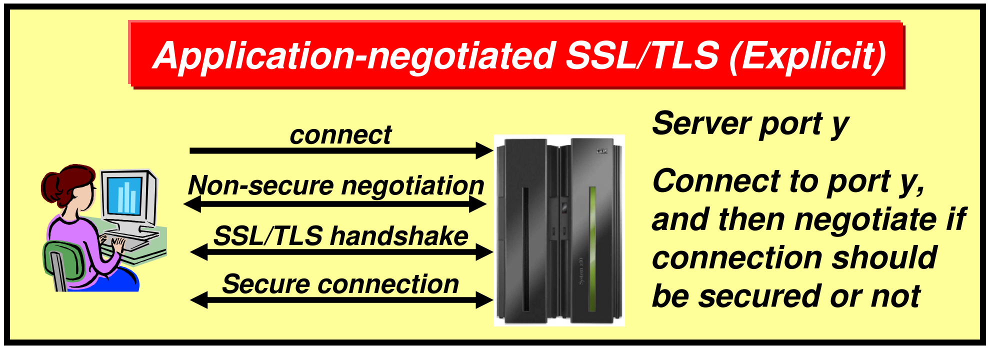
SSL/TLS handshake

Secure connection

Server port x

All connections to port x will be secure

- As soon as a connection has been established with the server, the SSL/TLS handshake starts
- Examples are the HTTPS port (443), and FTP’s secure port (990)
- AT-TLS considerations:
  - Can be done totally transparent to application code
    - This is referred to as an AT-TLS "Basic" application
  - Optionally the application may query SSL/TLS attributes, such as client user ID (if client authentication is used, cipher suite in use, etc)
    - This is referred to as an AT-TLS "Aware" application



Application-negotiated SSL/TLS (Explicit)

connect

Non-secure negotiation

SSL/TLS handshake

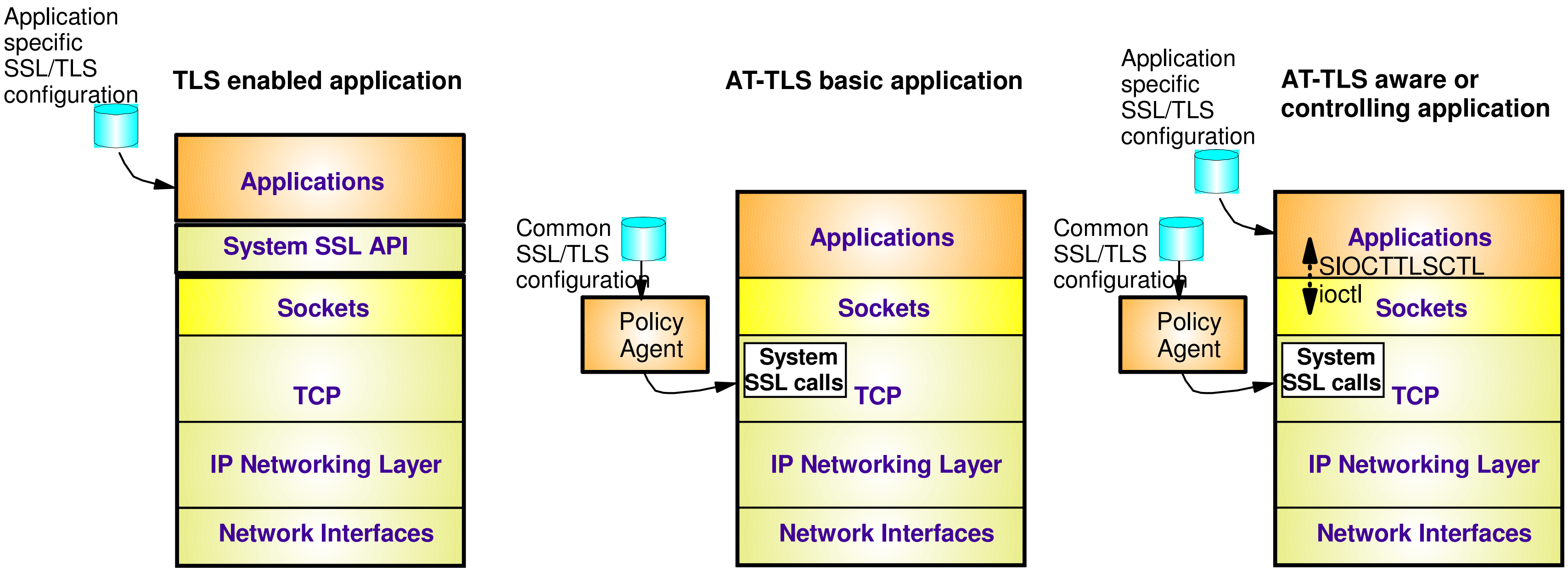
Secure connection

Server port y

Connect to port y, and then negotiate if connection should be secured or not

- Application protocol includes verbs to negotiate security protocol and options
- Examples are FTP that uses the AUTH FTP command to negotiate use of SSL/TLS or Kerberos, and in some cases a TN3270 server port (Conntype NegtSecure)
- AT-TLS considerations:
  - Application needs to "tell" AT-TLS when to start the SSL/TLS handshake
    - This is referred to as an AT-TLS "Controlling" application
  - Otherwise, use of AT-TLS is transparent to application
  - Optionally the application may query SSL/TLS attributes, such as client user ID (if client authentication is used, cipher suite in use, etc)

TLS configuration cases by application type

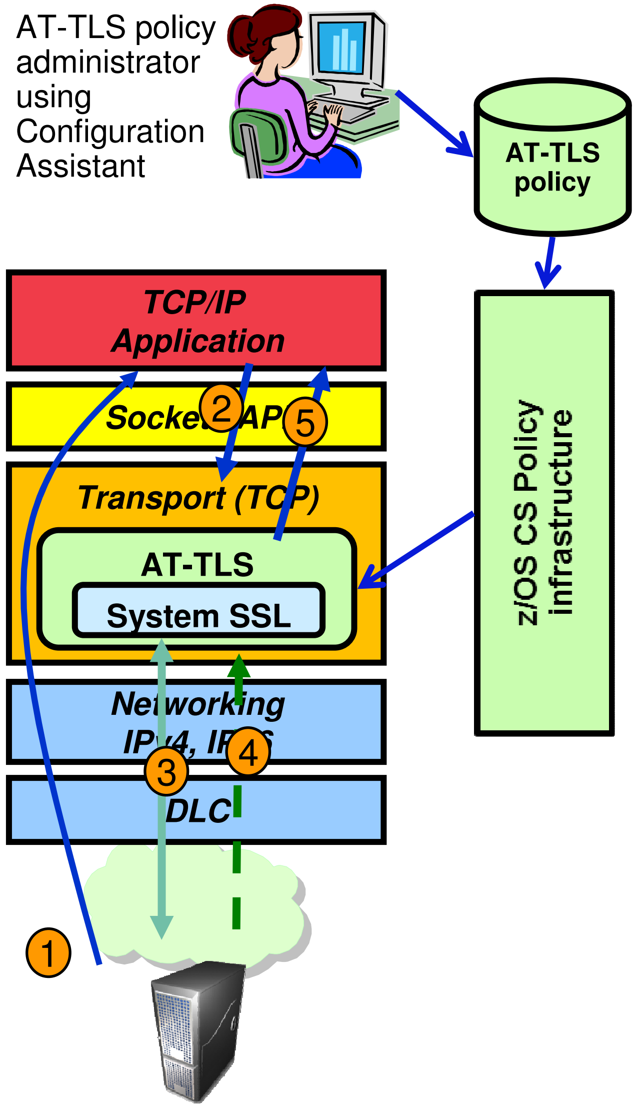


- TLS enabled application
  - Each application has its own configuration to control security policy and TLS functions
- AT-TLS basic application
  - All applications' security policy and TLS functions are governed by a single, consistent AT-TLS policy system-wide
- AT-TLS aware or controlling applications
  - Application specific policy retained but reduced to what application needs for awareness or controlling functions
  - AT-TLS policy continues to control overall AT-TLS function for the application

AT-TLS operation (z/OS as server)

Setup: AT-TLS policy is configured and deployed for the TCP application and the TCP application is started.

1. Client connects to server and connection is established
2. After accepting the new connection, the server issues a read request on the socket. The TCP layer checks AT-TLS policy and sees that AT-TLS protection is configured for this connection. As such, it prepares for the client-initiated TLS handshake
3. The client initiates the SSL handshake and the TCP layer invokes System SSL to perform the TLS handshake under identity of the server.
4. Client sends data traffic under protection of the new TLS session
5. TCP layer invokes System SSL to decrypt the data and then delivers the cleartext inbound data to the server



→ Unencrypted (cleartext) flows

→ SSL/TLS handshake flows

– – ► SSL/TLS-secured (encrypted) flows

AT-TLS operation (z/OS as client)

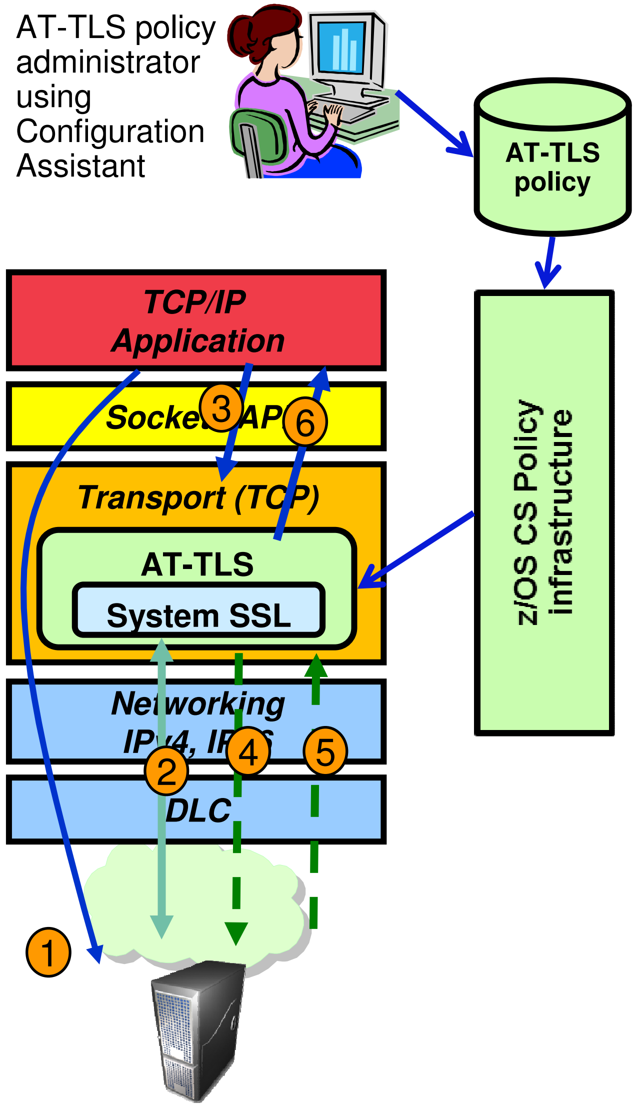
Setup: AT-TLS policy is configured and deployed for the TCP application and the TCP application is started.

1. z/OS client connects out to server and connection is established
2. TCP layer invokes System SSL to perform the TLS handshake under identity of the client application
3. z/OS client sends data to server
4. TCP layer invokes System SSL to encrypt queued data and then sends it to server
5. Server sends encrypted data, TCP layer invokes System SSL to decrypt it
6. TCP delivers inbound data to z/OS client in the clear

→ Unencrypted (cleartext) flows

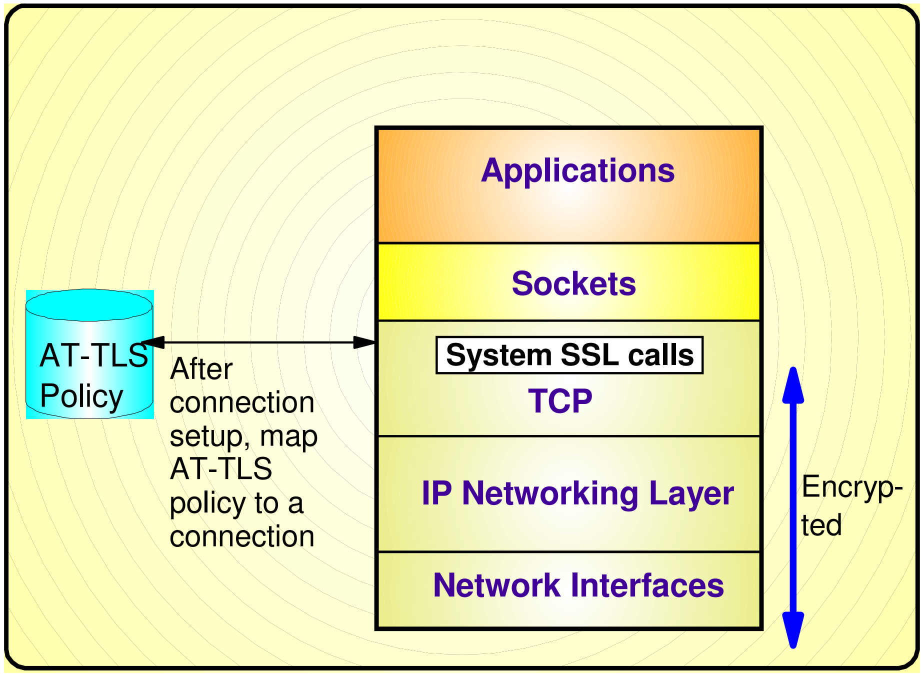
→ SSL/TLS handshake flows

– – ▶ SSL/TLS-secured (encrypted) flows



Mapping AT-TLS policy to a TCP connection

- An AT-TLS policy rule describes TLS requirements for a TCP connection
- Policy rule is mapped to a connection based on policy condition
  - – TCP/IP resource attributes
  - – Connection type attributes
  - – Local application attributes
- An AT-TLS policy rule is mapped to a connection at well defined points
  - – Outbound Connect
  - – First Select/Send/Receive after Accept
  - – SIOCTTLSCTL ioctl
- If a rule match is found, TCP/IP stack provides TLS protocol control based on the policy action
- Alternate method of mapping policy to a connection
  - – Secondary Map
    - • Used for applications that have one or more “secondary” connections and one “primary” connection
    - • Examples: FTP, rsh, rexec



AT-TLS policy conditions

Criteria	Description
Local address	Local IP address
Remote address	Remote IP address
Local port	Local port or ports
Remote port	Remote port or ports
Connection direction	• Inbound (applied to first Select, Send, or Receive after Accept) • Outbound (applied to Connect) • Both
User ID	User ID of the owning process or wildcard user ID
Jobname	Jobname of the owning application or wildcard jobname
Time, Day, Week, Month	When filter rule is active

AT-TLS policy actions

Criteria	Description
TLS enablement	Specifies whether TLS is enabled for connection matching the policy rule
TLS/SSL versions allowed	SSLv2, SSLv3, TLSv1, TLSv1.1, TLSv2.0
Cipher suites	Set of potential cryptographic algorithms (in order of preference) that this TLS server or client will accept during the TLS handshake
Role	• TLS client • TLS server • TLS server with client authentication
Client authentication type	• Passthru (bypass checking) • Required • Full (Accepted if provided by client) • SAFCheck
Authentication information	• Keyring identifier • Certificate label used for authentication • LDAP for certificate revocation list (CRL) processing
Data trace	Specifies whether to trace cleartext in datatrace or ctrace
AT-TLS trace levels	Specifies level of tracing
Handshake timeout	Time to wait for handshake to complete
Session key lifetime	When session key has been used this specified time period, a new session key must be created
Session ID requirements	Session ID cache size, Session ID timeout, Use sysplex-wide session ID cache
Secondary map used	Specifies whether a matching connection should be used as a "primary" connection in the "secondary policy mapping method"

AT-TLS support for TLS v1.2 and Related Features

…Added in z/OS V2R1

- Transport Layer Security (TLS) Renegotiation Extension (RFC 5746):
  - Provides a mechanism to protect peers that permit re-handshakes
  - When supported, it enables both peers to validate that the re-handshake is truly a continuation of the previous handshake

- Support Elliptic Curve Cryptography (ECC)
  - Twenty new ECC cipher suites
    - ECC cipher suites for TLS (RFC 4492)

- TLS Protocol Version 1.2 (RFC 5246):
  - Twenty-one new cipher suites
    - 11 new HMAC-SHA256 cipher suites
    - 10 new AES-GCM cipher suites

- Support for Suite B cipher suites (RFC 5430)
  - TLS 1.2 is required
  - ECC is required
  - Suite B has two levels of cryptographic strength that can be selected
    - 128 or 192 bit

AT-TLS configuration task steps

- Obtain x.509 certificates and update RACF keyrings
- Update any application-specific configuration files if necessary
- Enabling use of AT-TLS in the TCP/IP stack configuration
- Create AT-TLS policy using Configuration Assistant for z/OS Communications Server
- Create policy infrastructure using Configuration Assistant application setup task checklist

Obtain x.509 certificates and update RACF keyrings

- Same process as with SSL-enabled applications
  - More information on certificate acquisition, configuration using RACDCERT command in appendix

- Keyrings with certificates and private keys used for TLS sessions are specified in the AT-TLS policy

- Keyring can be specified at a:
  - A system image level
  - Policy rule level

Update any application configuration if needed - FTP example

- Some application configuration changes may be necessary if the application is either AT-TLS aware or AT-TLS controlling
- The FTP server is both AT-TLS aware and controlling
- Example below defines an FTP server that supports SSL/TLS connections, but does not require it
  - It depends on the client sending an AUTH command or not
- SSL/TLS is done by ATTLS in this example

```
EXTENSIONS         AUTH_TLS           ; Enable TLS authentication
TLSMECHANISM       ATTLS              ; Server-specific or ATTLS
SECURE_FTP         ALLOWED            ; Security required/optional
SECURE_LOGIN       NO_CLIENT_AUTH     ; Client authentication
SECURE_PASSWORD    REQUIRED           ; Password requirement
SECURE_CTRLCONN    PRIVATE            ; Minimum level of security CTRL
SECURE_DATACONN    PRIVATE            ; Minimum level of security DATA
TLSRFCLEVEL        RFC4217            ; SSL/TLS RFC Level supported
```

Enabling use of AT-TLS in the TCP/IP stack

- AT-TLS is enabled via a TCPCONFIG parameter

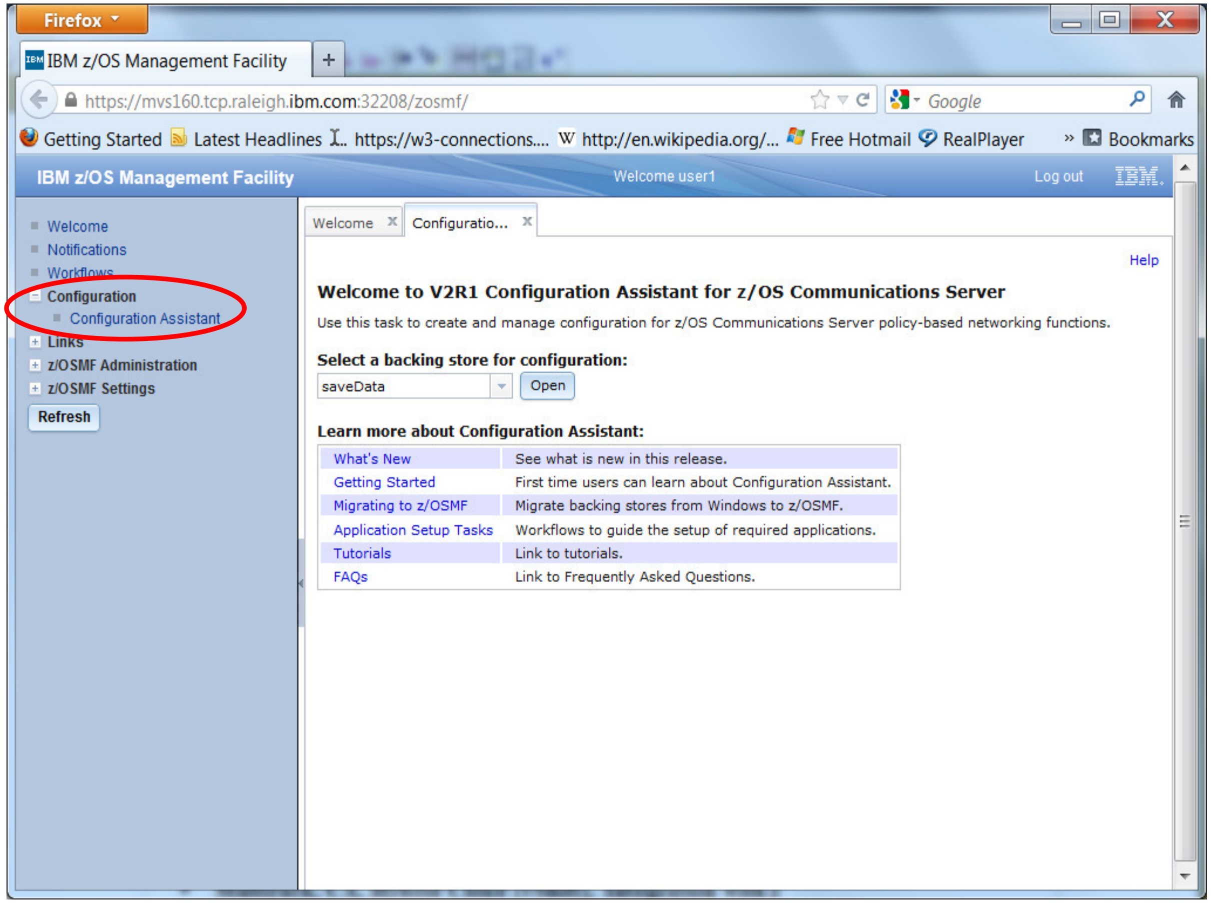
```
TCPConfig TTLS                         ; Enable AT-TLS policies
```

- There may be a short time period between TCP/IP parsing this configuration option and the actual AT-TLS policies being installed into the stack by Policy Agent
  - Since the stack doesn't yet have an AT-TLS policy, it doesn't know which connections to secure
  - What should it do if a new connection is being set up during this short time window?
  - You control that via a SERVAUTH profile:
    - **EZB.INITSTACK.system.stackname**
- When TCP/IP starts with TCPCONFIG TTLS specified, it will issue message EZZ4248E

```
EZZ4248E TCPCS WAITING FOR PAGENT TTLS POLICY
EZZ8771I PAGENT CONFIG POLICY PROCESSING COMPLETE FOR TCPCS : TTLS
EZZ4250I AT-TLS SERVICES ARE AVAILABLE FOR TCPCS
```

- Between messages EZZ4248E and EZZ4250I, the TCP/IP stack will only allow users permitted to the EZB.INITSTACK.system.stack SERVAUTH profile to establish TCP connections.
  - ***Note:*** make sure all your pertinent server address spaces (including PAGENT and OMPROUTE) run under user IDs that are permitted to this profile.

Policy-based network security on z/OS: Configuration Assistant



Welcome to V2R1 Configuration Assistant for z/OS Communications Server

Use this task to create and manage configuration for z/OS Communications Server policy-based networking functions.

Select a backing store for configuration:

saveData Open

Learn more about Configuration Assistant:

What's New	See what is new in this release.
Getting Started	First time users can learn about Configuration Assistant.
Migrating to z/OSMF	Migrate backing stores from Windows to z/OSMF.
Application Setup Tasks	Workflows to guide the setup of required applications.
Tutorials	Link to tutorials.
FAQs	Link to Frequently Asked Questions.

- **Configures:**
  - AT-TLS
  - IPSec and IP filtering
  - IDS
  - Quality of Service
  - Policy-based routing
- **Separate perspectives but consistent model for each discipline**
- **Focus on concepts, not details**
  - what traffic to protect
  - how to protect it
  - De-emphasize low-level details (though they are accessible through advanced panels)
- **z/OSMF-based web interface**
  - Standalone Windows application
    - Not supported after z/OS V1R13
- **Builds and maintains**
  - Policy files
  - Related configuration files
  - JCL procs and RACF directives
- **Supports import of existing policy files**

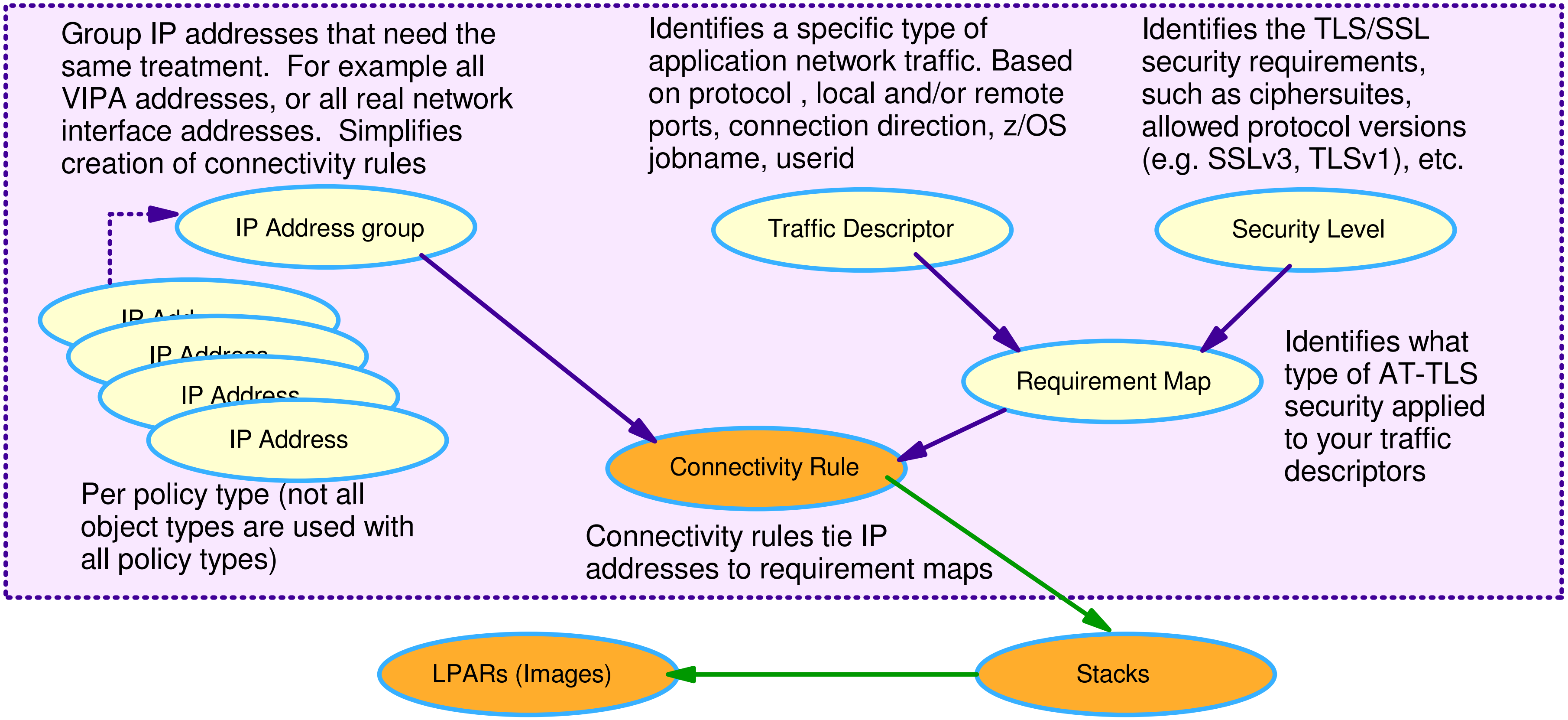
Configuration Assistant policy creation: general approach

- Wizards and dialogs guide you through a top-down approach to configuration
  - ➤ Navigational tree supports a bottom-up approach
    - – Allows an experienced user to bypass wizard screens

- Define system images and TCP/IP stacks
- Define security levels (reusable)
  - – Protection suites (e.g. gold, silver, bronze)
- Define requirements map (reusable)
  - – How to protect common scenarios (e.g. intranet, branch office, business partner)
  - – Set of traffic descriptors linked to security level
- Define connectivity rules
  - – A complete security policy for all traffic between two endpoints
  - – Specified data endpoints linked to a requirements map

Optimizations to this approach are provided for common applications!

Configuration Assistant reusable object model



Group IP addresses that need the same treatment. For example all VIPA addresses, or all real network interface addresses. Simplifies creation of connectivity rules

Identifies a specific type of application network traffic. Based on protocol , local and/or remote ports, connection direction, z/OS jobname, userid

Identifies the TLS/SSL security requirements, such as ciphersuites, allowed protocol versions (e.g. SSLv3, TLSv1), etc.

IP Address group

Traffic Descriptor

Security Level

IP Address

Requirement Map

Identifies what type of AT-TLS security applied to your traffic descriptors

Connectivity Rule

Per policy type (not all object types are used with all policy types)

Connectivity rules tie IP addresses to requirement maps

LPARs (Images)

Stacks

1. Create system image and TCP/IP stack image
2. Create one or more Requirement Maps to define desired security for common scenarios (e.g. intranet, branch office, business partner)
   - Create or reuse Security Levels to define security actions
   - Create or reuse Traffic descriptors to define application ports to secure
3. Create one or more Connectivity Rules between Data Endpoints (IP addresses) and associate with a configured Requirement Map

AT-TLS rule simplification with “pre-defined rules”

- The Configuration Assistant provides predefined AT-TLS connectivity rules for common applications configured for each stack so that policy rules for common applications can be configured in a few clicks.

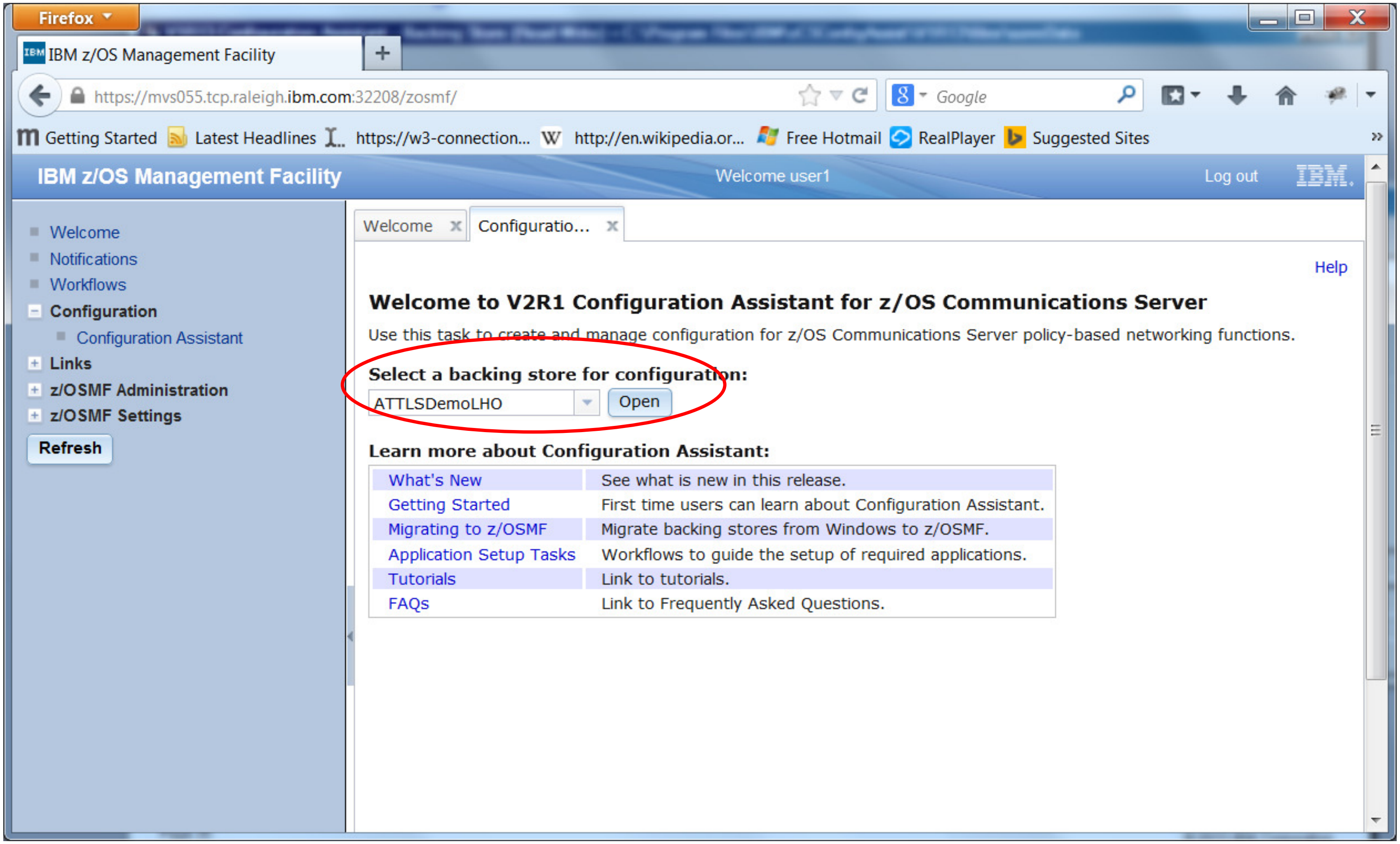
- In most cases, these rules need no modification and can be enabled for immediate use.

- Each rule defines an application with default port settings, key ring, and is associated with a default security level.

- The administrator can easily enable the rules they want to have in their policy and install the generated flat file.

The examples that follow use the pre-defined rule approach….

Open the backing store



Welcome to V2R1 Configuration Assistant for z/OS Communications Server

Use this task to create and manage configuration for z/OS Communications Server policy-based networking functions.

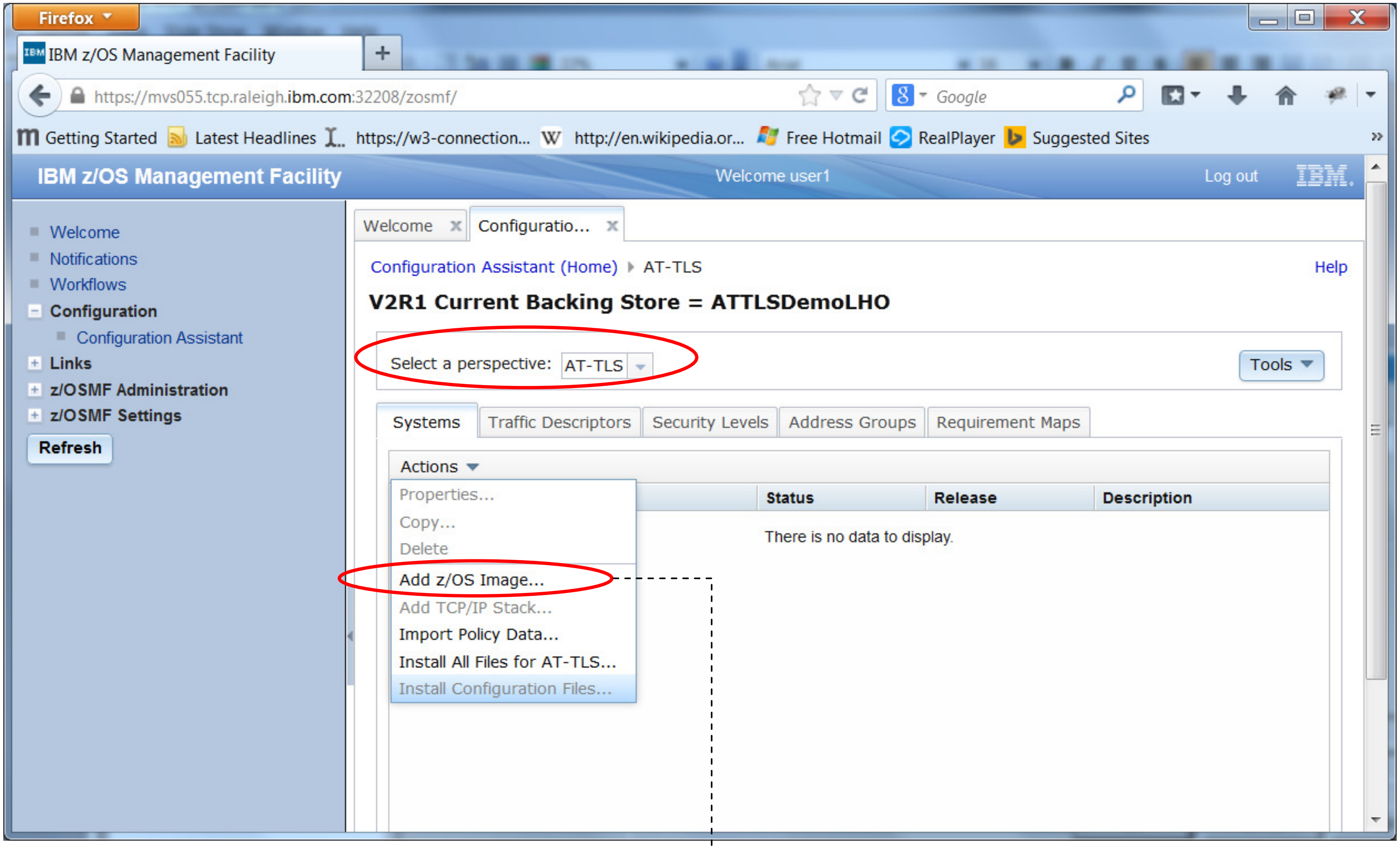
Select a backing store for configuration:

ATTLSDemoLHO ▾ Open

Learn more about Configuration Assistant:

What's New	See what is new in this release.
Getting Started	First time users can learn about Configuration Assistant.
Migrating to z/OSMF	Migrate backing stores from Windows to z/OSMF.
Application Setup Tasks	Workflows to guide the setup of required applications.
Tutorials	Link to tutorials.
FAQs	Link to Frequently Asked Questions.

Select a perspective (AT-TLS)



Configuration Assistant (Home) ▸ AT-TLS

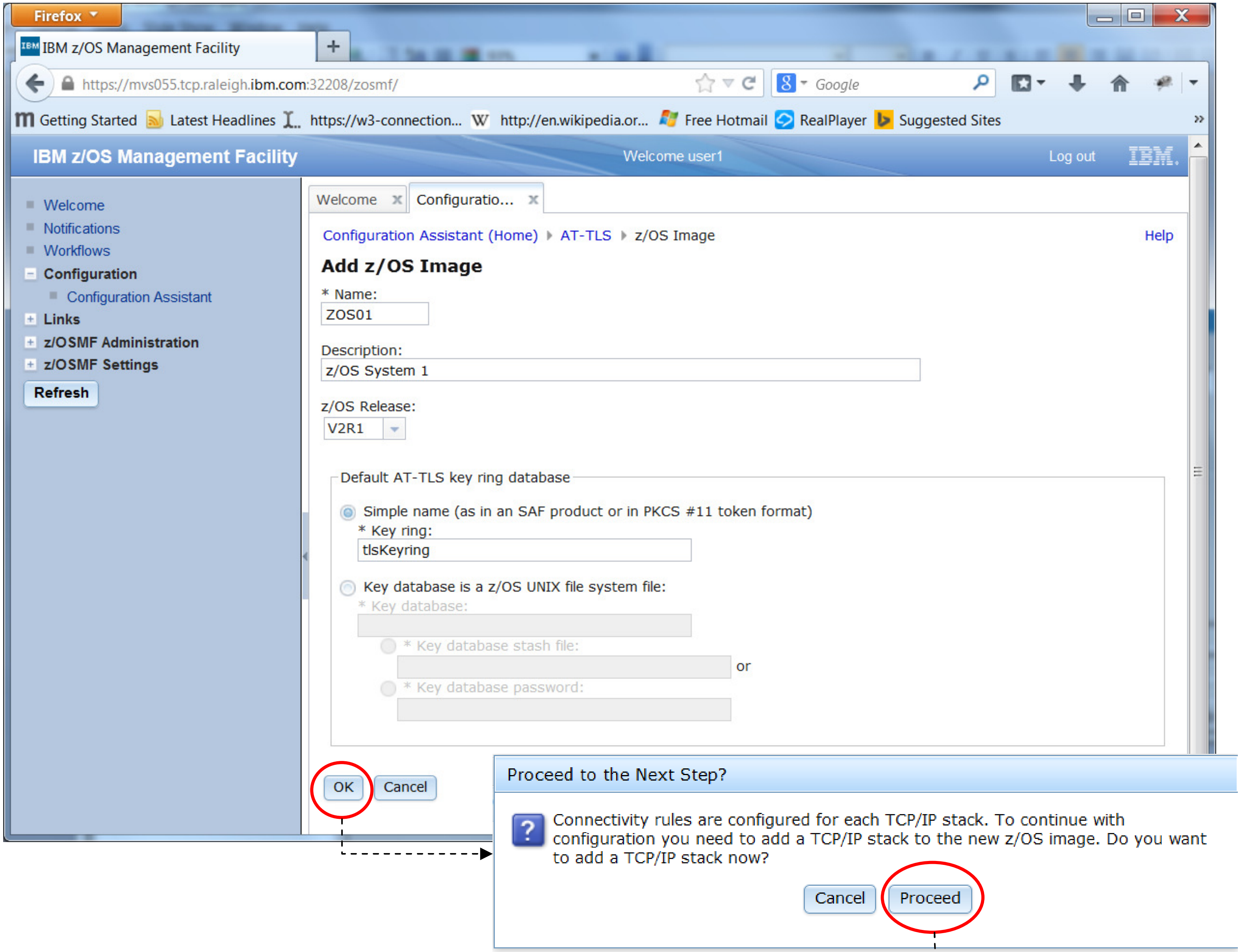
V2R1 Current Backing Store = ATTLSDemoLHO

Select a perspective: AT-TLS

There is no data to display.

Next page

Add a z/OS image and configure default key ring at image level



Add z/OS Image

* Name: ZOS01

Description: z/OS System 1

z/OS Release: V2R1

Default AT-TLS key ring database

● Simple name (as in an SAF product or in PKCS #11 token format)

* Key ring: tlsKeyring

○ Key database is a z/OS UNIX file system file:

* Key database:

○ * Key database stash file:

or

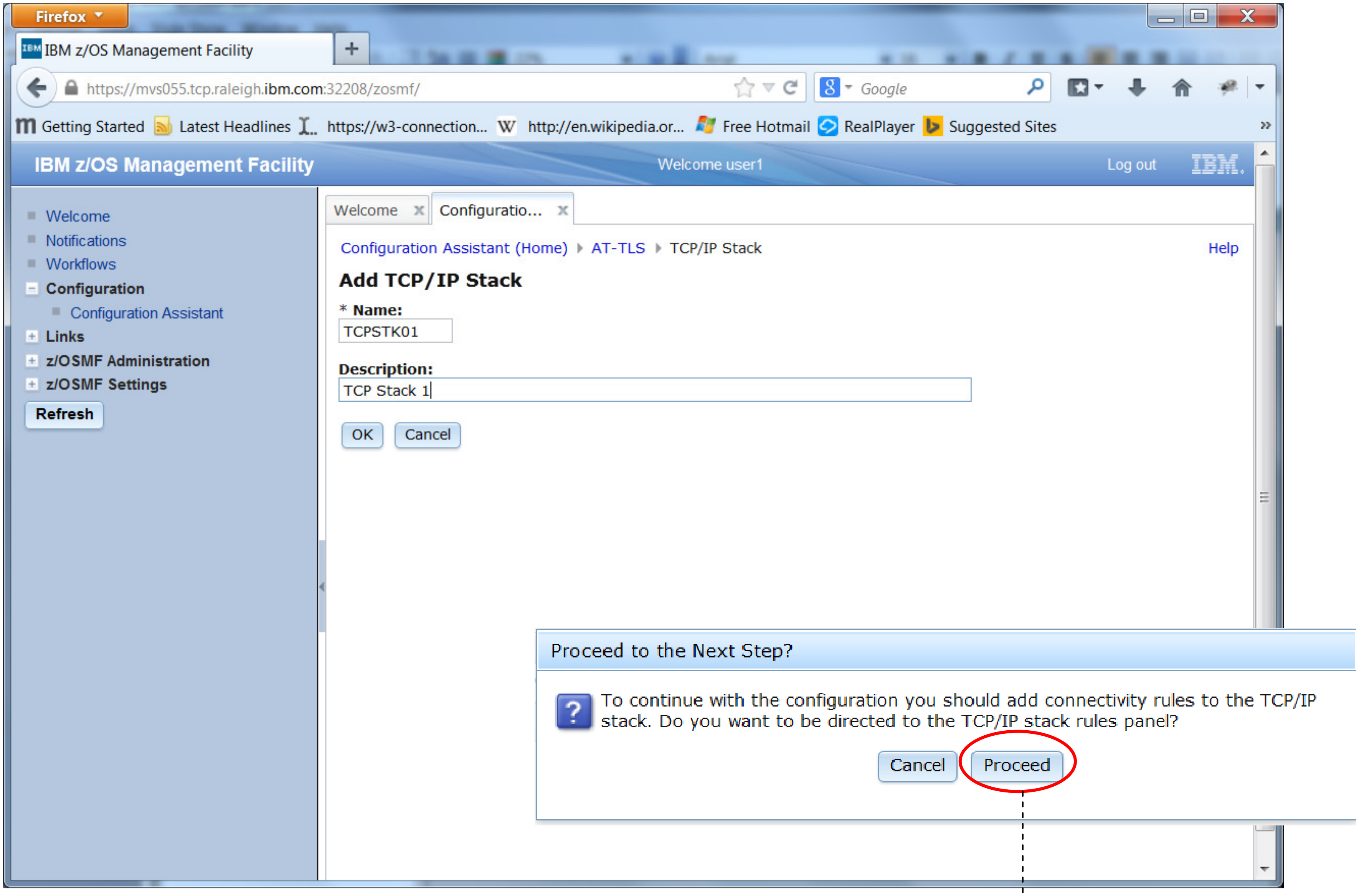
○ * Key database password:

Proceed to the Next Step?

Connectivity rules are configured for each TCP/IP stack. To continue with configuration you need to add a TCP/IP stack to the new z/OS image. Do you want to add a TCP/IP stack now?

Next page

Add a TCP/IP stack



Add TCP/IP Stack

* Name: TCPSTK01

Description: TCP Stack 1

OK | Cancel

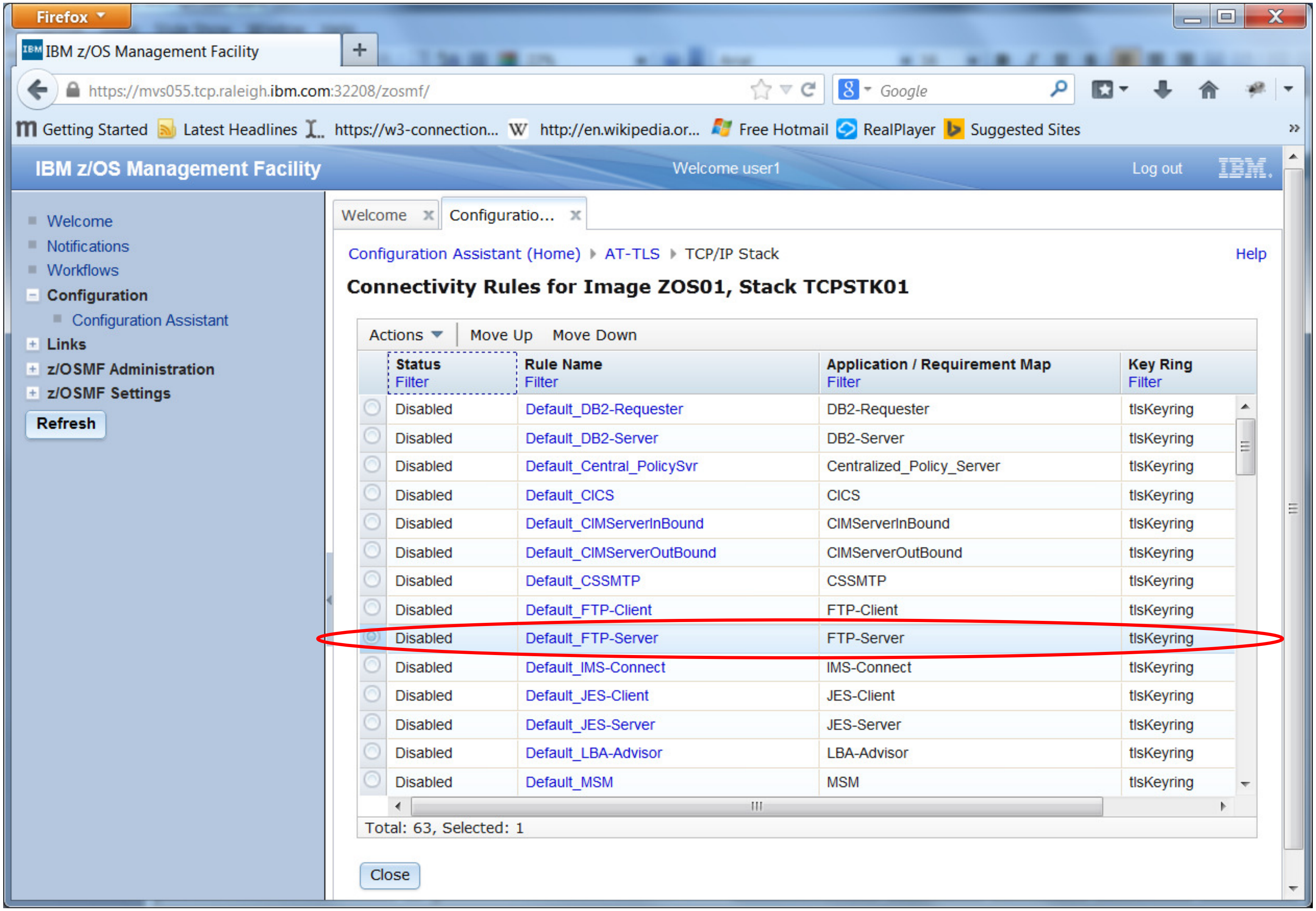
Proceed to the Next Step?

To continue with the configuration you should add connectivity rules to the TCP/IP stack. Do you want to be directed to the TCP/IP stack rules panel?

Cancel | Proceed

Next page

Examining the FTP server pre-defined connectivity rule



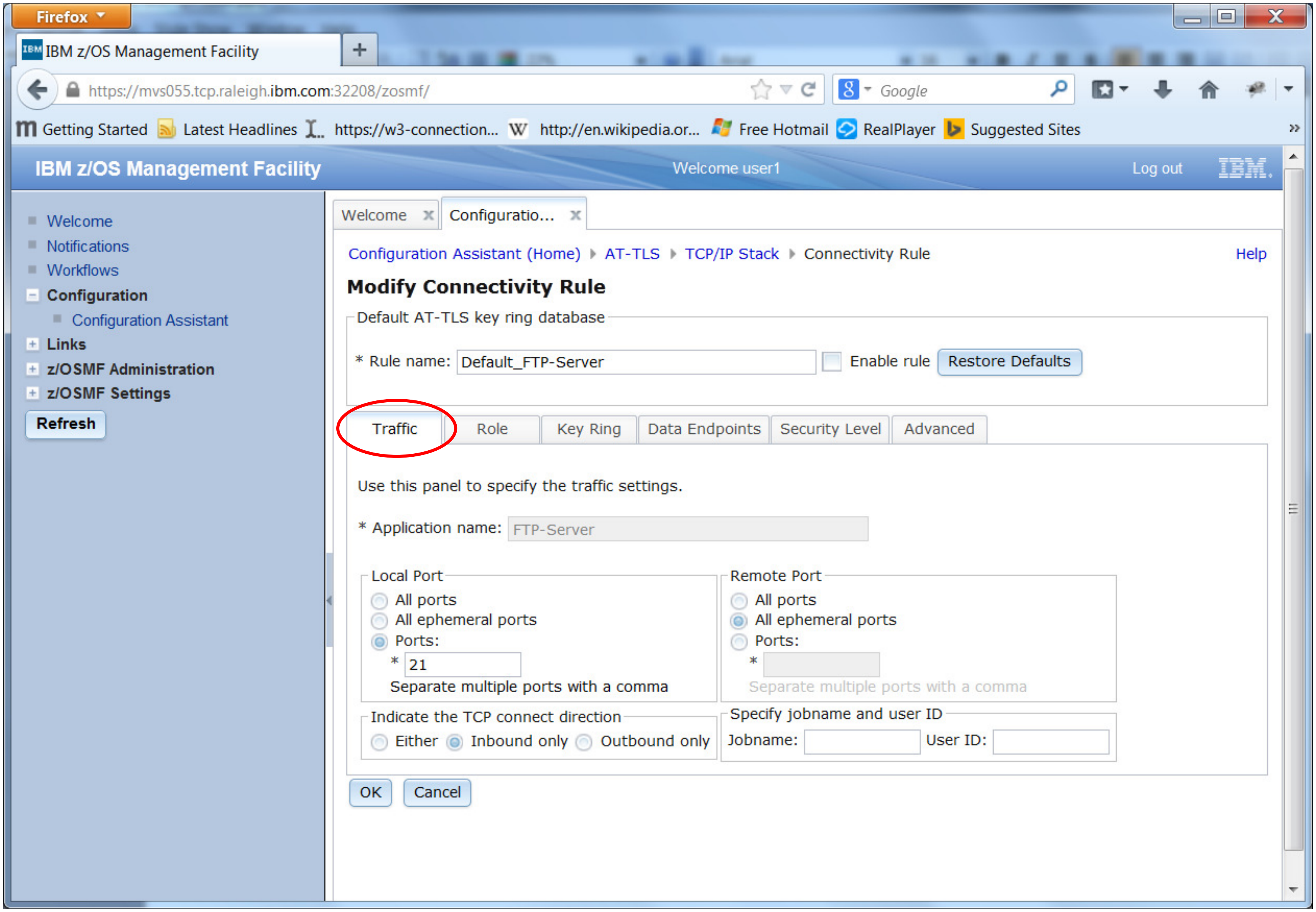
Configuration Assistant (Home) ▸ AT-TLS ▸ TCP/IP Stack

Connectivity Rules for Image ZOS01, Stack TCPSTK01

	Status Filter	Rule Name Filter	Application / Requirement Map Filter	Key Ring Filter
○	Disabled	Default_DB2-Requester	DB2-Requester	tlsKeyring
○	Disabled	Default_DB2-Server	DB2-Server	tlsKeyring
○	Disabled	Default_Central_PolicySvr	Centralized_Policy_Server	tlsKeyring
○	Disabled	Default_CICS	CICS	tlsKeyring
○	Disabled	Default_CIMServerInBound	CIMServerInBound	tlsKeyring
○	Disabled	Default_CIMServerOutBound	CIMServerOutBound	tlsKeyring
○	Disabled	Default_CSSMTP	CSSMTP	tlsKeyring
○	Disabled	Default_FTP-Client	FTP-Client	tlsKeyring
●	Disabled	Default_FTP-Server	FTP-Server	tlsKeyring
○	Disabled	Default_IMS-Connect	IMS-Connect	tlsKeyring
○	Disabled	Default_JES-Client	JES-Client	tlsKeyring
○	Disabled	Default_JES-Server	JES-Server	tlsKeyring
○	Disabled	Default_LBA-Advisor	LBA-Advisor	tlsKeyring
○	Disabled	Default_MSM	MSM	tlsKeyring

Total: 63, Selected: 1

Describe traffic



Modify Connectivity Rule

Default AT-TLS key ring database

* Rule name: Default_FTP-Server □ Enable rule [Restore Defaults]

Use this panel to specify the traffic settings.

* Application name: FTP-Server

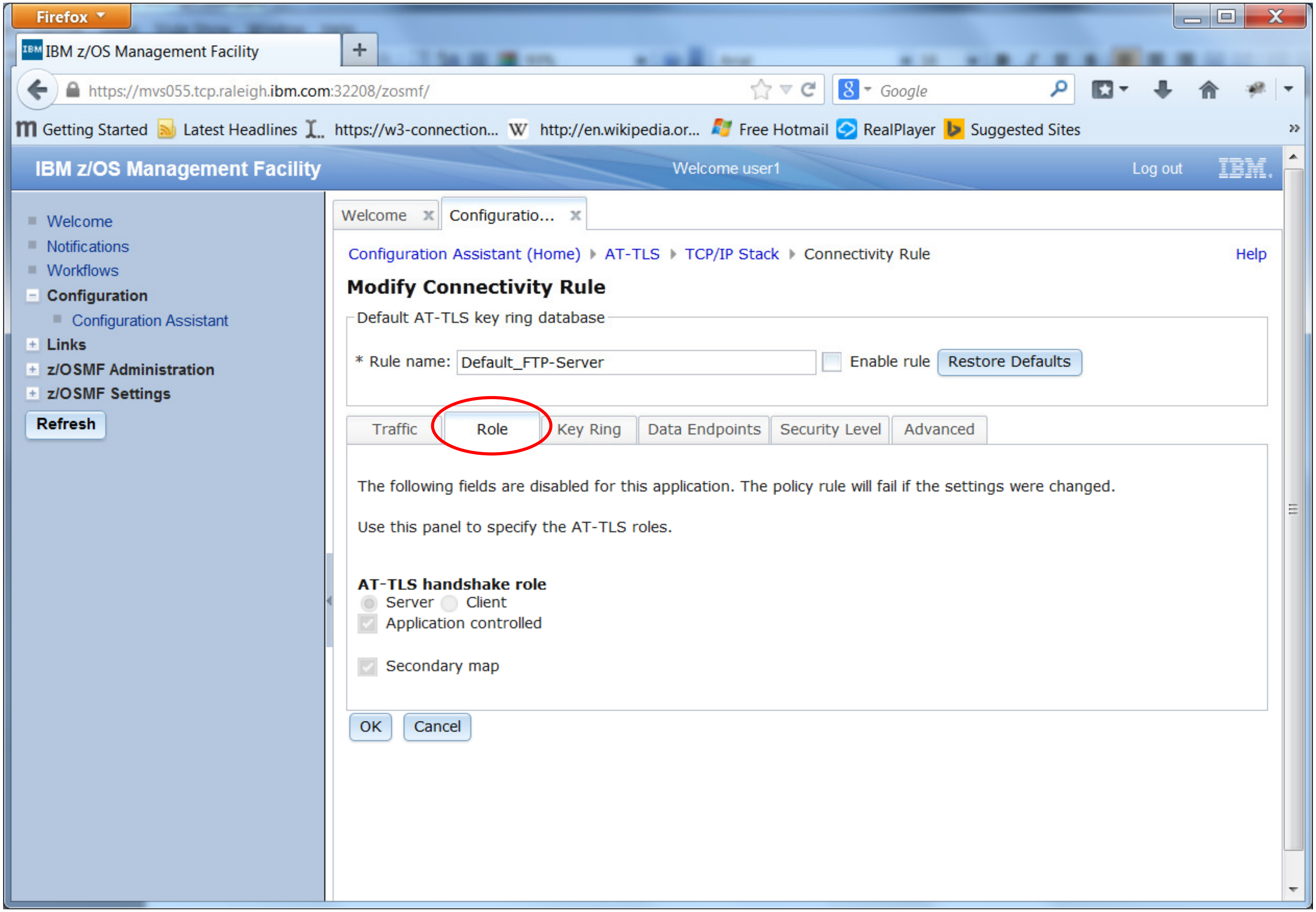
Local Port
○ All ports
○ All ephemeral ports
● Ports:
* 21
Separate multiple ports with a comma

Remote Port
○ All ports
● All ephemeral ports
○ Ports:
*
Separate multiple ports with a comma

Indicate the TCP connect direction
○ Either ● Inbound only ○ Outbound only

Specify jobname and user ID
Jobname: User ID:

Describe role – Not changeable



Modify Connectivity Rule

Default AT-TLS key ring database

* Rule name: Default_FTP-Server □ Enable rule Restore Defaults

The following fields are disabled for this application. The policy rule will fail if the settings were changed.

Use this panel to specify the AT-TLS roles.

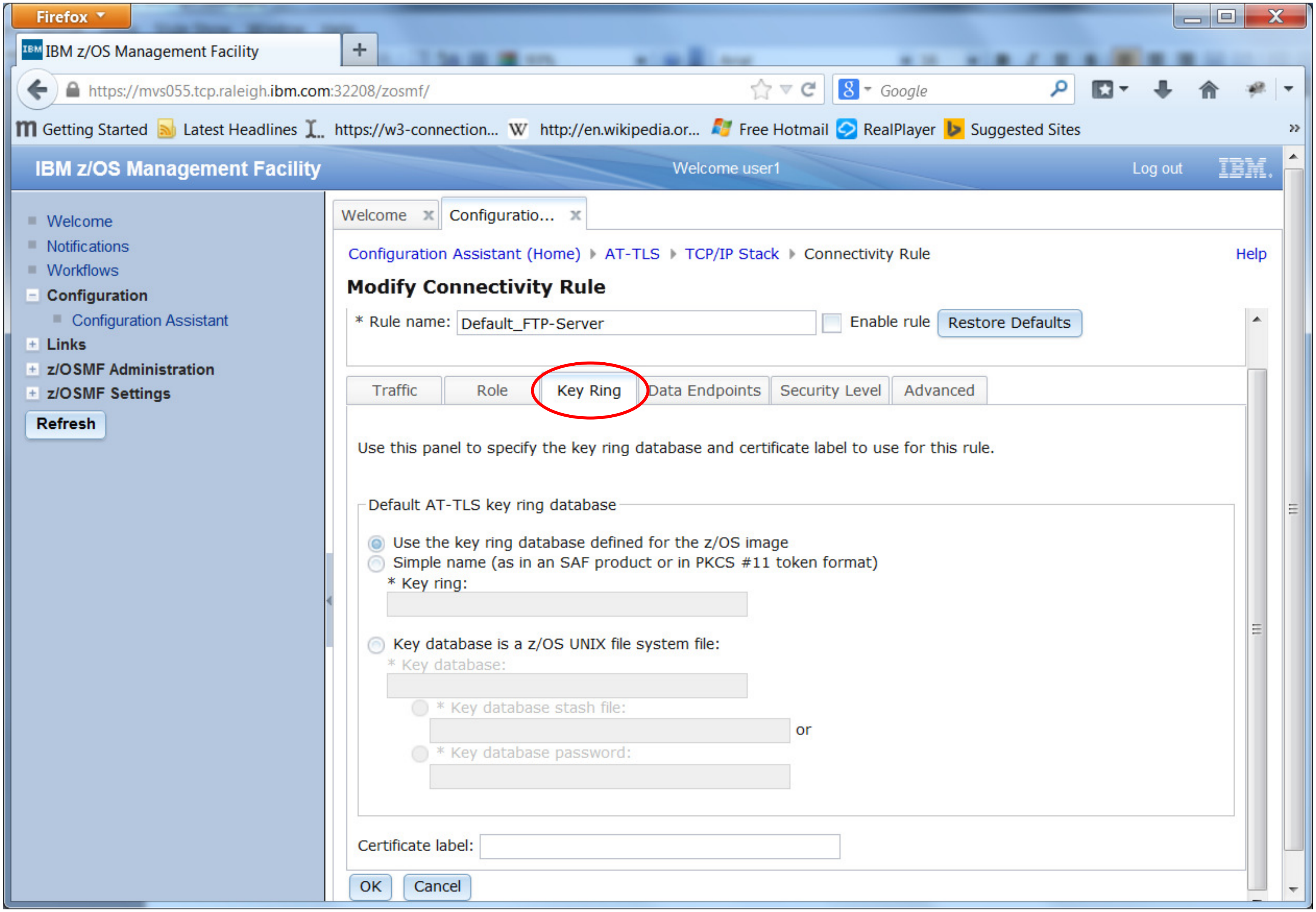
AT-TLS handshake role

● Server ○ Client

☑ Application controlled

☑ Secondary map

Define key ring – in this case use the z/OS image level key ring



Modify Connectivity Rule

* Rule name: Default_FTP-Server

□ Enable rule

Use this panel to specify the key ring database and certificate label to use for this rule.

Default AT-TLS key ring database

● Use the key ring database defined for the z/OS image

○ Simple name (as in an SAF product or in PKCS #11 token format)

* Key ring:

○ Key database is a z/OS UNIX file system file:

* Key database:

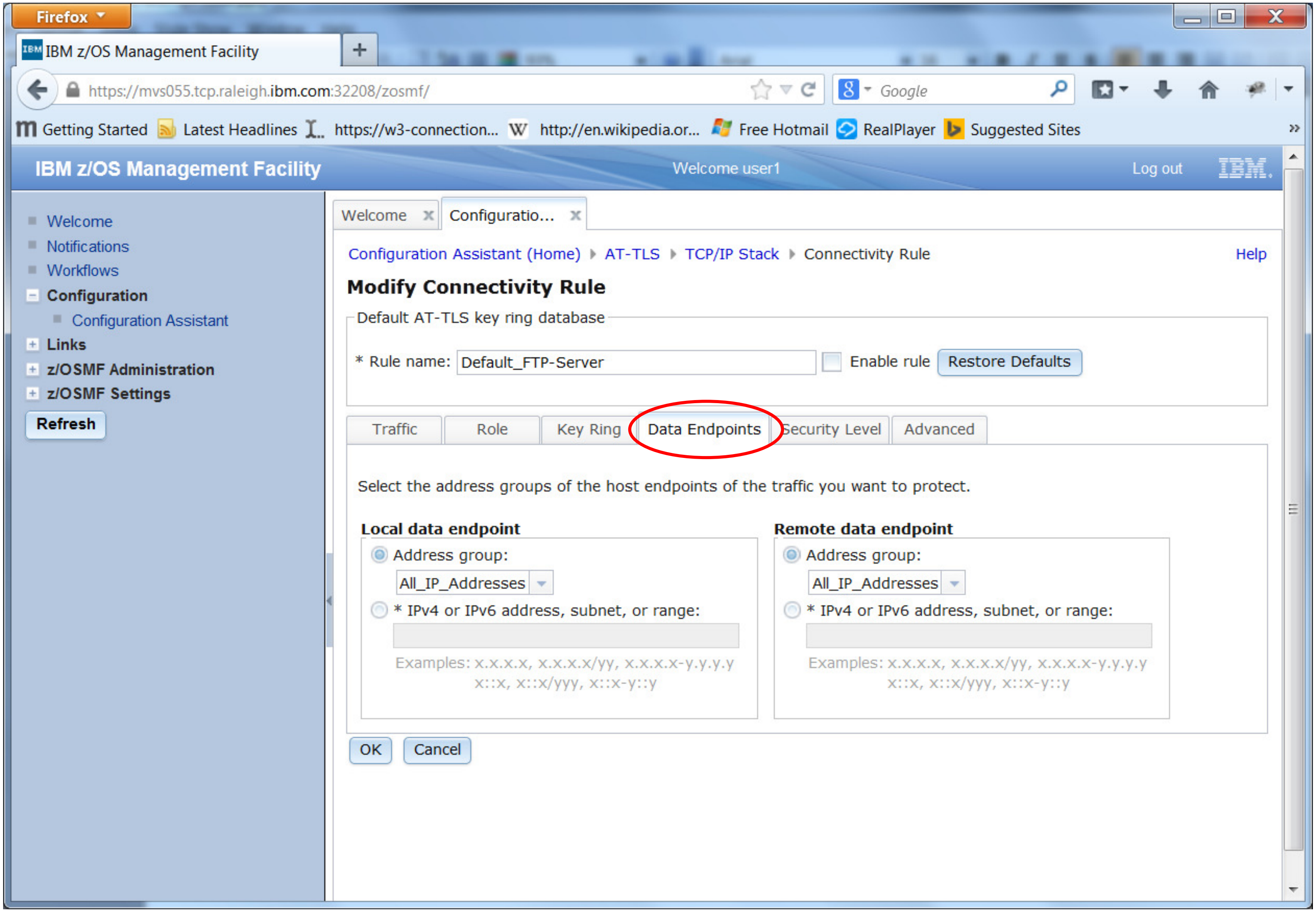
○ * Key database stash file:

or

○ * Key database password:

Certificate label:

Describe data endpoints – in this case apply rule to all endpoints



Configuration Assistant (Home) ▸ AT-TLS ▸ TCP/IP Stack ▸ Connectivity Rule

Modify Connectivity Rule

Default AT-TLS key ring database

* Rule name: Default_FTP-Server

□ Enable rule

Select the address groups of the host endpoints of the traffic you want to protect.

Local data endpoint

● Address group: All_IP_Addresses

○ * IPv4 or IPv6 address, subnet, or range:

Examples: x.x.x.x, x.x.x.x/yy, x.x.x.x-y.y.y.y
x::x, x::x/yyy, x::x-y::y

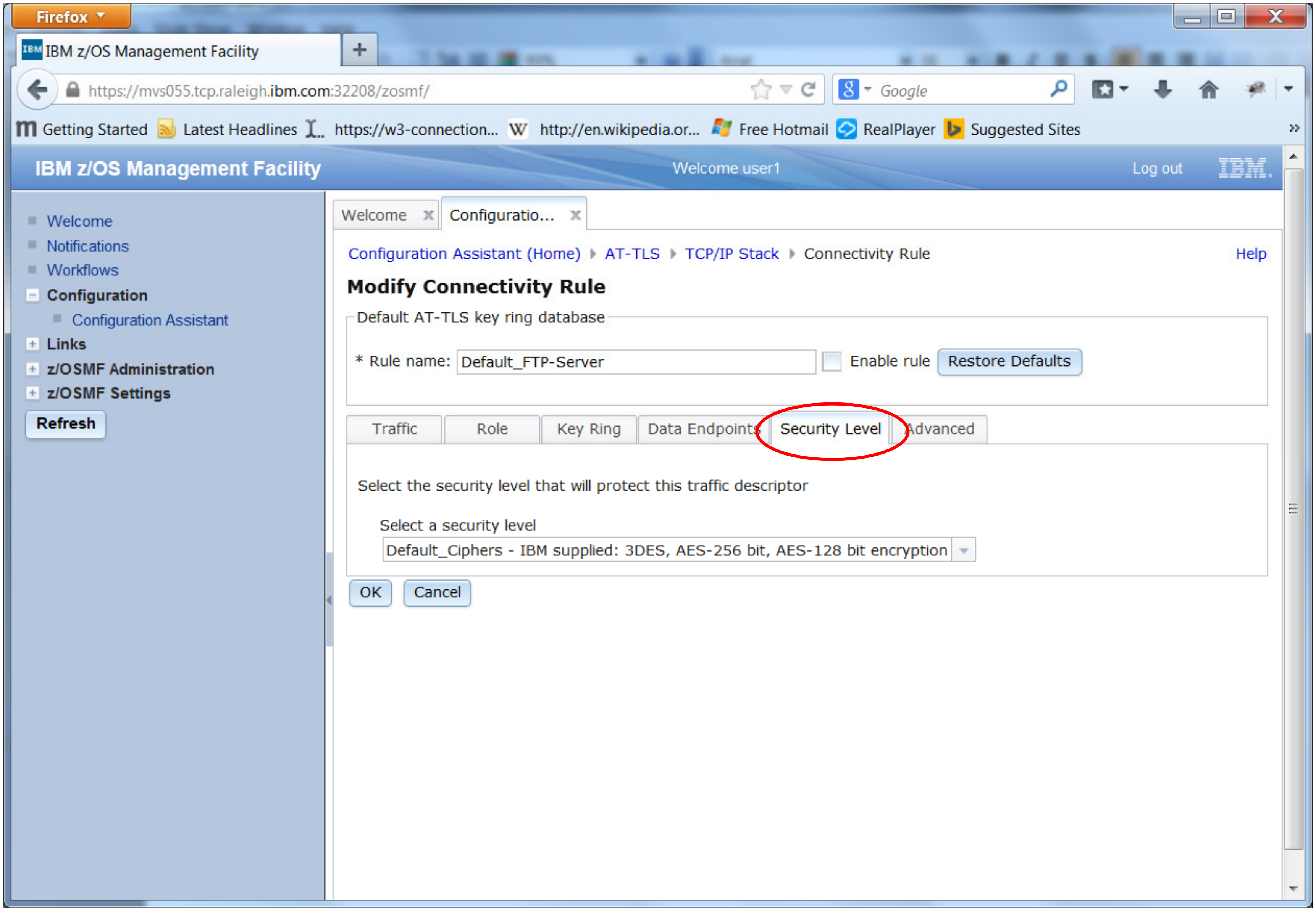
Remote data endpoint

● Address group: All_IP_Addresses

○ * IPv4 or IPv6 address, subnet, or range:

Examples: x.x.x.x, x.x.x.x/yy, x.x.x.x-y.y.y.y
x::x, x::x/yyy, x::x-y::y

Specify details of TLS protection



Configuration Assistant (Home) ▸ AT-TLS ▸ TCP/IP Stack ▸ Connectivity Rule

Modify Connectivity Rule

Default AT-TLS key ring database

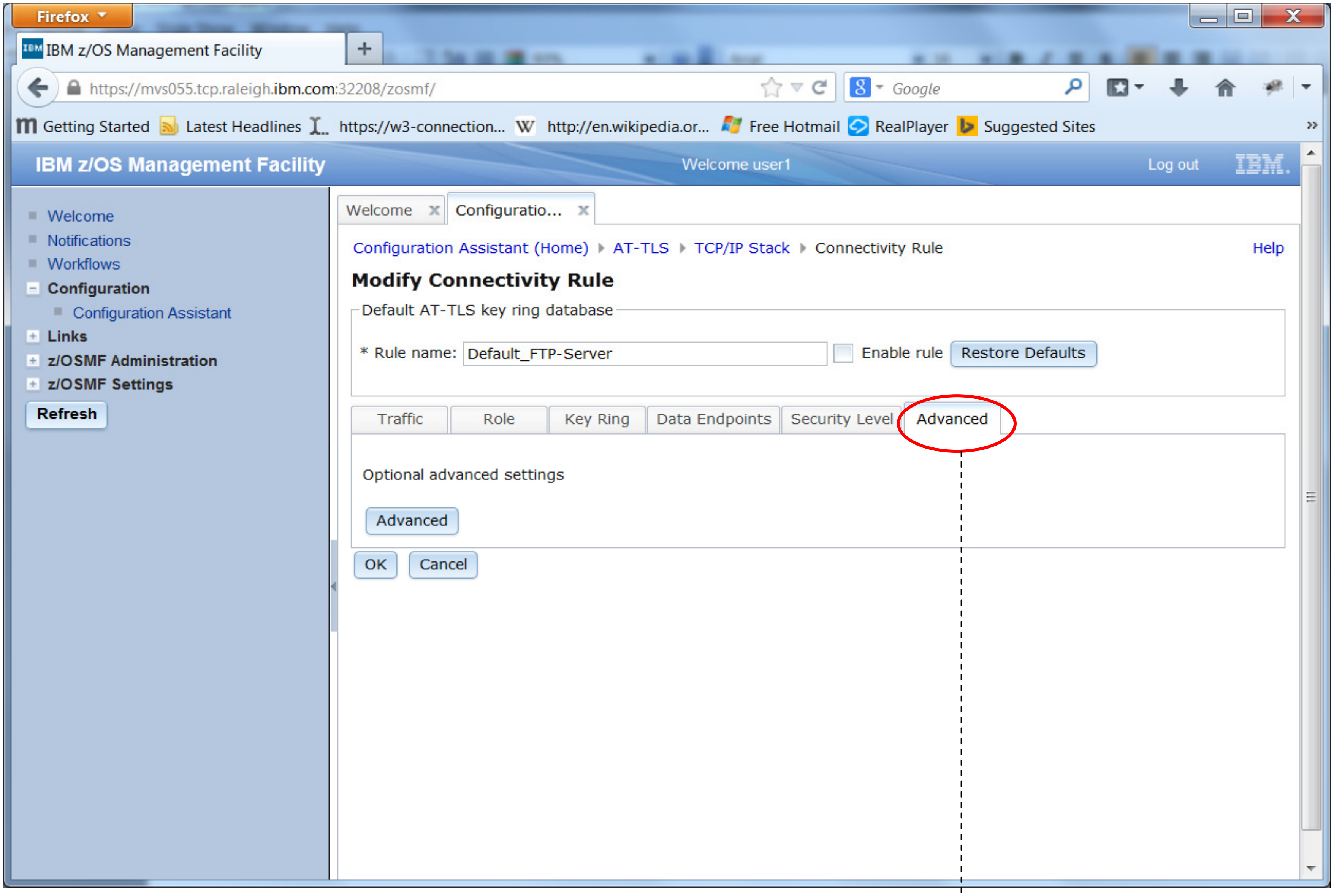
* Rule name: Default_FTP-Server □ Enable rule

Select the security level that will protect this traffic descriptor

Select a security level

Default_Ciphers - IBM supplied: 3DES, AES-256 bit, AES-128 bit encryption

Advanced Settings



Configuration Assistant (Home) ▸ AT-TLS ▸ TCP/IP Stack ▸ Connectivity Rule

Modify Connectivity Rule

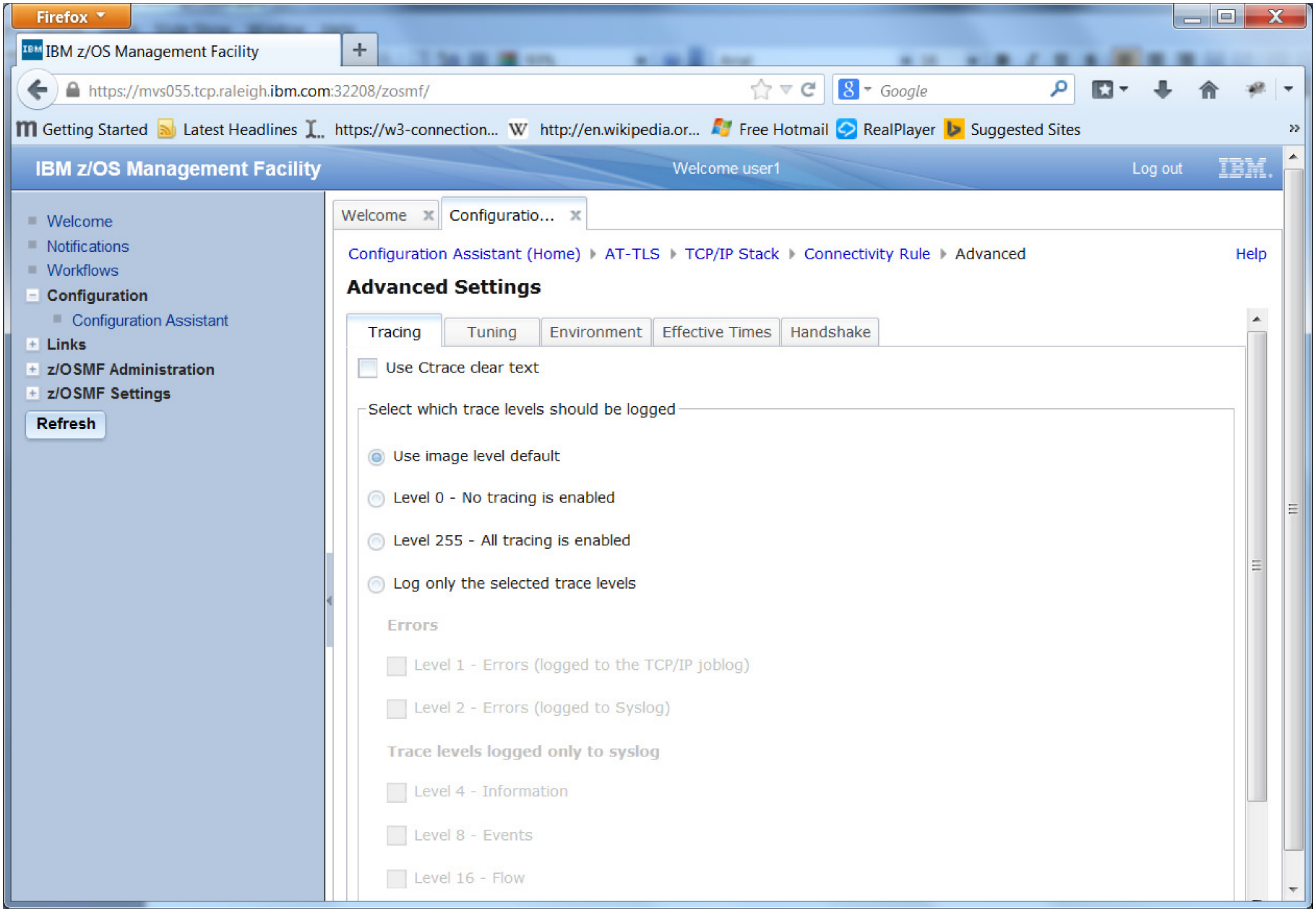
Default AT-TLS key ring database

* Rule name: Default_FTP-Server □ Enable rule

Optional advanced settings

Next page

Advanced settings – categories of available settings



Advanced Settings

Tracing | Tuning | Environment | Effective Times | Handshake

□ Use Ctrace clear text

Select which trace levels should be logged

● Use image level default

○ Level 0 - No tracing is enabled

○ Level 255 - All tracing is enabled

○ Log only the selected trace levels

Errors

□ Level 1 - Errors (logged to the TCP/IP joblog)

□ Level 2 - Errors (logged to Syslog)

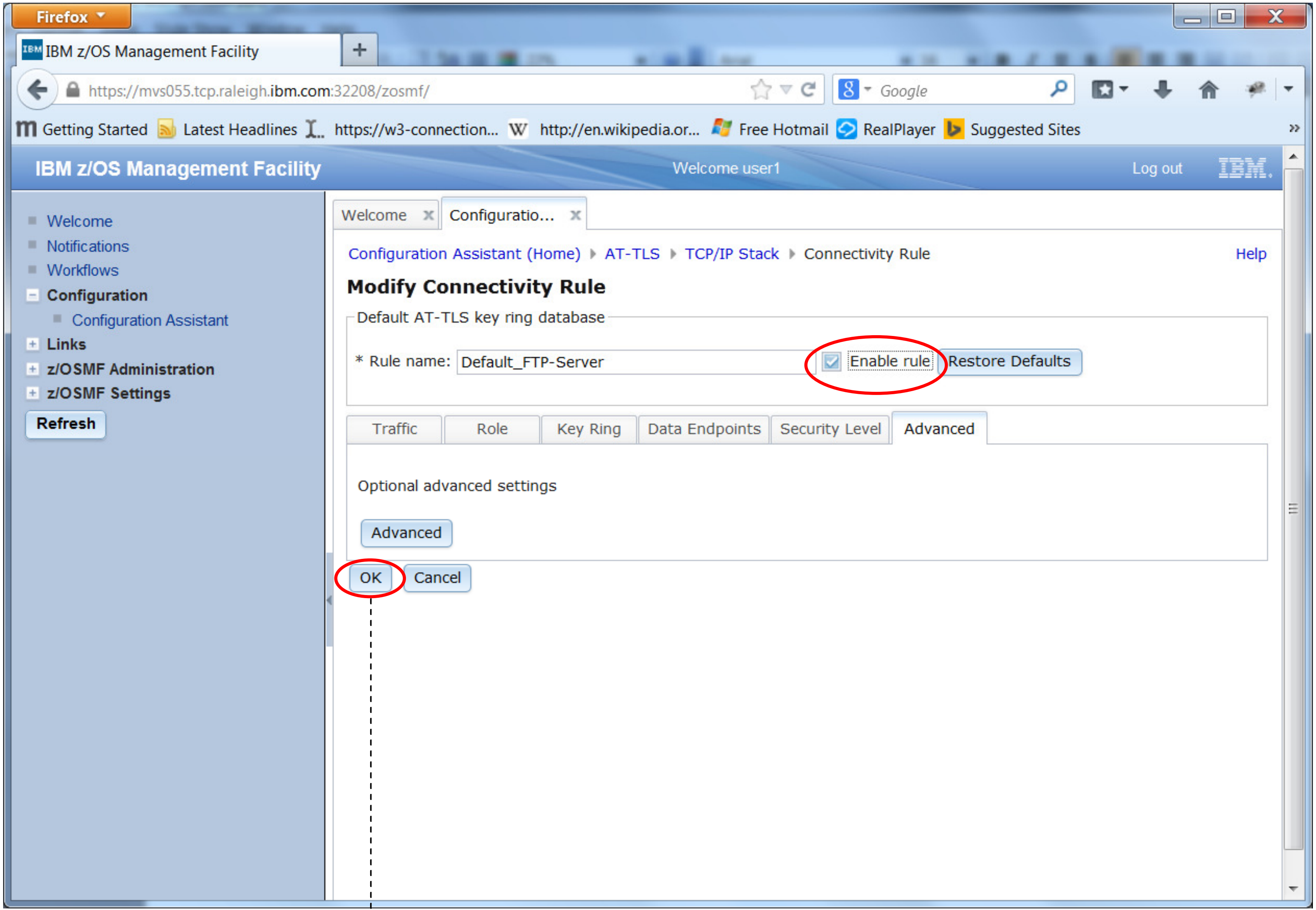
Trace levels logged only to syslog

□ Level 4 - Information

□ Level 8 - Events

□ Level 16 - Flow

Enable rule



Configuration Assistant (Home) ▸ AT-TLS ▸ TCP/IP Stack ▸ Connectivity Rule

Modify Connectivity Rule

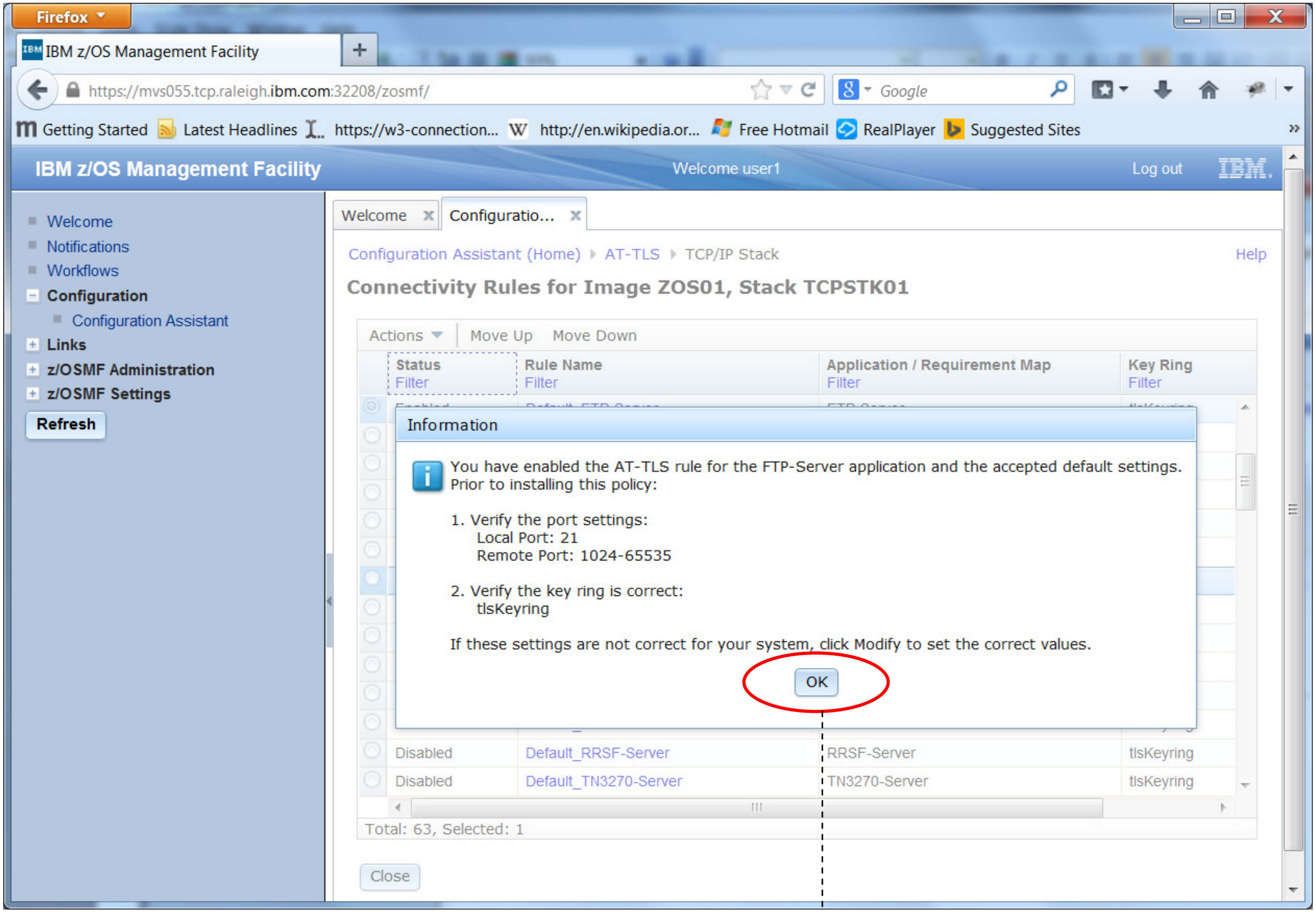
Default AT-TLS key ring database

* Rule name: Default_FTP-Server ☑ Enable rule Restore Defaults

Optional advanced settings

Next page

Are you sure?



Connectivity Rules for Image ZOS01, Stack TCPSTK01

Status	Rule Name	Application / Requirement Map	Key Ring
Filter	Filter	Filter	Filter
Disabled	Default_RRSF-Server	RRSF-Server	tlsKeyring
Disabled	Default_TN3270-Server	TN3270-Server	tlsKeyring

Total: 63, Selected: 1

Information

You have enabled the AT-TLS rule for the FTP-Server application and the accepted default settings. Prior to installing this policy:

1. Verify the port settings:
   Local Port: 21
   Remote Port: 1024-65535

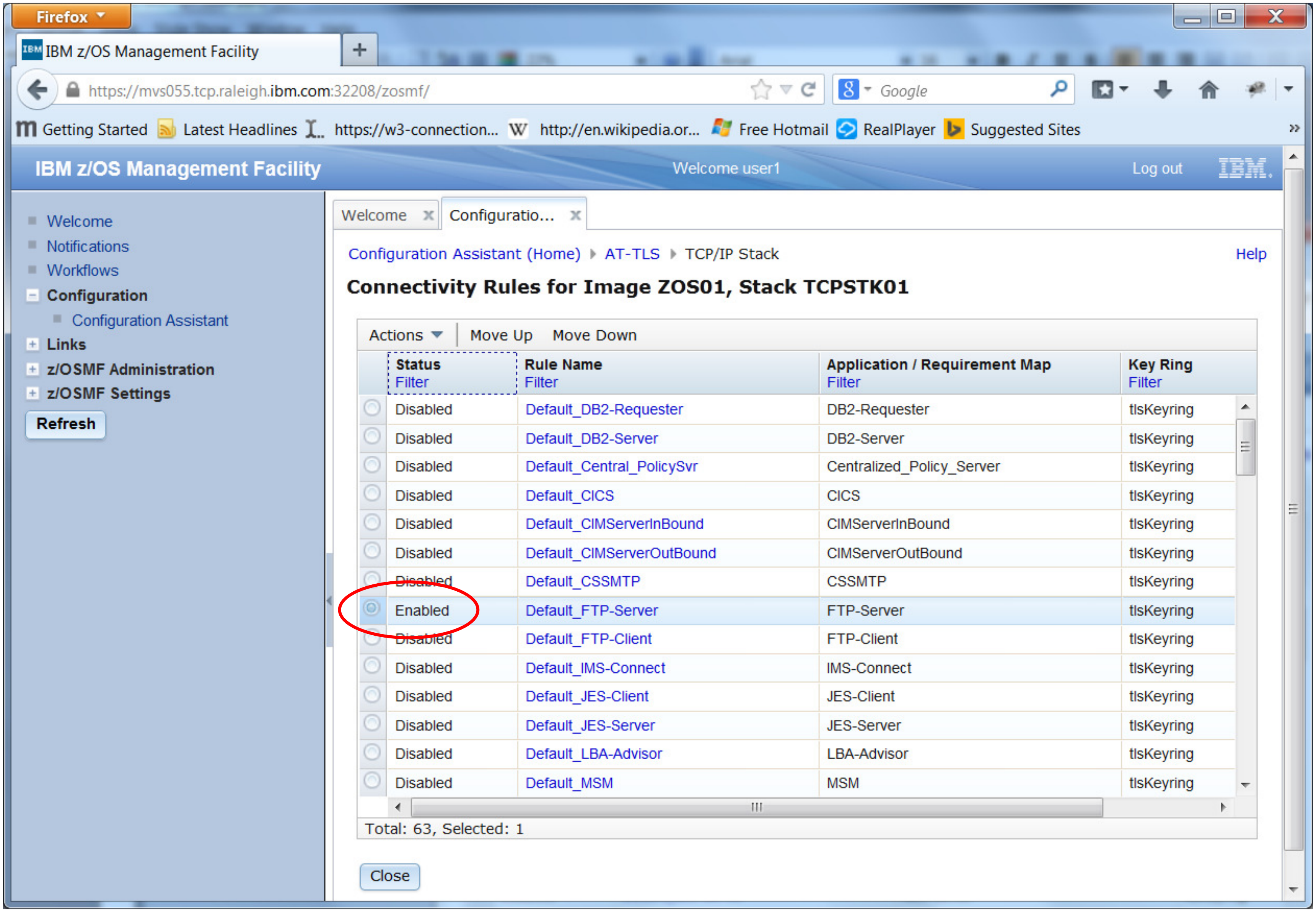
2. Verify the key ring is correct:
   tlsKeyring

If these settings are not correct for your system, click Modify to set the correct values.

OK

Next page

Predefined rule is now enabled

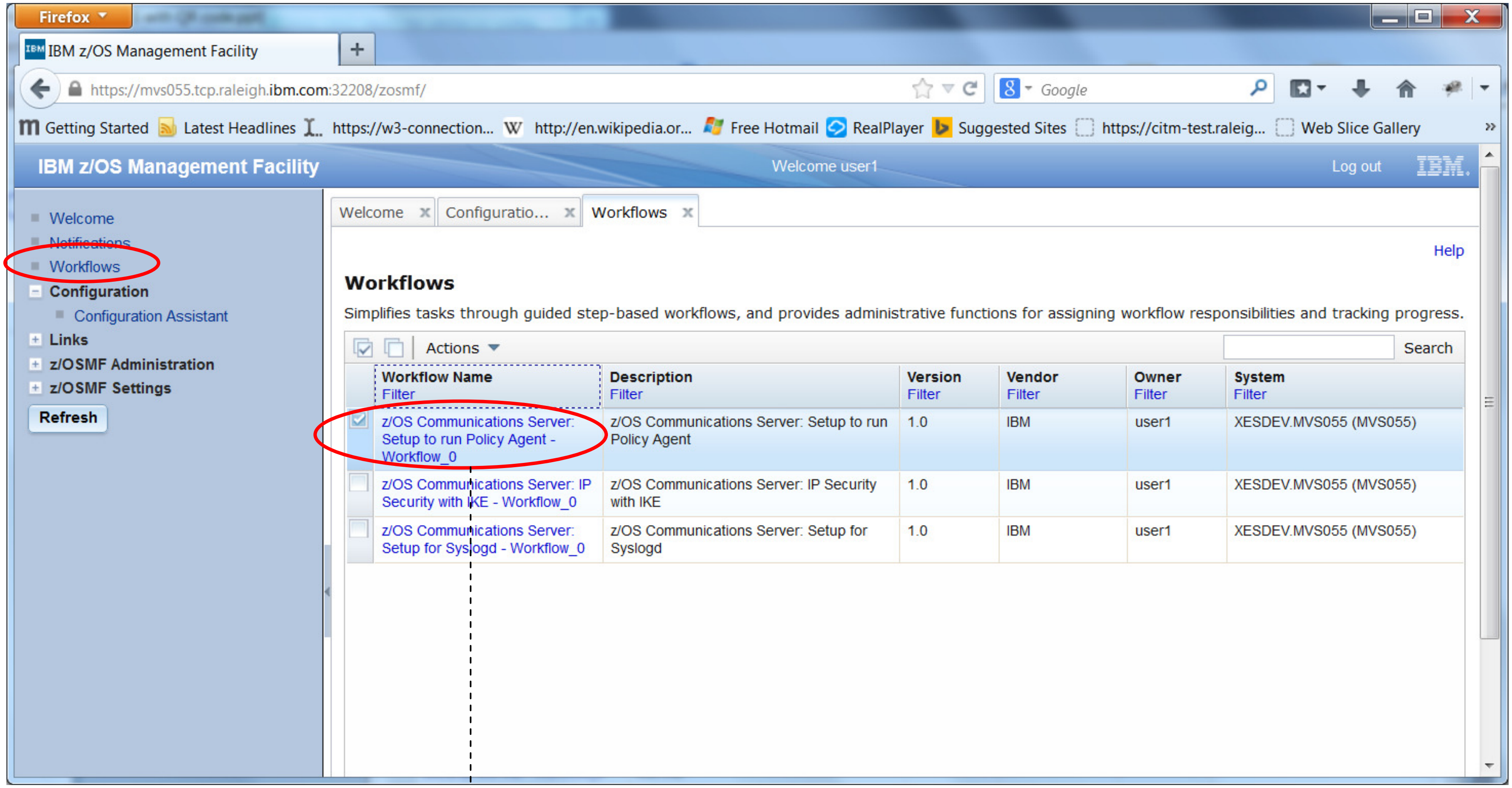


Connectivity Rules for Image ZOS01, Stack TCPSTK01

	Status Filter	Rule Name Filter	Application / Requirement Map Filter	Key Ring Filter
○	Disabled	Default_DB2-Requester	DB2-Requester	tlsKeyring
○	Disabled	Default_DB2-Server	DB2-Server	tlsKeyring
○	Disabled	Default_Central_PolicySvr	Centralized_Policy_Server	tlsKeyring
○	Disabled	Default_CICS	CICS	tlsKeyring
○	Disabled	Default_CIMServerInBound	CIMServerInBound	tlsKeyring
○	Disabled	Default_CIMServerOutBound	CIMServerOutBound	tlsKeyring
○	Disabled	Default_CSSMTP	CSSMTP	tlsKeyring
●	Enabled	Default_FTP-Server	FTP-Server	tlsKeyring
○	Disabled	Default_FTP-Client	FTP-Client	tlsKeyring
○	Disabled	Default_IMS-Connect	IMS-Connect	tlsKeyring
○	Disabled	Default_JES-Client	JES-Client	tlsKeyring
○	Disabled	Default_JES-Server	JES-Server	tlsKeyring
○	Disabled	Default_LBA-Advisor	LBA-Advisor	tlsKeyring
○	Disabled	Default_MSM	MSM	tlsKeyring

Total: 63, Selected: 1

Assistance with the z/OS System preparation tasks – All workflow view …. Found under "Workflows" not Configuration Assistant



Workflows

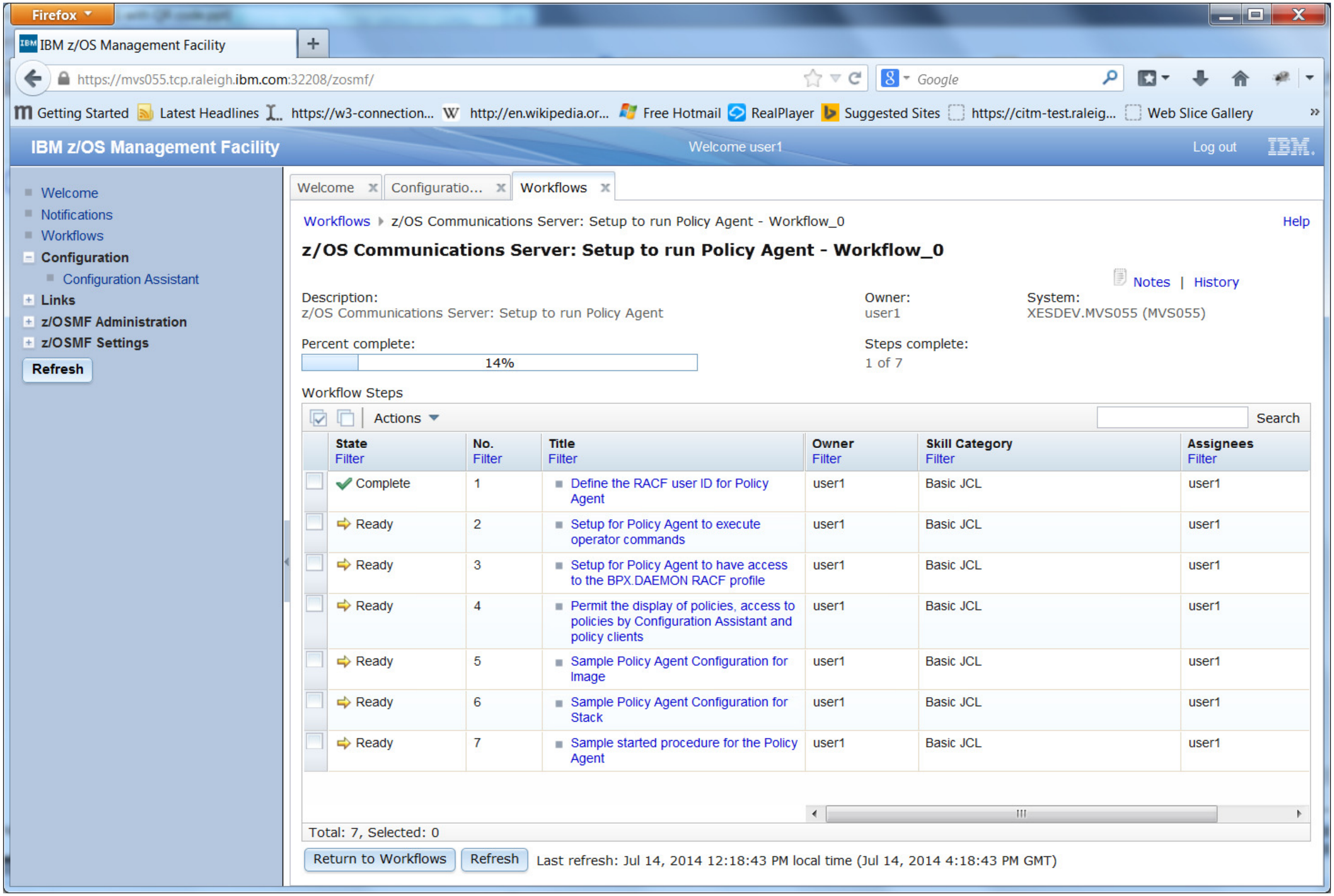
Simplifies tasks through guided step-based workflows, and provides administrative functions for assigning workflow responsibilities and tracking progress.

Actions ▾ Search

	Workflow Name Filter	Description Filter	Version Filter	Vendor Filter	Owner Filter	System Filter
☑	z/OS Communications Server: Setup to run Policy Agent - Workflow_0	z/OS Communications Server: Setup to run Policy Agent	1.0	IBM	user1	XESDEV.MVS055 (MVS055)
□	z/OS Communications Server: IP Security with IKE - Workflow_0	z/OS Communications Server: IP Security with IKE	1.0	IBM	user1	XESDEV.MVS055 (MVS055)
□	z/OS Communications Server: Setup for Syslogd - Workflow_0	z/OS Communications Server: Setup for Syslogd	1.0	IBM	user1	XESDEV.MVS055 (MVS055)

Next page

Assistance with the z/OS System preparation tasks –Specific workflow view



Workflows ▸ z/OS Communications Server: Setup to run Policy Agent - Workflow_0

z/OS Communications Server: Setup to run Policy Agent - Workflow_0

Description:
z/OS Communications Server: Setup to run Policy Agent

Owner:
user1

System:
XESDEV.MVS055 (MVS055)

Percent complete:
14%

Steps complete:
1 of 7

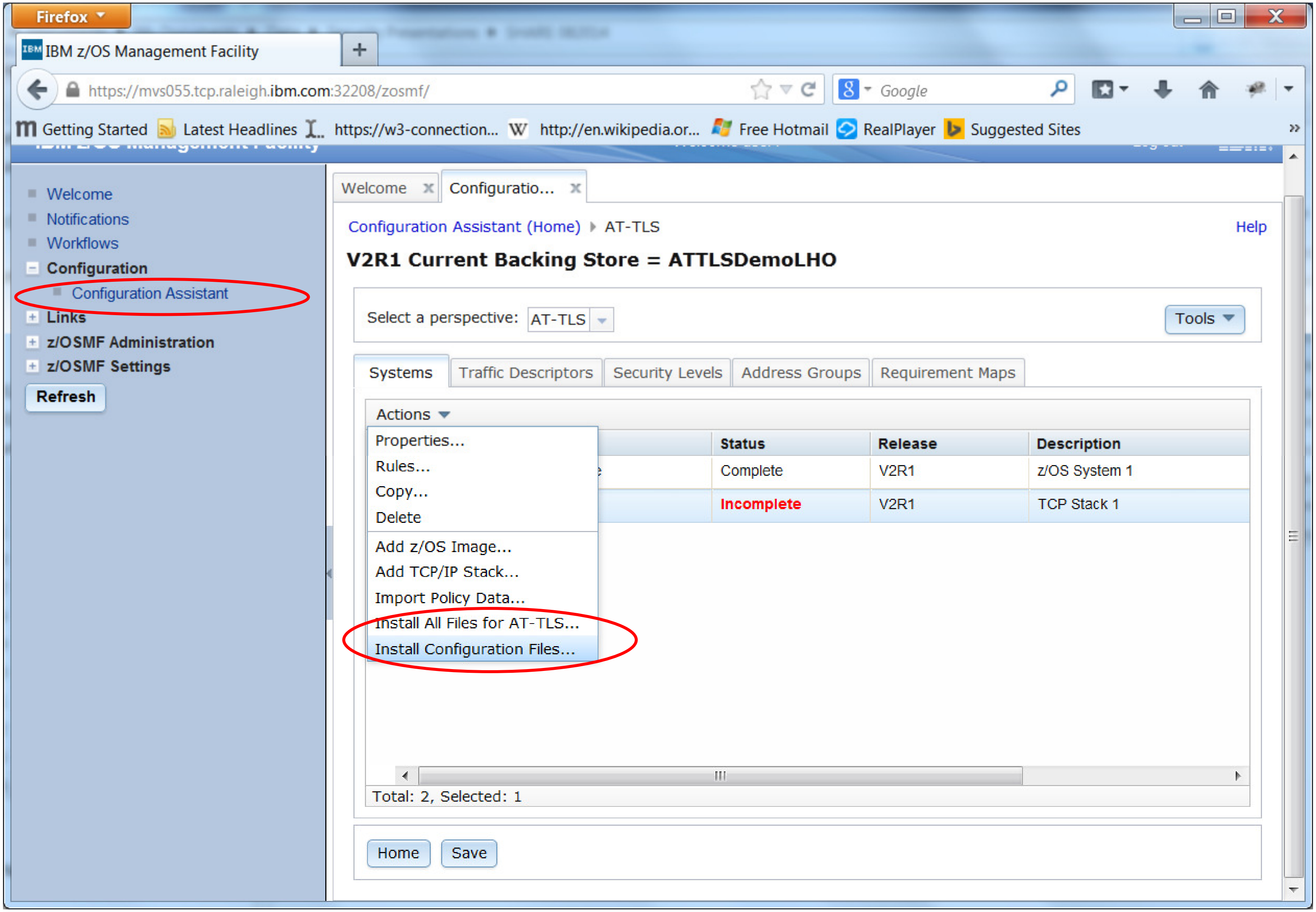
Workflow Steps

	State Filter	No. Filter	Title Filter	Owner Filter	Skill Category Filter	Assignees Filter
□	✓ Complete	1	Define the RACF user ID for Policy Agent	user1	Basic JCL	user1
□	⇨ Ready	2	Setup for Policy Agent to execute operator commands	user1	Basic JCL	user1
□	⇨ Ready	3	Setup for Policy Agent to have access to the BPX.DAEMON RACF profile	user1	Basic JCL	user1
□	⇨ Ready	4	Permit the display of policies, access to policies by Configuration Assistant and policy clients	user1	Basic JCL	user1
□	⇨ Ready	5	Sample Policy Agent Configuration for Image	user1	Basic JCL	user1
□	⇨ Ready	6	Sample Policy Agent Configuration for Stack	user1	Basic JCL	user1
□	⇨ Ready	7	Sample started procedure for the Policy Agent	user1	Basic JCL	user1

Total: 7, Selected: 0

Last refresh: Jul 14, 2014 12:18:43 PM local time (Jul 14, 2014 4:18:43 PM GMT)

How to install configuration and other related files



V2R1 Current Backing Store = ATTLSDemoLHO

Select a perspective: AT-TLS

Status	Release	Description
Complete	V2R1	z/OS System 1
Incomplete	V2R1	TCP Stack 1

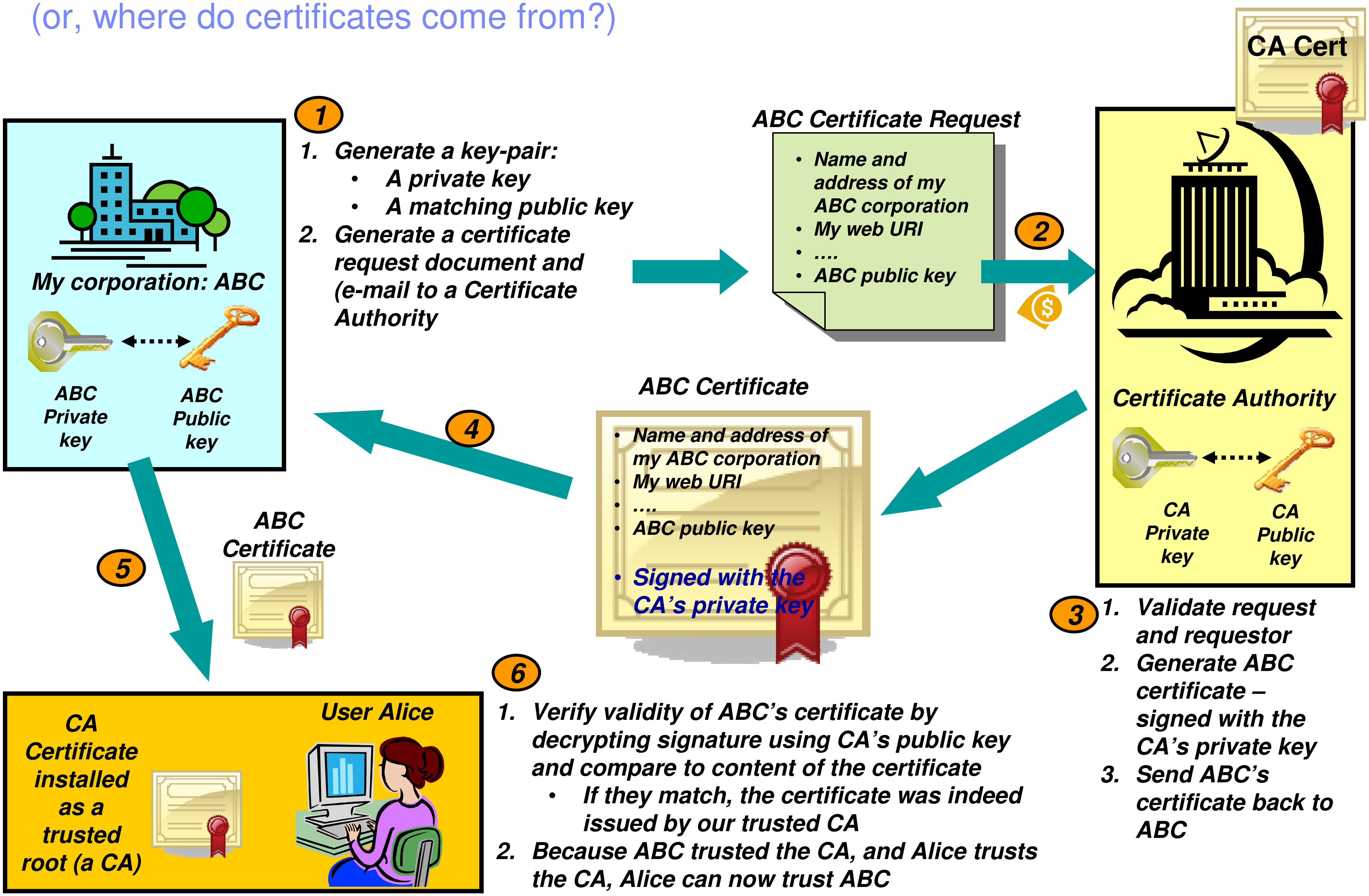
Total: 2, Selected: 1

Please fill out your session evaluation

- z/OS Communications Server Application Transparent TLS
- Session # 16082
- QR Code:

Appendix: Obtain x.509 certificates and update RACF keyrings

Trust relationships and Certificate Authorities
(or, where do certificates come from?)



CA Cert

My corporation: ABC

ABC Private key

ABC Public key

1

1. Generate a key-pair:
   - A private key
   - A matching public key
2. Generate a certificate request document and (e-mail to a Certificate Authority

ABC Certificate Request

- Name and address of my ABC corporation
- My web URI
- ….
- ABC public key

2

Certificate Authority

CA Private key

CA Public key

3

1. Validate request and requestor
2. Generate ABC certificate – signed with the CA's private key
3. Send ABC's certificate back to ABC

ABC Certificate

- Name and address of my ABC corporation
- My web URI
- ….
- ABC public key
- Signed with the CA's private key

4

5

ABC Certificate

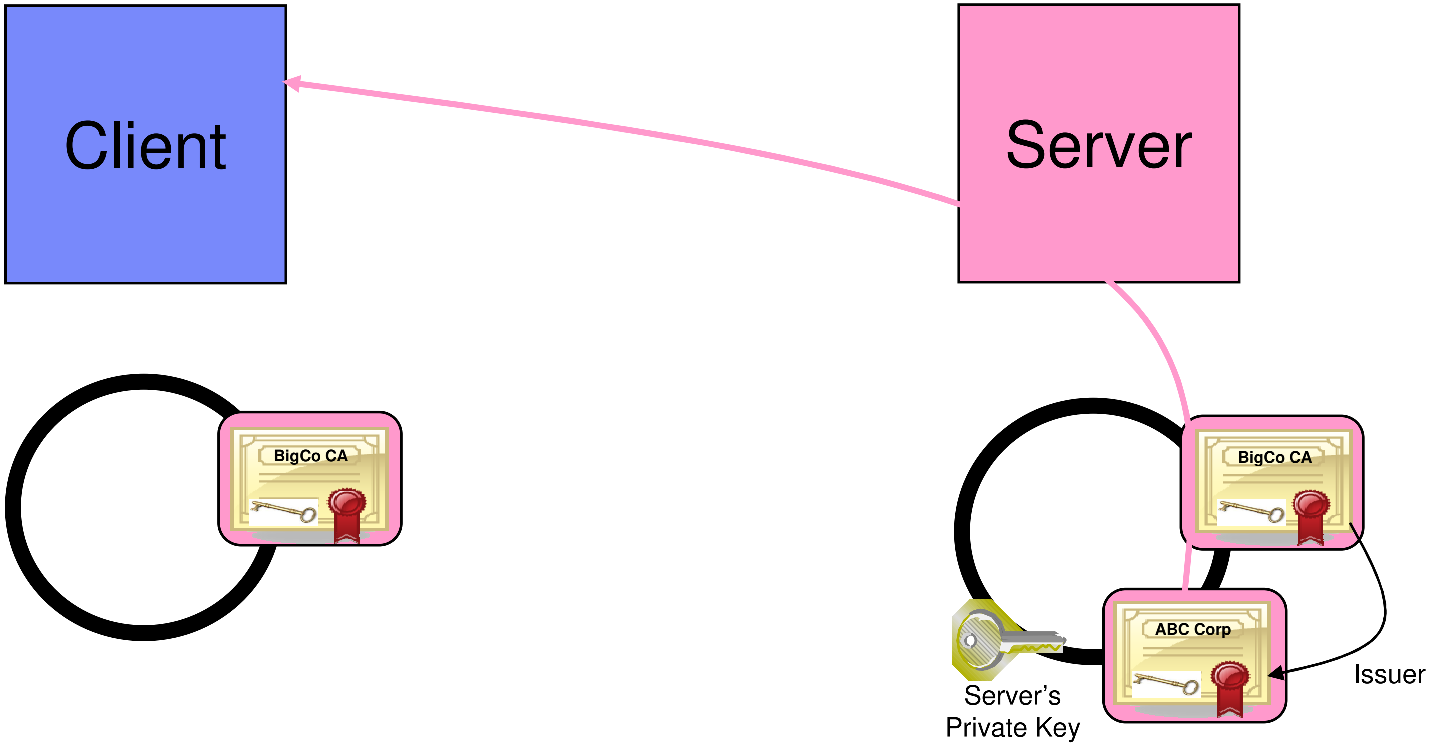
CA Certificate installed as a trusted root (a CA)

User Alice

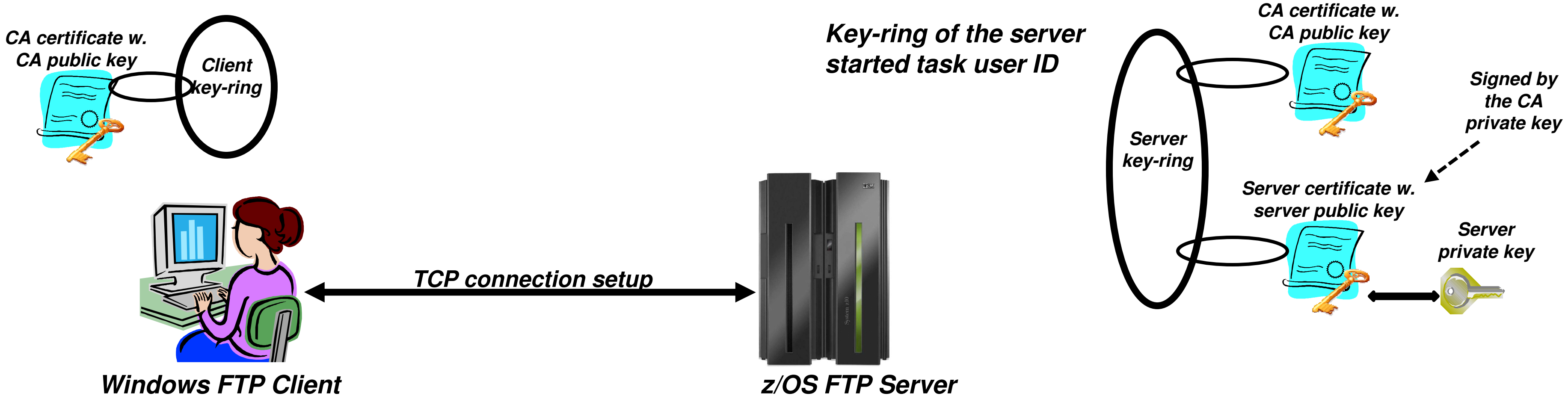
6

1. Verify validity of ABC's certificate by decrypting signature using CA's public key and compare to content of the certificate
   - If they match, the certificate was indeed issued by our trusted CA
2. Because ABC trusted the CA, and Alice trusts the CA, Alice can now trust ABC

Certificates in action: SSL server authentication



What is needed for z/OS Server authentication only (which is sufficient for encrypted data exchange)



CA certificate w. CA public key

Client key-ring

Key-ring of the server started task user ID

Server key-ring

CA certificate w. CA public key

Signed by the CA private key

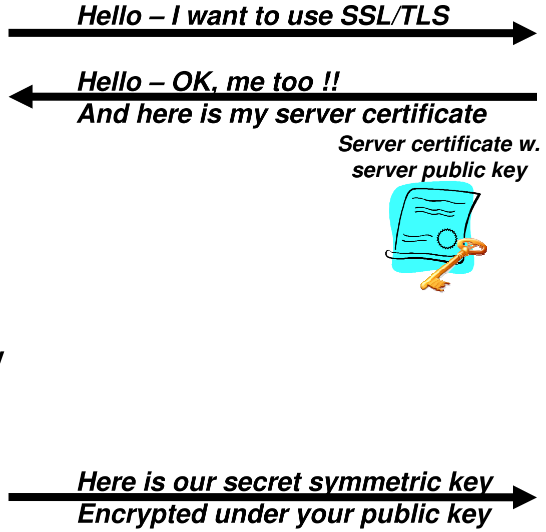
Server certificate w. server public key

Server private key

TCP connection setup

Windows FTP Client

z/OS FTP Server



Hello – I want to use SSL/TLS

Hello – OK, me too !!
And here is my server certificate

Server certificate w. server public key

Here is our secret symmetric key
Encrypted under your public key

1. **Verify server certificate has not expired**
2. **Verify server certificate is valid using CA's public key**
3. **Do optional checks on the server certificate**
4. **Store server's public key for later use**
5. **Generate symmetric key and encrypt under server's public key**

- **CA may be an external CA, such as Verisign, or it may be an in-house CA**
  - In both cases, the CA root certificate needs to be present at both the client and the server side
- **The server certificate is signed by the CA and is stored on the server side**
  - On z/OS, this will typically be the default certificate in the server's started task user ID's key-ring in RACF
- **During SSL handshake, the server certificate (not the server private key) is sent to the client**
  - The client verifies the certificates signature using the CA public key in its copy of the CA certificate

Create self-signed root certificate for test purposes

```
RACDCERT CERTAUTH GENCERT +
         SUBJECTSDN( +
           CN('MVS098 Certificate Authority') +
           OU('Z/OS CS V1R9', 'ENS', 'AIM', 'SWG') +
           O('IBM') +
           L('Raleigh') +
           SP('NC') +
           C('US') ) +
         SIZE(1024) +
         NOTBEFORE(DATE(2010-02-01)) +
         NOTAFTER(DATE(2020-12-31)) +
         WITHLABEL('ABCTLS CA') +
         KEYUSAGE(CERTSIGN) +
         ALTNAME( +
           DOMAIN('mvs098.tcp.raleigh.ibm.com') )
```

Create a self-signed root certificate and a private/public key-pair:

- ***CERTAUTH***
- ***KEYUSAGE(CERTSIGN)***
- ***Absence of a SIGNWITH option***

It can become a nightmare when these things expire, so don’t create certificates with too short a time span! (Your security czar will likely have an opinion on that)

- In a production environment, you would not need a self-signed root certificate. To sign server and personal certificates, you would use your company root certificate or an external Certificate Authority.
- For testing, a self-signed root certificate is useful. It allows you to familiarize yourself with keys and certificates and allows you to thoroughly test your secure FTP setup on z/OS before deploying it in production.

Create server certificate signed with your own root certificate

```
RACDCERT ID(TCPCS) GENCERT +
         SUBJECTSDN( +
           CN('MVS098 Server Certificate') +
           OU('Z/OS CS V1R11', 'ENS', 'AIM', 'SWG') +
           O('IBM') +
           L('Raleigh') +
           SP('NC') +
           C('US') ) +
         SIZE(1024) +
         NOTBEFORE(DATE(2010-02-01)) +
         NOTAFTER(DATE(2020-12-31)) +
         WITHLABEL('ABCTLS TCPSERV') +
         KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN) +
         ALTNAME( +
           DOMAIN('mvs098.tcp.raleigh.ibm.com') ) +
         SIGNWITH(CERTAUTH LABEL('ABCTLS CA'))
```

Create a server certificate signed with your own root certificate and a private/public key pair:

- ***ID(userID) – the started task user ID of your server***
- ***KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)***
- ***SIGNWITH(CERTAUTH LABEL('your root certificate')***

- In a production environment, you would use an alternative procedure after having generated the server key pair and certificate:
  - You would generate a certificate signing request and send it to your CA
  - Your CA would process your request and create a certificate signed with the CA private key
  - You would import the signed certificate into RACF

Alternative: use an external CA to sign your server certificate

```
RACDCERT ID(TCPCS) GENCERT +
         SUBJECTSDN( +
           CN('MVS098 Server Certificate') +
           OU('Z/OS CS V1R11', 'ENS', 'AIM', 'SWG') +
           O('IBM') +
           L('Raleigh') +
           SP('NC') +
           C('US') ) +
         SIZE(1024) +
         NOTBEFORE(DATE(2010-02-01)) +
         NOTAFTER(DATE(2020-12-31)) +
         WITHLABEL('ABCTLS TCPSERV') +
         KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN) +
         ALTNAME( +
           DOMAIN('mvs098.tcp.raleigh.ibm.com') )
RACDCERT ID(TCPCS) GENREQ (LABEL('ABCTLS TCPSERV')) +
         DSN('USER1.PKITEST.SERVERS.REQ')

(**** delay here while CA processes your request ****)

RACDCERT ID(TCPCS) +
         ADD('USER1.PKITEST.SERVERS.CRT') +
         TRUST +
         WITHLABEL('ABCTLS TCPSERV')
```

Create a server certificate and a private/public key pair:

- ***ID(userID) – the started task user ID of your server***
- ***KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)***

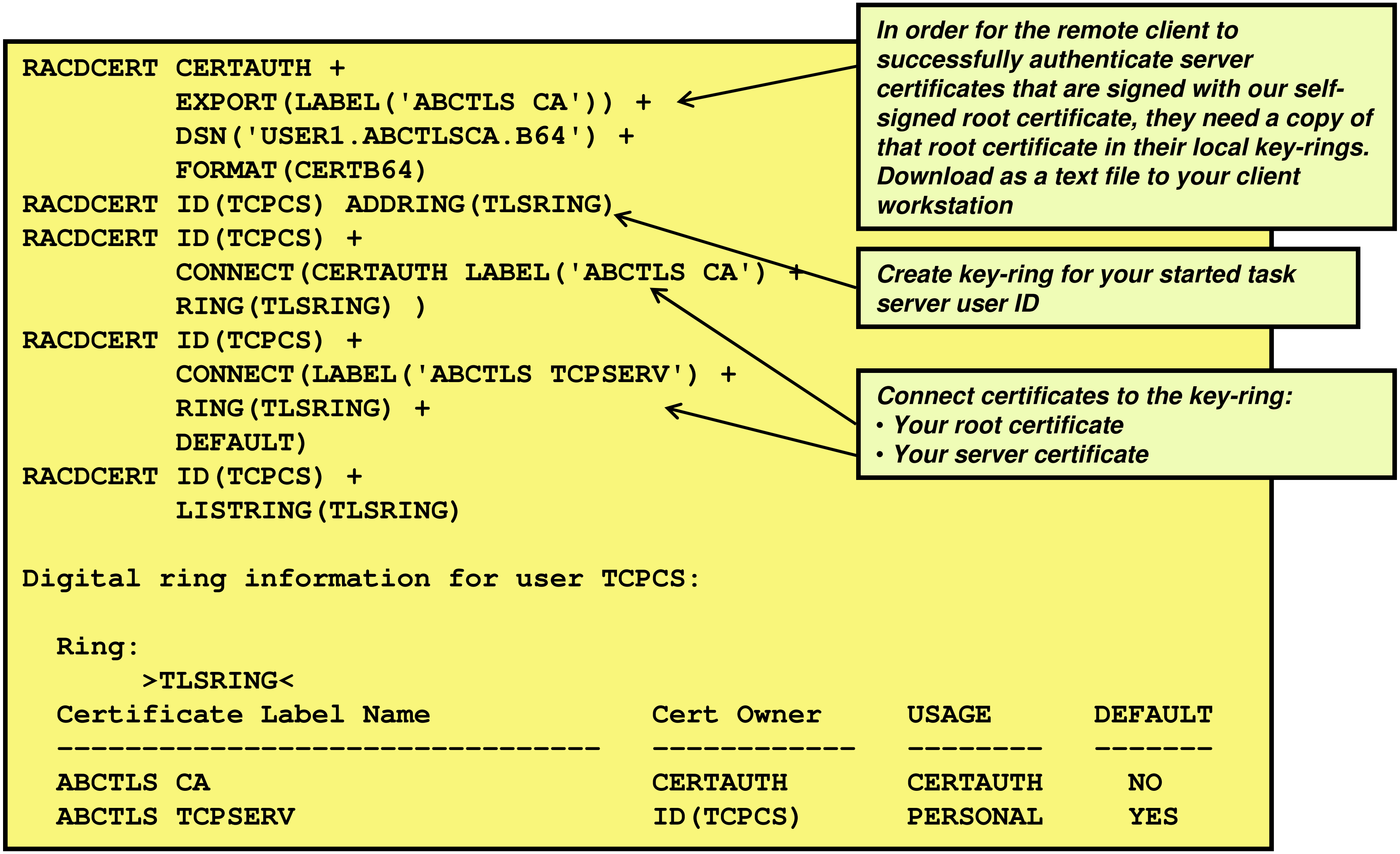
Generate a request to have the certificate signed by an external CA

- ***Send the request to the CA***
- ***Receive the response from the CA***

Add the signed certificate into RACF

If not already there, you also need to add the CA's root certificate to RACF as a CERTAUTH certificate !!

Create your z/OS server started task user ID key-ring and connect required certificates to it



```
RACDCERT CERTAUTH +
         EXPORT(LABEL('ABCTLS CA')) +
         DSN('USER1.ABCTLSCA.B64') +
         FORMAT(CERTB64)
RACDCERT ID(TCPCS) ADDRING(TLSRING)
RACDCERT ID(TCPCS) +
         CONNECT(CERTAUTH LABEL('ABCTLS CA') +
         RING(TLSRING) )
RACDCERT ID(TCPCS) +
         CONNECT(LABEL('ABCTLS TCPSERV') +
         RING(TLSRING) +
         DEFAULT)
RACDCERT ID(TCPCS) +
         LISTRING(TLSRING)

Digital ring information for user TCPCS:

  Ring:
       >TLSRING<
  Certificate Label Name             Cert Owner     USAGE      DEFAULT
  --------------------------------   ------------   --------   -------
  ABCTLS CA                          CERTAUTH       CERTAUTH     NO
  ABCTLS TCPSERV                     ID(TCPCS)      PERSONAL     YES
```

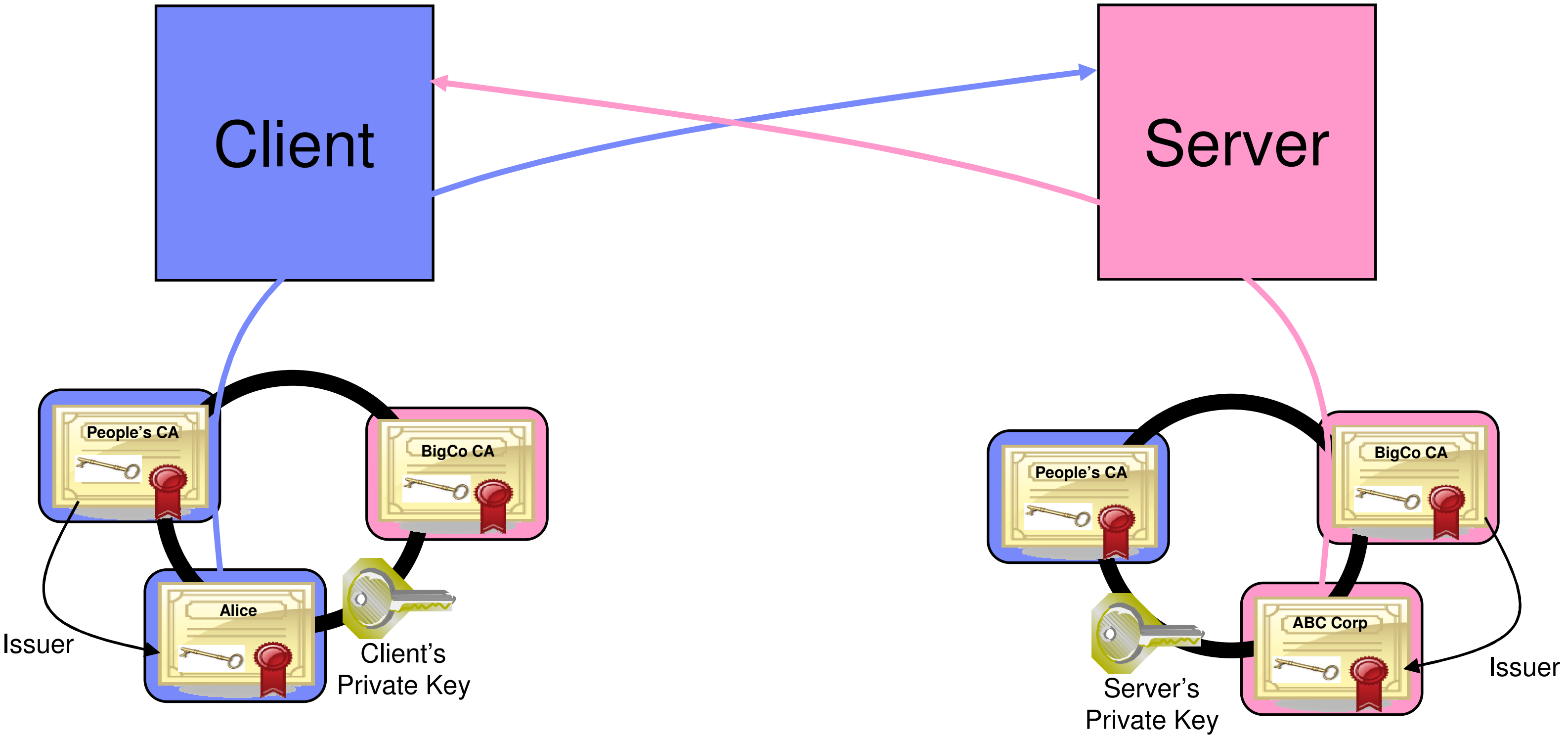
In order for the remote client to successfully authenticate server certificates that are signed with our self-signed root certificate, they need a copy of that root certificate in their local key-rings. Download as a text file to your client workstation

Create key-ring for your started task server user ID

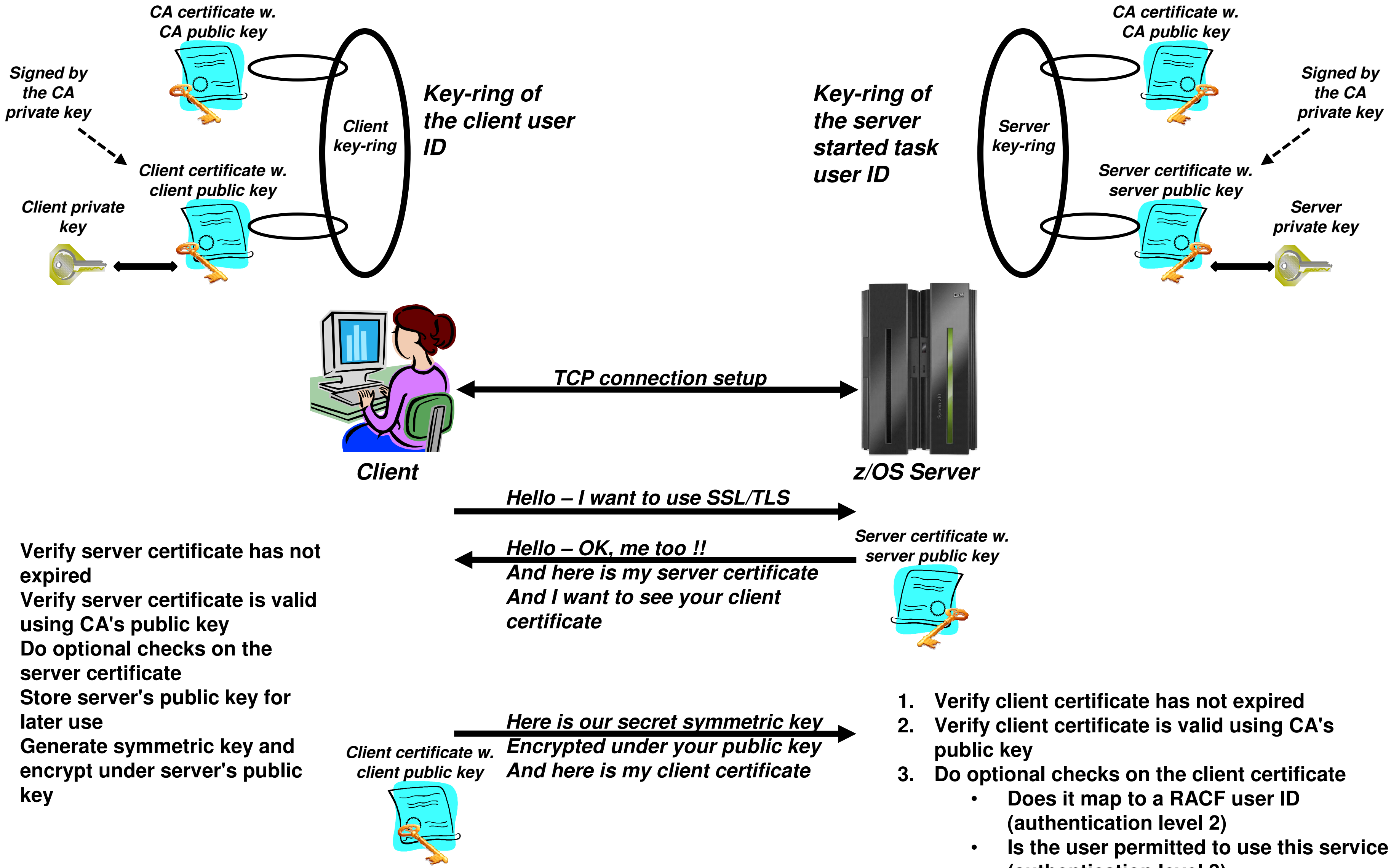
Connect certificates to the key-ring:
- *Your root certificate*
- *Your server certificate*

Certificates in action: SSL client authentication

(implies server authentication as well)



What is needed for z/OS Server and client authentication?



CA certificate w. CA public key

Signed by the CA private key

Client certificate w. client public key

Client private key

Client key-ring

Key-ring of the client user ID

Key-ring of the server started task user ID

Server key-ring

CA certificate w. CA public key

Signed by the CA private key

Server certificate w. server public key

Server private key

TCP connection setup

Client

z/OS Server

Hello – I want to use SSL/TLS

Hello – OK, me too !!
And here is my server certificate
And I want to see your client certificate

Server certificate w. server public key

Here is our secret symmetric key
Encrypted under your public key
And here is my client certificate

Client certificate w. client public key

1. **Verify server certificate has not expired**
2. **Verify server certificate is valid using CA's public key**
3. **Do optional checks on the server certificate**
4. **Store server's public key for later use**
5. **Generate symmetric key and encrypt under server's public key**

1. **Verify client certificate has not expired**
2. **Verify client certificate is valid using CA's public key**
3. **Do optional checks on the client certificate**
   - **Does it map to a RACF user ID (authentication level 2)**
   - **Is the user permitted to use this service (authentication level 3)**

For more information…

URL	Content
http://www.twitter.com/IBM_Commserver	IBM Communications Server Twitter Feed
http://www.facebook.com/IBMCommserver	IBM Communications Server Facebook Fan Page
http://www.ibm.com/systems/z/	IBM System z in general
http://www.ibm.com/systems/z/hardware/networking/	IBM Mainframe System z networking
http://www.ibm.com/software/network/commserver/	IBM Software Communications Server products
http://www.ibm.com/software/network/commserver/zos/	IBM z/OS Communications Server
http://www.ibm.com/software/network/commserver/z_lin/	IBM Communications Server for Linux on System z
http://www.ibm.com/software/network/ccl/	IBM Communication Controller for Linux on System z
http://www.ibm.com/software/network/commserver/library/	IBM Communications Server library
http://www.redbooks.ibm.com	ITSO Redbooks
http://www.ibm.com/software/network/commserver/zos/support/	IBM z/OS Communications Server technical Support – including TechNotes from service
http://www.ibm.com/support/techdocs/atsmastr.nsf/Web/TechDocs	Technical support documentation from Washington Systems Center (techdocs, flashes, presentations, white papers, etc.)
http://www.rfc-editor.org/rfcsearch.html	Request For Comments (RFC)
http://www.ibm.com/systems/z/os/zos/bkserv/	IBM z/OS Internet library – PDF files of all z/OS manuals including Communications Server