z/OS 3.2 IBM Education Assistant

Solution Name: SMTP AUTH support for CSSMTP

Solution Element(s): z/OS Communications Server

July 2025

Agenda

- Trademarks
- Objectives
- Overview
- Usage & Invocation
- Interactions & Dependencies
- Upgrade & Coexistence Considerations
- Installation & Configuration
- Summary
- Appendix

Trademarks

- See url http://www.ibm.com/legal/copytrade.shtml for a list of trademarks.
- Additional Trademarks: None

Objectives

Provide a high-level overview of the following Communications Server function in z/OS 3.2:

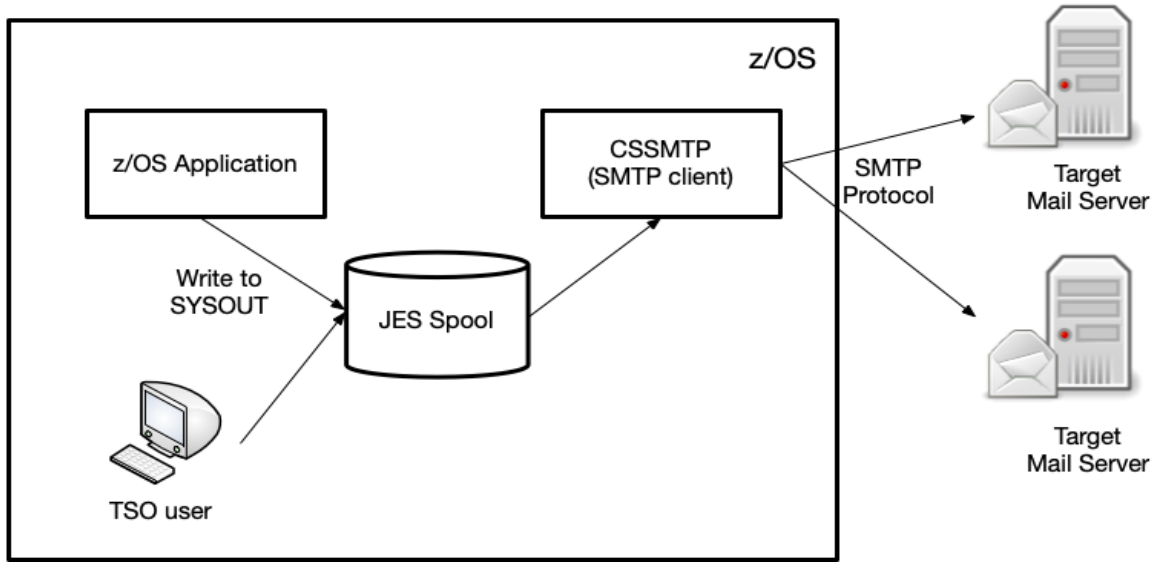
- SMTP AUTH support for CSSMTP

Overview

- Who (Audience)
  - Users who need to enable the server authentication (AUTH) of the originator of the email in a manner consistent with RFC 4954 due to regulatory compliance and mail server compatibility needs.
- What (Solution)
  - CSSMTP is updated to authenticate as a single entity or at the email level.
- Wow (Benefit / Value, Need Addressed)
  - Customers can now enable authentication in the mail servers to perform Auth Plain or Auth Login for each target server for CSSMTP as a single entity or email level.

Overview Background information – Communication server SMTP (CSSMTP)

- CSSMTP (Communications Server Simple Mail Transfer Protocol) is a mail client that was first shipped in z/OS V1R11.
- Mail server support is no longer available as a Comm Server function.
- The CSSMTP client acts as a gateway to remove mail messages from JES spool data sets and uses the SMTP protocol to forward mail messages to configured target servers for next hop or final delivery.



Overview - Problem statement

- SMTP AUTH has been a standard since 2007 (RFC 4954).
- It provides a way for mail servers (like Exchange or Postfix) to authenticate the originator of an email.
- CSSMTP supports the use of TLS to encrypt the mail as it traverses the network, but it does not provide email origin authentication.
- Customers need SMTP AUTH support for two primary reasons:
  1) Regulatory compliance – (EU regulations require email origin authentication)
  2) Compatibility with Microsoft Exchange servers. (Many customers have had to reconfigure their Microsoft Exchange servers to disable SMTP AUTH for the connections from z/OS CSSMTP – something that typically counters their internal email security policies).
- RFC-4954 specifies that the mail client will identify a SASL (Simple Authentication and Security Layer) mechanism to use for the authentication.
- There are several SASL mechanisms, but the one that is required by the RFC is PLAIN.
  - PLAIN is a simple username/password authentication mechanism that requires TLS for the connection to the mail server to protect the username and password.
  - Another SASL mechanism is LOGIN (which is equivalent to Microsoft Exchange “BasicAuth” support).

Overview - Solution

- CSSMTP is enhanced to support AUTH PLAIN and AUTH LOGIN methods.

- The AUTH PLAIN/LOGIN command initiates a SASL authentication exchange between the client and the server.

- Client and server MUST establish a TLS connection before sending the AUTH credentials.
  - For AUTH PLAIN, after a successful TLS connection CSSMTP will perform an additional hostname check to verify if the server hostname matches the certificate hostname (SAN/CN).
  - If the TLS connection with the target server has not been established or if the hostname check fails, CSSMTP will not attempt AUTH PLAIN and the connection to the server will be terminated.
  - If the TLS connection with the target server has not been established, CSSMTP will not attempt AUTH LOGIN and the connection to the server will be terminated.
- If the server supports both methods (PLAIN and LOGIN), CSSMTP will always choose PLAIN. CSSMTP will attempt the AUTH connection only after the server requests AUTH PLAIN or LOGIN.

Overview – Solution cont..

- The system administrator should configure the authentication username and password in RACF for secure storage. This is referred to as the AuthEntity.
- The CSSMTP configuration file will include a new parameter for the name of the AuthEntity.
  - (In the examples below: “target.mail.server” for single entity or <MailFrom> for email level authentication)
- If the CSSMTP configuration file includes an AuthEntity for a target server, CSSMTP will know that Auth should be attempted for that target server(once the server requests it).
- After a successful TLS connection and a hostname check, CSSMTP will retrieve the username and password from RACF and send it to the target server with the AUTH PLAIN or AUTH LOGIN command.
- If the authentication is successful, CSSMTP can start forwarding the mails to that server.
  - Otherwise, the connection will be terminated.
- If any target server enables AUTH at the email level and the UNDELIVERABLE statement is set to ReturnToMailFrom Yes, ReportMailFrom will be used to authenticate undeliverable mail. If ReportMailFrom is empty, the undeliverable mail will go to the deadletter directory.

Overview – Solution Auth flow

AUTH PLAIN Flow	AUTH LOGIN Flow
Server: 220-smtp.example.com ESMTP Server Client: EHLO client.example.com Server: 250-smtp.example.com Hello client.example.com Server: 250-AUTH GSSAPI DIGEST-MD5 PLAIN Server: 250-ENHANCEDSTATUSCODES Server: 250 STARTTLS Client: STARTTLS Server: 220 Ready to start TLS ... TLS negotiation proceeds, further commands protected by TLS layer ... Client: EHLO client.example.com Server: 250-smtp.example.com Hello client.example.com Server: 250 AUTH GSSAPI DIGEST-MD5 PLAIN Client: AUTH PLAIN Server: 334 Client: dGVzdAB0ZXN0ADEyMzQ= Server: 235 2.7.0 Authentication successful	Server: 220-smtp.example.com ESMTP Server Client: EHLO client.example.com Server: 250-smtp.example.com Hello client.example.com Server: 250-AUTH LOGIN PLAIN Server: 250-ENHANCEDSTATUSCODES Server: 250 STARTTLS Client: STARTTLS Server: 220 Ready to start TLS ... TLS negotiation proceeds, further commands protected by TLS layer ... Client: EHLO client.example.com Server: 250-smtp.example.com Hello client.example.com Server: 250 AUTH LOGIN Client: AUTH LOGIN Server: 334 DYT3jf4sdDR5 (Server Prompts for username) Client: TXktVXNlcm5hbWU= Server: 334 LIRdf2pekwW3 (Server Prompts for password) Client: TXktUGFzc3dvcmQ= Server: 235 Authentication successful

Username and password base 64 encoded

Username base 64 encoded

Password base 64 encoded

Usage & Invocation - RACF Sample updates

- EZARACF.sample was updated with the instructions on how to store the username and the password for the AuthEntity in RACF

```
//** The AUTHENTITY parameter on the TargetServer statement allows CSSMTP to retrieve the
//** username and the password from RACF to use for authentication with a mail server.

//**   Step 1
//** Activate the required classes:
//* SETROPTS CLASSACT(LDAPBIND)
//* SETROPTS CLASSACT(KEYSMSTR)
//* SETROPTS RACLIST(KEYSMSTR)

//** Step 2
//** For DES, define this profile, using your own secret consisting of 16 hexadecimal chars
//** Note : If you already have LDAP.BINDPW.KEY profile, please skip this step and go to step 3
//* RDEFINE KEYSMSTR LDAP.BINDPW.KEY SSIGNON(KEYENCRYPTED(0023528875DECFAC))
//* SETROPTS RACLIST(KEYSMSTR) REFRESH

//* Step 3
//** Value defined for AuthEntity is defined as a profile in the LDAPBIND class. The
//** username and password are defined in the PROXY segment of the profile in BINDDN
//** and BINDPW fields.
//* RDEFINE LDAPBIND target.mail.server PROXY(BINDDN('username') BINDPW('password'))
```

Continued on next chart …

Usage & Invocation - RACF Sample updates cont.

```
//** For AutEntity as <MailFrom>
//** If the AuthEntity is defined as <MAILFROM>, the email addresses
//** used in the Mail From fields should be defined as profiles
//** in the LDAPBIND class (one profile per email address).
//*
//** RDEFINE LDAPBIND test1@test.com PROXY(BINDDN('username') BINDPW('password'))
//** RDEFINE LDAPBIND test2@test.com PROXY(BINDDN('username') BINDPW('password'))
//*
//** or if the username sent on auth flow to match the email address
//** RDEFINE LDAPBIND test1@test.com PROXY(BINDDN('test1@test.com') BINDPW('password'))
//** RDEFINE LDAPBIND test2@test.com PROXY(BINDDN('test2@test.com') BINDPW('password'))
//*
//* Step 4
//** Give CSSMTP the authority to access the LDAPBIND profiles
//* RDEFINE FACILITY BPX.SERVER UACC(NONE)
//* PERMIT BPX.SERVER CLASS(FACILITY) ID(CSSMTP) ACCESS(READ)
//* SETROPTS RACLIST(FACILITY) REFRESH
```

Usage & Invocation - Configuration Updates

- The CSSMTP Configuration file is updated with a new AuthEntity parameter on the target server statement. The “Secure” parameter MUST be set to YES if configuring the AuthEntity.
- Most mail servers will only indicate support for AUTH PLAIN or AUTH LOGIN on the EHLO response after a TLS handshake has been completed. To ensure that CSSMTP learns of this support, it is recommended to set TLSEHLO YES in the Options statement.

Configuration for Authenticating CSSMTP as a single entity	MODIFY procname,DISPLAY,CONFIG

TargetServer
{
TargetIp 10.1.1.1
AuthEntity target.mail.server
Secure Yes
}
Options
{
TLSEHLO Yes
}
ReportMailFrom test@test.com
	TARGETSERVER:
	TARGETIP : 10.1.1.1
	CONNECTPORT : 25 CONNECTLIMIT : 5
	MAXMSGSENT : 0 MESSAGESIZE : 524288
	SECURE : YES CHARSET : ISO8859-1
	AUTHENTITY : target.mail.server

Usage & Invocation - Configuration Updates for Email Level Auth

- If email level authentication is needed, <MAILFROM> can be specified for AuthEntity. CSSMTP will use the email address in the MAIL FROM field to retrieve the username and password from RACF.
- If the AuthEntity is set to <MAILFROM>: ReportMailFrom will be used to authenticate undeliverable mails if the ReturnToMailFrom is set to Yes. If ReportMailFrom is empty, undeliverable mails will go to the deadletter directory if the deadletter action is set to Store.

Configuration for Authenticating CSSMTP as a single entity	MODIFY procname,DISPLAY,CONFIG

TargetServer
{
TargetIp 10.1.1.1
AuthEntity <MAILFROM>
Secure Yes
}
Options
{
TLSEHLO Yes
}
ReportMailFrom test@test.com
	TARGETSERVER:
	TARGETIP : 10.1.1.1
	CONNECTPORT : 25 CONNECTLIMIT : 5
	MAXMSGSENT : 0 MESSAGESIZE : 524288
	SECURE : YES CHARSET : ISO8859-1
	AUTHENTITY : <MAILFROM>

Usage & Invocation

Display targets updates

- Modify procname, display, targets command was updated with the Auth State for the target servers

  f cssmtp,d,targets

```
GLOBAL INFORMATION:
  MAIL SENT            : 0        TOTAL RETRY     : 0
  DEADLETTER           : 0        CURRENT RETRY: 0
  UNDELIVER            : 0
  EXTENDED RETRY  :
  CURRENT              : 0        TOTAL           : 0
 TARGET SERVER 10.1.1.1
  STATE          : ACTIVE
  ESMTP          : YES            MESSAGE SIZE    : 10240000
  STARTTLS       : YES            MAIL ATTEMPTS   : 0
  AUTH STATE  : SUCCESSFUL
  MAIL SENT      : 0              CONNECT FAIL    : 0
```

- Auth states

NOT CONFIGURED - AUTH entity not configured
CONFIGURED - AUTH entity configured
SAF FAILURE - Failed to retrieve username or password from SAF
SUCCESSFUL - AUTH connection successful
REJECTED - AUTH command rejected from the server (Could be the wrong username/password sent to the server or some other server problem)
REQUESTED - AUTH support requested by the server, but AUTH entity is not configured in CSSMTP
INVALID CERT - Failed server hostname check against the CA
EMAIL LEVEL - Authentication is done using the email address in the Mail From field

Usage & Invocation - SMF Updates (SMF Type 119, Subtype 48)

- The SMF type 119 subtype 48 record (CSSMTP configuration record) is enhanced to provide the AuthEntity configuration setting.
- Apart from that, a new Key Value to include the ReportMailFrom was added. This was omitted during a previous function update.

Name and Key	Length	Type	Description
SMF119ML_CD_RptMlFrom (47)	1-320	EBCDIC	ReportMailFrom setting
SMF119ML_CD_AuthEnty1 (48)	1-246	EBCDIC	AUTH entity value for target server 1
SMF119ML_CD_AuthEnty2 (49)	1-246	EBCDIC	AUTH entity value for target server 2
SMF119ML_CD_AuthEnty3 (50)	1-246	EBCDIC	AUTH entity value for target server 3
SMF119ML_CD_AuthEnty4 (51)	1-246	EBCDIC	AUTH entity value for target server 4

Usage & Invocation - SMF Updates (SMF Type 119, Subtype 48) cont…

- Two new fields that were previously missing from SMF records were added to the SMF119ML_CF section

  (CSSMTP configuration data section) of the SMF type 119 subtype 48 record.

Offset	Name	Length	Type	Description
162(X'A2')	SMF119ML_CF_RptSysoutCls	1	EBCDIC	ReportSysoutClass is a character field (A-Z or 1-9)
163(X'A3')	SMF119ML_CF_Flag5	1	BIT(8)	Reserved
	SMF119ML_CF_MailbxCmptblty		1… ….	Value from the MailBoxCompatibility statement: 0=Standard, 1=Long
	SMF119ML_CF_rsvd001		.1.. ….	Reserved
	SMF119ML_CF_rsvd002		..1. ….	Reserved
	SMF119ML_CF_rsvd003		…1 ….	Reserved
	SMF119ML_CF_rsvd004		…. 1…	Reserved
	SMF119ML_CF_rsvd005		…. .1..	Reserved
	SMF119ML_CF_rsvd006		…. ..1.	Reserved
	SMF119ML_CF_rsvd007		…. …1	Reserved

Usage & Invocation - SMF Updates (SMF Type 1154, Subtype 4)

- Updates to SMF type 1154, subtype 4 – CSSMTP client compliance evidence record.*

Offset	Name	Length	Type	Description
. . .				
25(X'19')	SMF1154_4_MLTS_Auth	1	Binary	Indicates whether Auth support is enabled. SMF1154_4_MLTS_AUTH_NO (0) - Auth support is not enabled. SMF1154_4_MLTS_AUTH_YES (1) – Auth support is enabled. Configured with the AuthEntity parameter on the Target Server statement.
26(X'1A')		2		Reserved. Set to 0.

- Note: If the CSSMTP AuthEntity is configured along with Secure Yes, Auth connection is attempted with the target server once the server requests it.

Interactions & Dependencies

- Software Dependencies
  - (Recommended Co-requisite): RACF supports encrypting the LDAPBIND entity with AES-256 protection beginning in z/OS V2R5 with APAR OA66458: NEW FUNCTION - Support for stronger encryption for RACF data encrypted with KEYSMSTR class profiles.

- Hardware Dependencies
  - None
- Exploiters
  - None

Upgrade & Coexistence Considerations

- To exploit this solution, all systems in the Plex must be at the new z/OS level: No
- List any toleration/coexistence APARs/PTFs. – N/A
- Upgrade involves only those actions required to make the new system behave as the old one did. – No upgrade action

Installation & Configuration

- Installation : None
- Configuration: CSSMTP configured with TLS enabled to at least one target mail server.

Summary

- CSSMTP is enhanced to support SMTP AUTH PLAIN and LOGIN methods.
- Client and server MUST establish a TLS connection before sending the Auth credentials.
- If the server supports both methods (PLAIN and LOGIN), CSSMTP will always choose PLAIN.
- CSSMTP with attempt the AUTH connection only after the server requests AUTH PLAIN or LOGIN.
- CSSMTP can be configured to do Auth as a single entity or at Email level.
- The system administrator should configure the username and password for the AuthEntity in RACF for secure storage.
- If the authentication is successful, CSSMTP can start forwarding mail to that server. Otherwise, the connection will be terminated.
- If any target server enables email level AUTH and the UNDELIVERABLE statement is set to ReturnToMailFrom Yes, ReportMailFrom will be used to authenticate undeliverable mail. If ReportMailFrom is empty, the undeliverable mail will be stored or deleted depending on the deadletter action.
- Modify procname, display, targets command was updated with the Auth State.
- SMF 119 subtype 48 and SMF 1154 subtype 4 was updated with AuthEntity settings.

Appendix

z/OS Communications Server Publications

- z/OS Communications Server: IP Configuration Guide
- z/OS Communications Server: IP System Administrator's Commands
- z/OS Communications Server: New Function Summary
- z/OS Communications Server: IP Diagnosis Guide
- z/OS Communications Server: IP Configuration Reference